

Forritunarmálið Fjölur
Notendahandbók

Snorri Agnarsson, Páll Björnsson, Jón Harðarson

25. febrúar 1996

Efnisyfirlit

I	Forritunarmálið Fjölur	7
1	Inngangur	9
2	Einingar	11
3	Segðir, stef og aðgerðir	13
3.1	Tölur og stafir	13
3.2	Tómagildið	13
3.3	Textastrengir	13
3.4	Hlunkar	14
3.5	Breytur	16
3.6	Röksegðir	17
3.7	Ef segðin	18
3.8	Aðgerðarsegðir	18
3.9	Fylkjasegðir	20
3.10	Valsegðin	21
3.11	Lykkjusegðir og útsegðin	22
3.12	Listasegðir	23
3.13	Stefköll, stef og skilasegðin	23
4	Einfalt dæmi	27
5	Einingaaðgerðir	29
5.1	Ítrun	29
5.2	Innflutningur	31
5.3	Samsetning	31
5.4	Hliðsetning	31
5.5	Ítrunarhliðsetning	31
6	Veitingar	37
7	Notkun Fjölnis	39
8	Málfræði Fjölnis	41
8.1	Athugasemdir í forritstexta	41
8.2	Frumeiningar málsins	41
8.3	Márit	48
9	Eftirmáli	55

10 Heimildir	57
II Einingin GRUNNUR	59
11 Einingin AFRITA	63
12 Einingin DOSDISK	65
13 Einingin DOSSKRIU	75
14 Einingin DOSSTIU	81
15 Einingin DOSYMISL	85
16 Einingin FLEYFÖLL	89
17 Einingin INN	91
18 Einingin KJARNI	93
19 Einingin LESALINU	113
20 Einingin SKRIFALIN	115
21 Einingin SNUA	117
22 Einingin STRENGIR	119
23 Einingin UT	121
III Viðaukar	123
24 Forritið Fjalla	125
25 Forritið Hvarer	127
26 Einingin BROT	129
27 Einingarnar FELAGRUN og FELAKJAR	133
28 Einingar fyrir fylki	135
28.1 Einingin HFYLKI	135
28.2 Einingin SFYLKI	136
29 Einingin HEXEININ	137
30 Einingin ISLRÖD	139
31 Einingin LYKLAR	141
32 Einingin QUICKSOR	143
33 Einingin MARGLID	145

34 Einingin MENGI	147
35 Einingin RADA	149
36 Einingin SKJAR	151
37 Einingin SLEMBI	155
38 Einingin STORFJOL	157
39 Einingin STORHEIL	161
40 Einingin TAFLA	165
41 Einingin TAYLOR	167
42 Einingin VIDBSKJA	171
43 Einingin VIDFÖNG	175
44 Einingar fyrir teiknun	177
44.1 Einingin TATAHGA	180
44.2 Einingin TATACGA	180
44.3 Einingin TATACGA3	180
44.4 Einingin TATAEGA	180
44.5 Einingin TATAMCGA	180
44.6 Einingin TATAVGA	180
IV Atriðisorðaskrá	181

Hluti I

Forritunarmálið Fjölnir

Kaflí 1

Inngangur

Handbók þessi inniheldur lýsingu á forritunarmálinu Fjölni. Fjölnir er frábrugðin flestum öðrum forritunarmálum á þrennan hátt:

- Fjölnir er íslenskt forritunarmál
- Fjölnir er listavinnslumál
- Fjölnir er einingaforritunarmál

Síðarnefndu atriðin tvö eru það sem gerir Fjölni að öflugum forritunarmáli. Listavinnsla er mikilvægt einkenni á forritunarmáli. Listavinnslan gerir öll gildi í forritunarmálinu jafn auðveld í meðförum og gerir forritara kleift að leysa vandamál á auðveldan hátt sem eru erfið eða nærri ókleif í öðrum forritunarmálum. Einingaforritunin gerir forritaranum síðan kleift að pakka saman sínum lausnum í einingar sem nota má aftur og aftur. Listavinnslan í Fjölni er svipuð og í öðrum listavinnslumálum svo sem LISP og LOGO. Einingaforritunin í Fjölni er aftur á móti mjög frábrugðin einingaforritun í öðrum einingaforritunarmálum svo sem MODULA-2, Ada og C++. Einingaforritunin í Fjölni gefur forritaranum kost á að gera fleira en í þessum forritunarmálum og á einfaldari hátt.

Kafli 2

Einingar

Forrit í Fjölni eru byggð úr einingum. Fjölnir gefur möguleika á að skilgreina einingar sem eru söfn stefja (undirforrita) og breyta, og tengja einingarnar saman á alla hugsanlega vegu. Stef og breytur sem skilgreindar eru með nafni í einhverri einingu eru sögð vera flutt út úr þeirri einingu eða sagt er að einingin innihaldi þessi stef og breytur. Í stefjum innan einingar má kalla á önnur stef eða nota breytur utan einingarinnar og kallast slík tilvísun innflutningur, sagt er að einingin vísi í slík stef eða breytur. Einingaraðgerðir eru notaðar til að tengja þessar tilvísanir raunverulegum stefjum og breytum í einhverri einingu.

Grundvallarhugmyndin í aðferð Fjölnis við einingaforritun er sú að líta á einingar sem innsetningar, svipað og í stærðfræðinni. Eining sem t.d. inniheldur atriðin a, b og c skilgreinir þá innsetningu sem setur tiltekna skilgreiningu í stað nafnanna a, b og c.

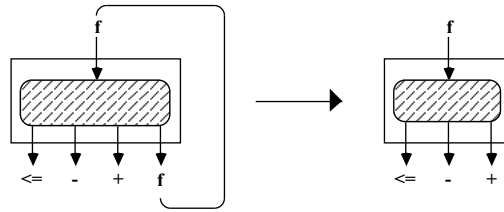
Dæmi um forritun einingar:

```
"fibo" =
{
f ->
  stef(;n)
  stofn
  ef n <= 2 þá
    skila 1
  annars
    skila f(;n-1)+f(;n-2)
  eflok
  stofnlok
};
```

Þegar skrá er þýdd sem inniheldur textann að ofan verður eining skrifuð í skrána FIBO.EIN. Þessi eining inniheldur stefið f, og vísar í stefin f, <=, - og +. Við lítum á þessa einingu sem innsetningu sem setur stefið hægra megin örvarinnar í stað nafnsins f vinstra megin örvarinnar. Nöfnin sem koma fyrir vinstra megin við ör segjum við vera útflutt úr einingunni. Stef sem kallað er á úr einingunni segjum við vera innflutt í eininguna.

Nú er eðlilegt að ætla að stefið f eigi hér að vera endurkvæmt. Til þess að svo megi verða, tengjum við tilvísunina í f (hið innflutta f) við útflutta stefið f í sömu einingu. Til þess notum við einingaaðgerðina ítrun. Aðgerð sú er táknuð með stafnum “!”. Við getum myndað nýja einingarskrá IFIBO.EIN með eftirfarandi forritstexta:

```
"ifibo" = ! "fibo" ;
```



Mynd 2.1: Einföld ítrun

Sú eining inniheldur endurkvæmt stef f , og vísar í (flytur inn) tvíundarstefin \leq , $-$ og $+$. Þessari aðgerð er lýst í mynd 2.1.

Ástæðan fyrir því að við notum slíka aðgerð í Fjölni, í stað þess að hafa það sjálfgefið að stef séu endurkvæm, er sú að við viljum gefa allar tengingar undir stjórn forritarans. Með ítrun og öðrum einingaáðgerðum má tengja einingar saman á allan hugsanlegan hátt. Aðrar nauðsynlegar einingaáðgerðir heita hliðsetning, samsetning og innflutningur. Hliðsetning er notuð til að steypa saman tveim einingum án þess að tengja nokkuð. Samsetning og innflutningur tengja eina einingu ofan á aðra, á eilítið mismunandi hátt. Við munum síðar sjá nákvæmlega hvernig þessar áðgerðir vinna.

Kafli 3

Segðir, stef og aðgerðir

Við höfum nú rétt fengið nasasjón af einingaaðgerðum þeim sem Fjölnir býður upp á. Við skulum nú sjá hvaða segðir Fjölnir býður upp á. Segð (e. expression) er formúla eða þvíumlíkt sem skilar gildi þegar reiknað er úr henni. Í mörgum forritunarmálum eru notaðar bæði segðir og setningar (e. *statement*), þar sem setningar eru skipanir sem ekki skila gildi. Í Fjölni er ekki gerður greinarmunur á setningu og segð. Allar segðir skila einhverju gildi.

3.1 Tölur og stafir

Tölur í Fjölni eru ýmist fjöldatölur, heiltölur eða fleytitölur. Stafir eru táknaðir með einum staf með einföldum gæsalöppum utan um, til dæmis 'a'. Reyndar eru stafir jafngildir tölum. Til dæmis er stafurinn 'a' jafngildur tölunni 97 vegna þess að 'a' er stafur 97 í ASCII táknrófinu. Fjöldatölur hafa gildi frá 0 til 65535. Heiltölur hafa gildi frá -32768 til 32767. Fleytitölur hafa gildi frá 1.11×10^{-308} til 1.79×10^{308} , um það bil. Nákvæmar tiltekið er fleytitala í Fjölnisforriti annað hvort núll eða hana má skrifa sem $(-1)^f \times (1 + \frac{n}{2^{17}}) \times 2^k$, þar sem n er milli 0 og 65535, k er milli -1023 og 1023 og f er 0 eða 1.

3.2 Tómagildið

Eitt gildi hefur nokkra sérstöðu í Fjölnisforritum. Það er kallað tómagildið eða ósatt. Gildi þetta er m.a. notað sem sanngildið "ósatt" en öll önnur gildi eru látin standa fyrir sanngildið "satt". Sanngildi eru notuð í ýmsum segðum til að stýra útreikningum. Í forritum er táknið [] látið standa fyrir tómagildið. Tómagildið er ekki tala. Talan 0 hefur því sanngildið "satt".

3.3 Textastrengir

Strengir eru táknaðir með runu af stöfum með tvöföldum gæsalöppum utan um, til dæmis "dæmi um streng" eða "". Strengir sem felldir eru inn í forritstextann, eins og í segðinni

```
x := "abc"
```

eru fylki af bætum, og inniheldur fyrsta bætið (bætið í sæti núll) lengd þess texta sem strengurinn inniheldur. Varast ber að breyta slíkum strengjum, heldur taka þá fyrst afrit af strengnum til þess að breyta ekki strengfastanum. Hafa skal í huga að stafur 'd' er ekki það sama og strengur "d".

Í grunneiningunni GRUNNUR sem flest forrit nota eru m.a. stef til að meðhöndla strengi. Eftirfarandi strengjastef eru í grunninum:

strengur Einundarstef sem býr til streng af tiltekinni lágmarksstærð. Viðfangið er tala og strengurinn sem skilað er, er a.m.k. af þeirri stærð. Hámarksstærð strengja getur verið mismunandi eftir því hvaða grunneining er notuð, en er a.m.k. 4096 bæti (4K).

strengsetjabæti Stef til að setja tiltekna tölu í tiltekið sæti (bæti) í tilteknum streng. Ef breytan x inniheldur streng af lágmarksstærð 12 þá veldur segðin `strengsetjabæti(;x,11, 'a')` því að stafurinn 'a' er settur í sæti 11 í strengnum x .

strengsækjabæti Tvíundarstef til að sækja bæti úr tilteknu sæti í tilteknum streng. Ef x er strengur af lágmarksstærð 30 þá veldur segðin

```
y := x \strengsækjabæti 29
```

því að breytan y fær tölugildi sem er bætið í sæti 29 í strengnum x .

strengsetjaorð Stef sem setur tiltekið orð (tvö bæti) á tiltekinn stað í tilteknum streng. Ef x er strengur af lágmarksstærð 13 þá veldur segðin

```
strengsetjaorð( ;x,11,257)
```

því að talan 257 er sett í strenginn x í sæti 11 og 12.

strengsækjaorð Tvíundarstef sem sækir orð (tölu) úr streng. Ef x er strengur af lágmarksstærð 10 þá skilar segðin

```
x \strengsækjaorð 8
```

orði því sem geymt er í sætum 8 og 9 í x .

erstrengur Einundarstef sem segir til um hvort viðfangið er strengur. Til dæmis skilar segðin

```
\erstrengur "abc"
```

sanngildinu "satt" (þ.e.a.s. ekki tómagildinu) en segðin

```
\erstrengur 12
```

skilar "ósatt" (tómagildinu).

strengstærð Einundarstef sem tekur streng sem viðfang og skilar stærð hans. Stærð strengja er reyndar ávallt eitthvert veldi af tveimur. Til dæmis skilar segðin

```
\strengstærð "abcd"
```

tölunni 8 (eitt bæti er notað fyrir lengdina þrjár, fjögur bæti eru fyrir stafina a,b,c og d, og þrjú bæti eru ónotuð).

3.4 Hlunkar

Í Fjölni eru gildi sem kallast hlunkar. Hlunkar þessir eru fylki af öðrum gildum. Hlunkar eru til af mismunandi stærðum, allt frá því að innihalda einungis eitt gildi upp í að innihalda 1024 mismunandi gildi. Einingin GRUNNUR inniheldur stef til að búa til nýja hlunka, setja tiltekið gildi í tiltekið sæti í tilteknum hlunk, og sækja gildið úr tilteknu sæti í tilteknum hlunk. Ein tegund hlunka er sérlega mikilvæg, en það er hlunkur með tveim sætum. Sérstök stef eru í grunninum til að meðhöndla slíka hlunka vegna þess hve mikið slíkir hlunkar eru notaðir. Slíkir hlunkar eru kallaðir pör.

Almenn stef í grunninum til að meðhöndla hlunka.

hlunkur Einundarstef sem býr til nýjan hlunk. Viðfangið er stærð hlunksins, og öll sætin í nýja hlunknum hafa tómagildið sem upphafsgildi. Segðin

```
x := \hlunkur 10
```

veldur því að breytan `x` inniheldur hlunk af lágmarksstærð 10. Sætin í hlunknum eru númeruð frá 0 til 9.

hlunksetja Stef til að setja gildi í tiltekið sæti í hlunk. Ef `x` er hlunkur af stærð a.m.k. 31 þá veldur segðin

```
hlunksetja(;x,30,[1,2,3])
```

því að sæti 30 í `x` inniheldur listann `[1,2,3]`. Stefið `hlunksetja` skilar ekki neinu sérstöku gildi.

hlunksækja Tvíundarstef til að sækja gildið úr tilteknu sæti í hlunk. Ef `x` er hlunkur af lágmarksstærð 13 þá veldur segðin

```
y := x \hlunksækja 12
```

því að breytan `y` fær gildið í sæti 12 í hlunknum `x`.

hlunkstærð Einundarstef sem skilar stærð hlunks. Ef `x` er hlunkur af stærð 64 þá veldur segðin

```
y := \hlunkstærð x
```

því að breytan `y` fær gildið 64. Reyndar er stærð allra hlunka eitthvert veldi af tveim. Þegar hlunkur af tiltekinni lágmarksstærð er búinn til, er stærðin ákvörðuð sem minnsta veldi af tveim sem fullnægir lágmarksskilyrðinu.

erhlunkur Einundarstef sem segir til um hvort viðfangið er hlunkur. Til dæmis skilar segðirnar

```
\erhlunkur 1
```

og

```
\erhlunkur "meðalstór strengur"
```

báðar “ósatt” (þ.e.a.s. tómagildinu), en segðin

```
\erhlunkur \hlunkur 10
```

skilar “satt” (ekki tómagildinu).

Stef til að meðhöndla pör.

haus Einundarstef sem tekur par sem viðfang og skilar gildi því sem geymt er í sæti 0 í parinu. Gildið í sæti 0 í tilteknu pari köllum við haus parsins.

hali Einundarstef sem tekur par sem viðfang og skilar gildi því sem geymt er í sæti 1 í parinu. Gildið í sæti 1 í tilteknu pari köllum við hala parsins.

: Tvíundarstef til að búa til par. Viðföngin tvö verða haus og hali parsins. Til dæmis skilar segðin `1: [2,3]` listanum `[1,2,3]`.

erpar Einundarstef sem segir til um hvort viðfangið er par. Til dæmis skila segðirnar

```
\erpar 1
```

og

```
\erpar "meðalstór strengur"
```

báðar “ósatt” (þ.e.a.s. tómagildinu), en segðirnar

```
\erpar [1,2]
```

og

```
\erpar \hlunkur 2
```

skila “satt” (ekki tómagildinu).

Lýsa má innbyrðis samhengi stefjanna `haus`, `haus` og `:` með eftirfarandi jöfnum:

```
\haus (x:y) = x
\hali (x:y) = y
```

Eftirfarandi eining inniheldur stefið lengd, sem skilar lengd viðfangsins, sem verður að vera eiginlegur listi:

```
!{
lengd ->
  stef(;x)
  stofn
    ef x þá
      \stækka \lengd \hali x,
    annars
      0,
    eflok,
  stofnlok
}
```

3.5 Breytur

Stef í Fjölni geta innihaldið staðværar breytur, sem einungis eru til staðar meðan stef það sem inniheldur þær er í framkvæmd. Í hvert skipti sem hafin er framkvæmd stefs eru búnar til nýjar útgáfur af þessum staðbundnu breytum, og þeim er öllum gefið tómagildið sem upphafsgildi nema annað sé tekið fram. Vísa má í breytur til að nota gildi þeirra, og breytur má gefa gildi með gildisveitingarsegð, til dæmis veldur segðin

```
x:=x+1
```

Því að breytan `x` er stækkuð um einn (við gerum ráð fyrir að tvíundarstefið “+” sé tengd okkar venjulega samlagningarstefi á fjöldatölur). Einnig má breyta gildum breyta með stefköllum, eins og við munum sjá seinna.

Einingar geta einnig innihaldið víðværar (e. *global* eða *external*) haldgildar (e. *static*) breytur. Breytur þessar halda sínu gildi milli kalla á stef í einingunni og taka þátt í öllum einingaaðgerðum. Tilvísun í slíka breytu er aðeins lögleg ef breytan hefur verið skilgreind sem innflutt í haus stefsins. Tilvísunina verður þá að tengja útfluttri breytu úr einhverri einingu. Einingin


```
{ x -> breyta }
```

flytur út breytu x . Allar víðværar breytur innihalda tómagildið við upphaf keyrslu forritsins. Með eingaaðgerðunum má tengja víðværar breytur og smíða einingar sem innihalda faldar breytur, sem notandi einingarinnar hefur ekki aðgang að, en halda sínu gildi milli kalla. Eftirfarandi eining inniheldur stefið slembi sem skilar nýrri slembitölu í hverju kalli. Breytan sæði er falin innan einingarinnar og notendur einingarinnar hafa ekki aðgang að henni.

```
"slembi" =
{
slembi ->
  stef(;)
  innflutt sæði
  stofn
    ef sæði þá
      sæði:=sæði * 513 + 1801,
    annars
      sæði:=12345,
    eflok,
  stofnlok
}
*
{ sæði -> breyta }
;
```

3.6 Röksegðir

Fjölirnir býður upp á rökaðgerðirnar “ekki”, “og” og “eða”. Þessar aðgerðir eru að tvennu leyti frábrugðnar stefjum í Fjölni. Í fyrsta lagi er skilgreining þeirra föst, þar eð þær eru ekki fluttar inn í þær einingar sem nota þær, og einingar geta ekki flutt þessar aðgerðir út. Í öðru lagi eru tvífundarrökaðgerðirnar “og” og “eða” búnar þeim eiginleika að þær athuga fyrst gildi vinstra viðfangsins, og gildi hægra viðfangsins er ekki reiknað nema nauðsyn beri til. Segðin ekki S skilar tómagildinu þá og því aðeins að S skili ekki tómagildinu. Að öðru leyti er útkoman ekki skilgreind, og skal því einungis nota hana sem rökildi. Segðin S_1 og S_2 og \dots og S_n reiknar gildi segðanna S_1 til S_n í röð þar til ein þeirra skilar tómagildinu. Ef engin þeirra skilar tómagildinu þá skilar segðin gildinu sem S_n skilaði. Ef S_i er fyrsta segðin í rununni sem skilar tómagildinu þá skilar runan tómagildinu og ekki er reiknað úr segðunum S_{i+1} til S_n . Svipað gildir um segðina S_1 eða S_2 eða \dots eða S_n . Hún reiknar gildi segðanna S_1 til S_n í röð þar til ein þeirra skilar öðru en tómagildinu. Ef allar þeirra skila tómagildinu þá skilar segðin tómagildinu. Ef S_i er fyrsta segðin í rununni sem skilar ekki tómagildinu þá skilar runan því gildi og ekki er reiknað úr segðunum S_{i+1} til S_n . Rökaðgerðirnar hafa lægsta forgang af öllum aðgerðum, og innbyrðis forgangur þeirra er slíkur að “ekki” er framkvæmd fyrst, síðan “og”, og að lokum “eða”. Sem dæmi um notkun þessara aðgerða má taka segðina

```
x=0 og villa(;"deilt með núlli") eða y/x
```

Í þessari segð er fyrst athugað hvort $x=0$. Ef svo er þá er segðin

```
villa(;"deilt með núlli")
```

framkvæmd, sem væntanlega veldur því að forritið stöðvast. Að öðrum kosti er segðin y/x reiknuð og gildi hennar skilað sem gildi allrar segðarinnar.

3.7 Ef segðin

Dæmi um ef segðina gæti litið þannig út:

```
ef x<10 þá 1 annarsef x<100 þá 2 annars 3 eflok
```

Þessi ef segð er framkvæmd á eftirfarandi hátt: Fyrst er náð í gildið á breytunni x . Síðan er tvíundarstefinu “<” beitt á það gildi og töluna 10. Út úr þeirri beitingu fáum við eitthvert gildi. Ef það gildi er satt (þ.e.a.s. ekki tómagildið) þá skilar ef segðin tölunni 1. Ef gildið er tómagildið þá er næsti möguleiki reiknaður, og gildi breytunnar x er borinn saman við töluna 100. Úr þeim samanburði (eða stefbeitingu) fáum við aftur gildi sem annað hvort er satt (ekki tómagildið) eða ósatt (tómagildið). Ef gildið er satt þá skilar ef segðin tölunni 2, annars skilar hún tölunni 3.

Almennt geta ef segðir innihaldið ótakmarkaðan fjölda af “annarsef” hlutum, og “annars” hlutinn má vera til staðar en þarf ekki að vera til staðar. Ef segðin skilar ávallt gildi síðustu reiknaðrar segðar. Til dæmis skilar segðin

```
ef 1>2 þá 1 eflok
```

tómagildinu, og segðin

```
ef 2>1 þá eflok
```

skilar einnig tómagildinu.

3.8 Aðgerðarsegðir

Í Fjölni má beita þeim stefjum sem aðgerðum sem einungis hafa inntaksviðföng. Einundaraðgerðum er beitt með því að setja þær framan við segðir þær sem beita skal þeim á. Til dæmis eru

```
- x
```

og

```
\ekki x
```

tvær löglegar segðir með einundarstefjum “-” og “ekki”. Aðgerðanöfn eru ýmist táknuð með stafnum “\” framan við stafarunu (sjá nánar í kaflanum um málfræði Fjölnis) eða runu frátekkinna aðgerðarstafa.

Lögleg aðgerðanöfn eru til dæmis +, -, \haus, ++, <<, \hali, \plús, \summa og <=>.

Ólögleg aðgerðanöfn eru til dæmis +summa og plús.

Tvíundaraðgerðum má beita með því að setja þær milli þeirra segða sem beita skal þeim á. Til dæmis eru

```
x+y
```

og

```
x \módúlus y
```

löglegar segðir með tvíundarstefjum “+” og “módúlus”. Gildi viðfanga aðgerðar eru reiknuð frá vinstri til hægri. Aðgerðarnöfnum má beita eins og stefnöfnum, til dæmis er segðin

```
+ (;1,2)
```

jafngild segðinni

```
1+2
```

Forgangur	Fyrsti stafur aðgerðarnafns
7	*, / eða %
6	+ eða -
5	<, > eða =
4	
3	&
2	:
1	allir aðrir stafir

Tafla 3.1: Forgangur aðgerða

Tvíundaraðgerðir hafa mismunandi forgang, og er forgangi aðgerða úthlutað eftir því hvaða staf nafn þeirra hefst á, eins og sýnt er í töflu 3.1.

Fjölnisþýðandinn mun sjá til þess að aðgerðir með hærri forgangi séu framkvæmdar á undan aðgerðum með lægri, nema svigar stýri röðinni nánar. Einundaraðgerðir hafa hærri forgang en tvíundaraðgerðir. Segðirnar

```
\mínus x < y & y < z
```

og

```
((\mínus x) < (y & y)) < z
```

eru því jafngildar.

Athugið að rökaðgerðirnar “eða”, “og” og “ekki” hafa minni forgang en allar aðrar aðgerðir. Segðin

```
ekki x<y og y<z
```

vinnur því eins og við eigum að venjast, hún er jafngild segðinni

```
(ekki (x<y)) og (y<z)
```

Tvíundaraðgerðir tengjast til vinstri, fyrir utan aðgerðir sem hafa nöfn sem byrja á tákniinu “:”, en þær tengjast til hægri. Segðin

```
1+2+3
```

er því jafngild segðinni

```
(1+2)+3
```

en segðin

```
1:2:[]
```

er jafngild segðinni

```
1:(2:[])
```

Skilgreina má nýjar aðgerðir á sama hátt og önnur stef eru skilgreind. Til dæmis mætti skilgreina veldisupphafningu á eftirfarandi hátt:

```
"veldi" =
!
{
^ ->
stef(;x,n)
```

```

    stofn
ef n=0 þá
    1,
    annarsef n%2=1 þá
        x*((x*x)^(n/2)),
    annars ;; ef n%2=0 þá
        (x*x)^(n/2),
    eflok,
    stofnlok
}
;

```

Þessa aðgerð má þá kalla á í segð svo sem 2^{10} , sem þá væntanlega skilar gildinu 1024. Þetta er hraðvirk aðferð til veldishafningar í fjöldatöluveldi.

3.9 Fylkjasegðir

Fjöldnir býður upp á segðir svo sem

```
f[i] := f[i] + 1,
```

og

```
x := f[k] + 2,
```

til að auðvelda notkun á gögnum sem hegða sér eins og fylki. Mikilvægt er að gera sér grein fyrir því að fylki í Fjöldni eru alls ekki eins fastmótuð að merkingu og almennt gerist í forritunarmálum. Almennt er gildisveiting í Fjöldni á forminu

```
f[i]:=g
```

nákvæmlega jafngild stefkallinu

```
fylkissetja1(;f,i,g)
```

og fylkissegð á forminu

```
f[i]
```

er nákvæmlega jafngild segðinni

```
fylkissækja1(;f,i)
```

Það er á ábyrgð notandans að þessi undirforrit (fylkissetja1 og fylkissækja1) séu hönnuð á þann veg að f hagi sér sem fylki eða tafla. Almennt kemur vídd fylkjanna fram í nöfnum þeirra stefja og aðgerða sem notuð eru til að setja gildi í fylki og sækja gildi úr fylki. Segðin

```
f[i1,...,iN]
```

er því jafngild segðinni

```
fylkissækjaN(;f,i1,...,iN)
```

og segðin

```
f[i1,...,iN]:=g
```

er jafngild segðinni

```
fylkissetjaN(;f,i1,...,iN,g)
```

Sem dæmi um notkun fylkjasegða mætti tengja í grunninn með tengingunum

```
{
fylkissækja1 -> strengsækjabæti
fylkissetja1 -> strengsetjabæti
}
```

og þá mætti nota segðina `s[10]` til að vísa í bæti 10 í strengnum `s`, og `s[1]:='a'` til að setja stafinn 'a' í bætsæti 1 í strengnum.

Í einingunni GRUNNUR eru fylkissækja1 og fylkissetja1 tengdar stefjum sem vinna eins og strengsetjabæti, hlunksetja, strengsækjabæti og hlunksækja, eftir atvikum. Því má nota segðina `h[10]` til að vísa í gildi 10 í hlunknum `h`, og segðina `s[10]` til að sækja bætigildið í sæti 10 í strengnum `s`.

Einnig eru tiltækar einingar til að meðhöndla almennari fylki, til dæmis strjál fylki, og notandinn getur að sjálfsgöðu hannað sín eigin fylki.

Fylki sem innihalda fylki má nota sem fjölvíð fylki. Til dæmis er segðin

```
f[i][j]:=10,
```

fullkomlega lögleg segð sem er jafngild segðinni

```
fylkissetja1(;f[i],j,10)
```

eða

```
fylkissetja1(;fylkissækja1(;f,i),j,10)
```

3.10 Valsegðin

Fjölnir býður upp á valsegð sem líkist "case", "select" eða "switch"-setningum í öðrum forritunarmálum. Í valsegðinni er eitt fjöldatölugildi notað til velja einhvern valkost úr endanlegum fjölda kosta. Dæmi um valsegðina er

```
val i%15 úr
kostur 0 þá
    skrifastreng("talan 15 gengur upp í i"), 2,
kostur 5,10 þá
    skrifastreng("talan 5 gengur upp í i"), 1,
kostur 3,6,9,12 þá
    skrifastreng("talan 3 gengur upp í i"), 1,
annars
    skrifastreng("hvorki 5 né 3 ganga upp í i"), 0,
vallok
```

Eins og allar aðrar segðir í Fjölni skilar valsegðin gildi. Gildið sem valsegðin skilar er gildi síðustu segðar sem reiknuð er innan valsegðarinnar. Í dæminu að ofan skilar segðin því fjölda þeirra prímatalna sem gengið hefur verið úr skugga um að gangi upp í `i`. Ekki er nauðsynlegt að hafa annars-hluta í valsegðinni. Sé annars-hlutanum sleppt jafngildir það því að annars-hlutinn sé

```
annars
[]
```

Kostirnir sem taldir eru upp í valsegðinni mega einungis innihalda fjöldatölufasta og staffasta. Setja má kosti á forminu `a..b` sem jafngilda upptalningunni `a,a+1,...,b`. Til dæmis

```

val s úr
kostur '0'..'9' þá
    skrifastreng("tölustafur"),
kostur 'a'..'z','A'..'Z', þá
    skrifastreng("bókstafur"),
vallok

```

3.11 Lykkjusegðir og útsegðin

Fjöldnir býður upp á þrjár gerðir af lykkjusegðum. Dæmi um lykkjusegðir eru:

```

meðan x>1 lykkja
    lógariþmi:=lógariþmi+1,
    x:=x/2,
lykkjulok

```

og

```

fyrir( i:=1 ; i<10 ; i:=i+1) lykkja
    skrifatölu(i),
    skrifastreng(" í öðru veldi er "),
    skrifatölu(i*i),
    nýlína(;),
lykkjulok

```

og

```

lykkja
    x:=lesatölu(;),
    ef x=0 þá út eflok,
    y:=y+x,
lykkjulok

```

Meðan lykkjan vinnur þannig að í byrjun hvorrar umferðar er reiknað út úr skilyrðinu, sem táknað er með segðarunu milli lykilorðanna “meðan” og “lykkja”. Ef útkoman úr þeim reikningi er sönn (ekki tómagildið) þá er reiknað úr segðunum milli “lykkja” og “lykkjulok”, einni af annarri, og síðan byrjað á nýtt.

Fyrir segðin vinnur líkt og “for” setning í C forritunarmálinu. Segðin

```

fyrir(A;B;C) lykkja
    D,
lykkjulok

```

vinnur svipað og segðarunan

```

A,
meðan B lykkja
    D,
    C,
lykkjulok

```

fyrir utan það að út-segðir í segðarununum A, B og C eru ekki tengdar þessari lykkju, heldur ytri lykkju, ef einhver er. Þriðja gerðin af lykkjusegð er óendanleg lykkja. Til að ljúka framkvæmd óendanlegrar lykkjusegðar má framkvæma útsegð inni í henni. Þegar útsegð er framkvæmd veldur hún því að framkvæmd lykkjunnar er lokið og lykkjan skilar sínu gildi, sem er alltaf hið sama fyrir allar tegundir lykkjusegðar, þ.e.a.s. tómagildið.

Einnig má að sjálfsögðu ljúka lykju með því að framkvæma skilasegð eða einfaldlega hætta keyrslu forritsins með því að kalla á stýrikerfið á viðeigandi hátt (t.d. stefið hætta í einingunni GRUNNUR).

Meðan lykkju má breyta í jafngilda óendanlega lykkju. Meðan lykkjan

```
meðan A lykkja
  B,
  lykkjulok
```

er jafngild lykkjunni

```
lykkja
  ef A þá
    [], ;; gerir ekkert
  annars
    út, ;; stekkur út úr lykkjunni
  eflok,
  B,
  lykkjulok
```

nema að hliðstætt gildir um út-segðir í segðarununni A og gildi í fyrir-lykkjunni.

3.12 Listasegðir

Segðir eins og $[1, a, x+1]$ eru kallaðar listasegðir. Segð þessi er jafngild segðinni

```
1:a:(x+1):[]
```

Almennt er segðin

```
[A,B,...,F]
```

jafngild segðinni

```
A:B:...:F:[]
```

Merking slíkra segða er því háð merkingu aðgerðarinnar “:”, nema að segðin [] hefur fasta merkingu, hún skilar alltaf tómagildinu.

3.13 Stefköll, stef og skilasegðin

Samkvæmt málfræði Fjölnis (sjá málrít í kaflanum um málfræði) líta stefskilgreiningar svona út:

```
nafn ->
  stef(runa_af_nöfnum;runa_af_nöfnum)
  breytuskilgreiningar
  stofn
    runa_af_segðum
  stofnlök
```

Til dæmis:

```
stærstisameiginlegideilir ->
  stef(n;x,y)
  staðvær t
  stofn
```

```

n:=1,
lykkja
  ef x=0 þá
    út,
  eflok,
  n:=n+1,
  t:=y%x,
  y:=x,
  x:=t,
lykkjulok,
skila y,
stofnlok

```

Stefið stærstisameiginlegideilir að ofan tekur þrjú viðföng, n , x og y . Viðfangið n er innútviðfang. Það þýðir að þegar kallað er á stefið má setja breytu fyrir leppinn n í kallsegðinni, og sú breyta fær þá ef til vill nýtt gildi við kallið. Hin viðföngin, x og y , eru gildisviðföng. Í kalli má þá setja gildi fyrir leppana x og y . Stefið að ofan má kalla á í segð, til dæmis

```
d:=stærstisameiginlegideilir(k;100,125)
```

Slík segð veldur því að gildin 100 og 125 eru sett í breytur x og y í stofni stefsins, og breytan n í stofni stefsins fær gildi breytunnar k í kallsegðinni. Stefið er síðan framkvæmt og skilar að lokum sínu gildi sem sett er í breytuna d . En breytan k fær einnig gildi það sem breytan n í stefinu hafði við lok framkvæmdar stefsins. Kallaðferðin byggist því á því að senda gildi fram og til baka milli kallsegðar og stefs sem kallað er á.

Lítum á annað dæmi. Ef kallað er á stefið

```

f ->
  stef(x,y;a,b)
  stofn
    x:=a,
    y:=b,
  stofnlok

```

með kallinu

```
f(z,z;1,2)
```

þá fær breytan z gildið 2 vegna þess að afritunin af gildunum fer fram frá vinstri til hægri. Ef engin skilasegð er framkvæmd í stefinu sem kallað er á, þá skilar kallið gildi síðustu segðar sem reiknað er úr í kallstefinu. Segðin `skila y` í fyrra dæminu að ofan hefði því eins getað verið einfaldlega `y`.

Ef segðin

```
skila 10
```

er framkvæmd þá veldur hún því að inningu stefs er hætt, og tölunni 10 er skilað til kallsegðar sem gildið úr kallinu. Almennt veldur

```
skila S
```

þar sem S er einhver segð því að reiknað er gildið á segðinni S , og gildi því er skilað til baka í kallsegðina.

Þegar vísað er í stef í kalli kallast það innflutningur á viðkomandi stefi þar eð tilvísunina þarf einhverna tíma að tengja útfluttu stefi í einhverri einingu.

Önnur aðferð til að kalla á stef er sú að geyma stefið í breytu og kalla á það seinna. Til að geyma stef f í breytu x notum við gildisveitingu eins og eftirfarandi:


```
x := stef f(1,2)
```

segðin `stef f(1;2)` vísar í stefið `f`, sem verður að vera stef með einu innútvíðfangi og tveimur innviðföngum. Almennt getum við smíðað svokallað stefgildi með segð á sniðinu

```
stef <stefnafn>(<innútfjöldi>;<innfjöldi>)
```

Seinna má kalla á stefið með segð á sniðinu

```
<nafn breytu>(<innútvíðföng>;<innviðföng>)
```

þar sem breytan inniheldur viðkomandi stefgildi.

Kafli 4

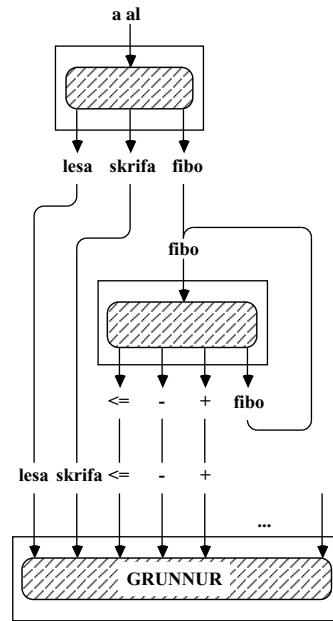
Einfalt dæmi

Eftirfarandi forrit les inn fjöldatölu, beitir Fibonacci fallinu á hana, og prentar útkomuna.

```
"forrit" < aðalstef
{
aðalstef ->
    stef();
    staðvær n,m
    stofn
        n:=lesa();
        m:=f(;n),
        skrifa(;m)
    stofnlok
}
*
!
{
f ->
    stef(;n)
    stofn
        ef n<=2 þá
            1
        annars
            f(;n-1)+f(;n-2)
        eflok
    stofnlok
}
*
"grunnur"
;
```

Í þessu forriti eru notaðar þrjár einingar. Fyrsta einingin inniheldur stefið aðalstef sem er stef það sem framkvæmt er þegar forritið er keyrt. Næsta eining inniheldur stefið f, og á þá einingu er beitt ítrunaraðgerðinni til þess að tengja endurkvæmar tilvísanir í f. Innflutningsaðgerðinni er síðan beitt til að tengja stefið f við tilvísunina í aðalstefinu. Út úr þeirri aðgerð fáum við einingu sem inniheldur aðalstef, og vísar í stefin lesa, skrifa, +, - og <=. Mynd 4.1 lýsir tengingum þessum.

Einingin GRUNNUR er loks flutt inn, en hún skilgreinir ýmis grunnstef svo sem lesa, skrifa, +, - og <=. Þar með höfum við einingu sem inniheldur aðalstef, og vísar ekki í neitt. Þessa einingu skrifum við sem forrit



Mynd 4.1: Fibonacci forrit

í skrána FORRIT.EXE. Keyrsla þessa forrits felst í því að framkvæma aðalstef einu sinni.

Forritið að ofan sýnir notkun á innflutningi og ítrun. Ítrun er eina einingaaðgerðin sem beitt er á staka einingu, og áhrif hennar eru að tengja innflutt atriði (stef eða breytur) við útflytt atriði af sama nafni. Útkoman er því eining sem inniheldur sömu nöfn og upphaflega einingin, en vísar aðeins í þau nöfn sem ekki eru til í einingunni.

Innflutningur tengir útflytning úr seinni einingunni við innflutning í þeirri fyrri, og einingin sem út kemur inniheldur sömu nöfn og fyrri einingin, og vísar í öll nöfn sem vísað er í úr seinni einingunni og þau sem vísað er í úr þeirri fyrri og ekki eru til í þeirri seinni. Við munum sjá nánari skilgreiningu á þessum aðgerðum hér á eftir.

Kafli 5

Einingaaðgerðir

Fjöldnir býður upp á fimm einingaaðgerðir. Ein þeirra, ítrun, er einundaraðgerð. Hún er táknuð með aðgerðarstafnum "!" . Hinar eru tvíundaraðgerðir, kallaðar innflutningur, samsetning, hliðsetning og ítrunarhliðsetning. Þær eru táknaðar með aðgerðarstöfunum "*", ":", "+", og "&". Tvíundaraðgerðirnar eru framkvæmdar frá vinstri til hægri, t.d. gildir jafnan $A * B * C = (A * B) * C$ fyrir allar einingar A, B og C. Forgangur tvíundaraðgerðanna er slíkur að fyrst er innflutningur (táknaður með "*") framkvæmdur, síðan samsetning (táknuð með ":"), síðan hliðsetning (táknuð með "+") og loks ítrunarhliðsetning (táknuð með "&"). Einundaraðgerðin ítrun hefur hæsta forgang. Í eftirfarandi köflum er aðgerðum þessum lýst.

5.1 Ítrun

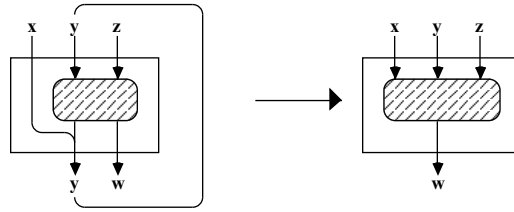
Íhugum einingu "A" sem skilgreind er með eftirfarandi forritstexta:

```
"A" =
{
x -> y
y ->
    stef(;a,b)
    stofn
        y(;1,2),
        w(a,b;3),
    stofnlök
z ->
    stef(;a,b)
    stofn
        y(;3,1),
        w(b,a;4),
    stofnlök
}
;
```

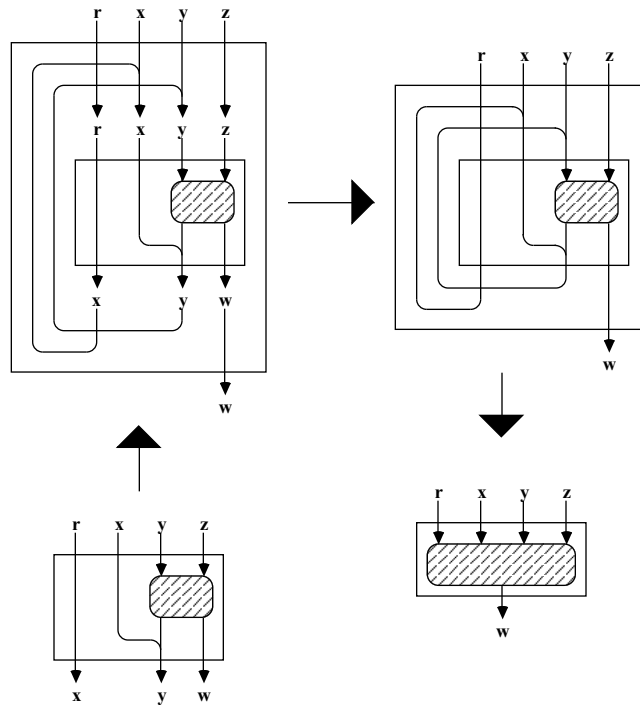
Eining þessi inniheldur nöfnin x,y og z. Hún vísar í y og w. Mynd 5.1 lýsir tengingum sem gerðar eru við ítrun þessarar einingar. Einingin !"A" inniheldur því atriðin x, y og z, og vísar í atriðið w. Útflutningarnir x og y eru reyndar sama stefið.

Lítum nú á flóknara dæmi. Látum eininguna "B" vera skilgreinda með:

```
"B" =
{
```



Mynd 5.1: Einfalt dæmi um ítrun



Mynd 5.2: Ítarlegt dæmi um ítrun

```

r -> x
x -> y
y ->
    stef(a,b;c)
    stofn
        w(a;1,2),
        y(b,a;2),
    stofnlok
z ->
    stef(;a)
    stofn
        skila a
    stofnlok
}
;

```

Ítrun einingarinnar "B" má þá lýsa með mynd 5.2. Einingin !"B" inniheldur þá stefin r,x,y og z, og vísar í stefið w. Útflutningarnir r,x og y eru sama stef.

5.2 Innflutningur

Þegar einingaraðgerðin "innflutningur" er framkvæmd tengjast innflutningar og útflutningar á nákvæmlega sama hátt og í samsetningu. Munurinn á þessum aðgerðum er sá að nöfn þau sem einingin A*B inniheldur eru nákvæmlega þau sömu og einingin A inniheldur. Mynd 5.3 lýsir þessu.

5.3 Samsetning

Lítum nú á einingaraðgerðina "samsetning". Mynd 5.4 lýsir tengingum þeim sem framkvæmdar eru þegar unnið er úr samsetningunni A : B. Einingin A (efsta einingin til vinstri á teikningunni) inniheldur x,y og z, og vísar í p og w. Einingin B inniheldur q,p,s og w, og vísar í r,w og t. Samsetta einingin A:B (neðst á myndinni) inniheldur allt það sem annað hvort A eða B innihalda. Í einingunni A:B er búið að tengja tilvísanir úr A við innihald B. Örvarnar milli t-anna, p-anna og w-anna í vinstri hluta teikningarinnar sýna þessar tengingar.

5.4 Hliðsetning

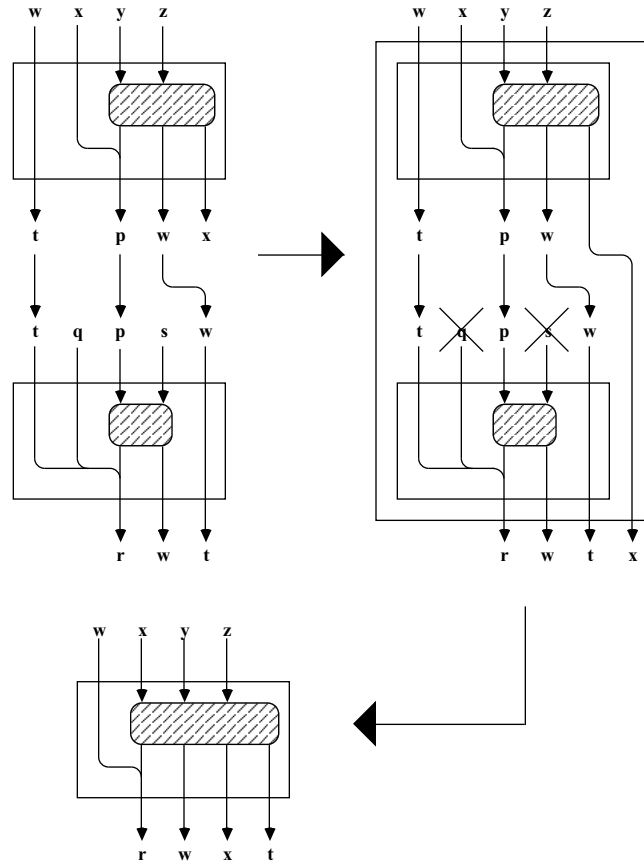
Mynd 5.5 lýsir hliðsetningu tveggja eininga, A og B. Einingin A (lengst til vinstri á myndinni) inniheldur x,y og z, og vísar í r og w. Einingin B inniheldur p, q, r og s og vísar í s, t og w. Einingin A+B fæst með því að skella einingunum saman án þess að tengja nokkuð.

5.5 Ítrunarhliðsetning

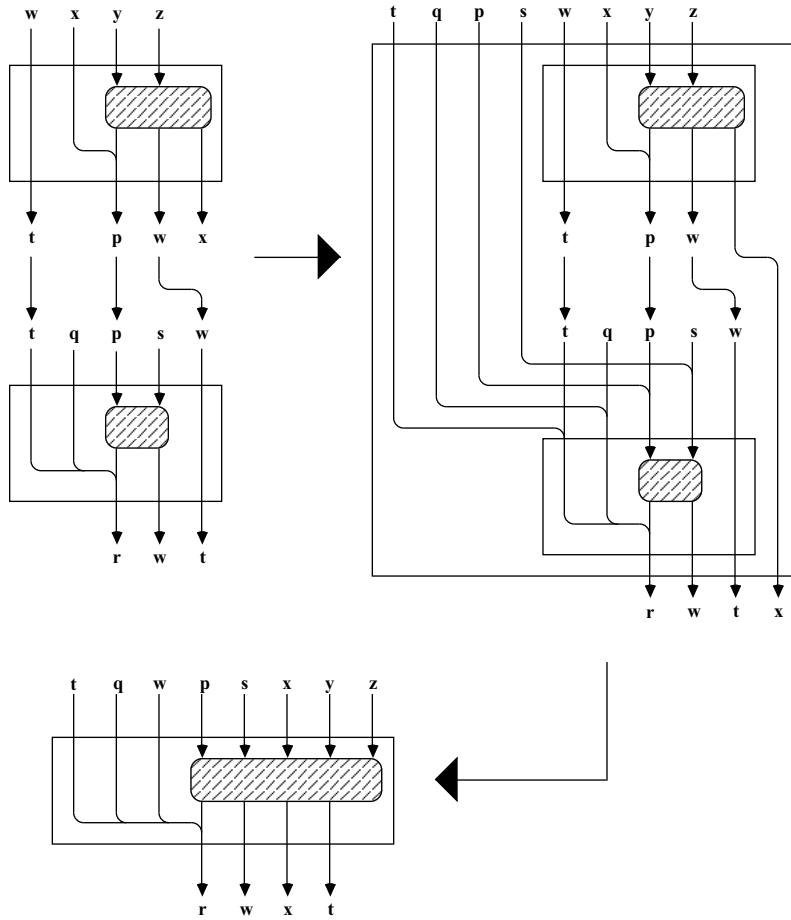
Sú einingaraðgerð sem sennilega er auðveldust í notkun þeim sem forritað hafa í öðrum forritunarmálum sem ekki bjóða einingaforritun er ítrunarhliðsetning. Hún er táknuð með aðgerðarstafnum "&" og er skilgreind á grundvelli ítrunar og hliðsetningar á eftirfarandi hátt fyrir einingar A og B:

$$A \& B = ! (A + B)$$

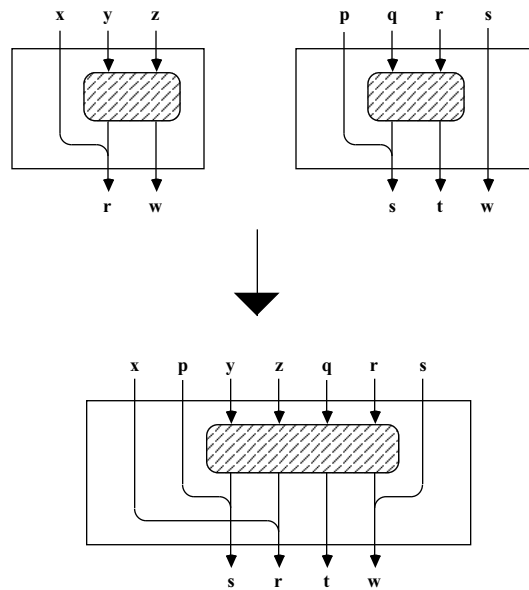
Þegar einingar A og B eru ítrunarhliðsettar þá er því allt tengt sem unnt er að tengja, bæði innan hvernar einingar fyrir sig og milli eininganna. Ef einhver æskir þess að nota Fjölni á svipaðan hátt og forritunarmál,



Mynd 5.3: Innflutningur



Mynd 5.4: Samsetning



Mynd 5.5: Hliðsetning

eins og t.d. C, sem ekki gefa kost á flóknum tengingum milli eininga þá getur hann skrifað sín forrit á eftirfarandi sniði:

```
"FORRIT" < aðal
{
aðal ->
    stef(;)
    stofn
    ...
    stofnlok
...
}
&
"GRUNNUR"
;
```

Þegar skrá sem inniheldur texta eins og að ofan er þýdd er smíðuð ein keyrsluhæf skrá, FORRIT.EXE. Forritið er smíðað með því að tengja saman textaeininguna og eininguna GRUNNUR. Þessi aðferð gengur að sjálfsögðu ekki ef einingarnar innihalda stef með sama nafni. Þá verður að nota aðrar einingaaðgerðir til að tengja.

Til þess að losna við að þýða allt aftur þegar breytingar eru gerðar má skipta forritinu að ofan upp í einingar. Þá má hafa nokkrar skrár á eftirfarandi sniði:

```
"EININGi" =
{
...
}
;
```

Þegar skrá sem inniheldur texta eins og að ofan er þýdd er smíðuð ein einingarskrá, EININGi.EIN. Loks má þýða eina skrá á eftirfarandi sniði til að tengja allt saman og smíða forritið:

```
"FORRIT" < aðal
{
aðal ->
    stef(;)
    stofn
    ...
    stofnlok
}
&
"EINING1"
&
...
&
"EININGn"
&
"GRUNNUR"
;
```

Þegar þessi skrá er þýdd eru allar einingarnar tengdar saman og keyrsluskráin FORRIT.EXE er skrifuð.

Kaflí 6

Veitingar

Forritstexti í Fjölni er runa af veitingum. Hver veiting smíðar eina einingu eða eitt forrit. Veitingar eru því tvenns konar, forritsveitingar og einingaveitingar. Forritsveitingar eru á eftirfarandi sniði:

```
nafn_keyrsluskrár < nafn_aðalstefs eining ;
```

Þegar Fjölnisþýðandinn vinnur úr forritsveitingu byrjar hann á að vinna úr einingunni. Einingin verður að innihalda stef sem heitir `nafn_aðalstefs` og það stef má ekki taka nein viðföng. Einingin má ekki vísa í nein stef eða breytur. Þá er skrifuð skrá á disk sem heitir `nafn_keyrsluskrár.EXE`. Sú skrá inniheldur keyrsluhæft forrit. Keyrsla þessa forrits felst í því að keyra stefið `nafn_aðalstefs` einu sinni.

Einingaveitingar eru af öðru hvoru eftirfarandi sniða:

```
nafn_einingarskrár = eining ;
```

```
nafn_einingarbreytu = eining ;
```

Í fyrra tilfellinu er eining smíðuð og geymd í skrá á disk. Í því seinna er eining smíðuð og geymd í svokallaðri einingarbreytu sem búin er til í fyrsta skiptið sem vísað er í hana. Einingarbreyta þessi er aðeins til staðar meðan þýðing þessi fer fram, að henni lokinni er einingin glötuð. Á meðan þessari þýðingu stendur má vísa í þessa einingarbreytu með nafni, eins oft og verða vill, til þess að nota einingu þá sem smíðuð var.

Kafli 7

Notkun Fjölnis

Í Fjölni eru þýðandi og tengir í sama forriti. Til að þýða forritstexta í skráum x1.fjo, x2.fjo, o.s.frv. er notuð skipunarunan

```
C:\>FJÖLNIR x1  
C:\>FJÖLNIR x2
```

...

eða skipunin

```
C:\>FJÖLNIR x1 x2 ...
```

Skrár sem innihalda forritstexta ættu því að hafa viðurnefnið “.FJO”. Skrár sem innihalda þýddar einingar hafa viðurnefnið “.EIN”. Í þeim eru geymdar upplýsingar sem gerir Fjölni kleyft að villuprófa tengingar þær sem gerðar eru og skrifa villuboð ef rangar tengingar eru gerðar (til dæmis er rangt að tengja kall í stef sem taka skal eitt viðfang við stef sem tekur tvö viðföng). Þýðandinn var að öllu leyti skrifaður í Fjölni og smalamáli. Það er því óhætt að fullyrða að nota megi Fjölni til að leysa verulega stór verkefni.

Þegar þýðandinn er keyrður skrifar hann á skjáinn vísbendingar um framvindu þýðingarinnar. Ef ekki er óskað eftir slíkum útskriftum, eða óskað er nánari upplýsinga, má gefa þýðandanum fyrirmæli um slíkt með rofum. Einnig er rofi sem stýrir notkun minnis í hlaða. Þrjú rofar eru tiltækir notendum Fjölnis:

E Veldur því að Fjölnir keyrir án þess að skrifa á skjá.

T Veldur því að Fjölnir birtir vísbendingar um tengingar þær sem gerðar eru þegar einingaraðgerðir eru framkvæmdar.

Mx Stýrir því hversu mikið minni er notað í hlaða í EXE skrá sem Fjölnir skrifar. Stafurinn x skal vera tölustafur í sextándakerfi, 0..9 eða A..F. Stafirnir A..F standa fyrir tölurnar 10..15 eins og kunnugt er. Stærð minnissvæðisins sem notað er undir hlaða er ákvarðað af formúlunni $4096 \cdot (x+1)$. Ef enginn M rofi er notaður jafngildir það rofa MF, þ.e.a.s. notuð verða 64K undir hlaða. Yfirleitt þarf ekki meira en 4K (4096 bæti) í hlaða, en ef forritið notar djúpa endurkvæmni þarf meiri hlaða.

Rofana má ýmist gefa á skipanalínu þegar Fjölnir er keyrður eða setja má rofana í umhverfisbreytu í MS-DOS. Til að gera þýðandann hljóðlátan er t.d. notuð skipun á forminu

```
FJÖLNIR/E x1 x2 ...
```

Svipaða aðferð má nota til að láta þýðandann verða hljóðlátan að staðaldri. Til þess nægir að gefa MS-DOS skipunina

```
SET FJROFAR=E
```

Skilgreina má leitarlóðir fyrir tilvísanir í einingaskrár í texta þeim sem þýddur er. Til þess er notuð MS-DOS skipun á forminu

```
SET FJLEIT=<leið>1;...;<leið>n
```

Skipun þessi veldur því að leitað er að einingaskrár sem ekki finnast, í efnisskrám (e. *directory*) <leið>1 til <leið>n. T.d. ef einingar eru ávallt geymdar í efnisskránni C:\EIN þá má gefa skipunina:

```
SET FJLEIT=C:\EIN
```

og má þá vísa í þessar einingar án þess að þurfa að skeyta C:\\EIN\\ framán á nöfn þeirra.

Kafli 8

Málfræði Fjölnis

8.1 Athugasemdir í forritstexta

Í forritslínu má setja athugasemd með því að hefja hana á ; ; . Þýðandinn mun þá hlaupa yfir afganginn af þeirri línu, frá ; ; til línuloka. Til dæmis er línan

```
    meðan x eða y lykkja    ; ; finnum næsta prímtölupátt
```

jafngild línunni

```
    meðan x eða y lykkja
```

8.2 Frumeiningar málsins

Forrit í Fjölni eru byggð úr eftirtöldum frumeiningum:

- Lykilorð og nöfn
- aðgerðir og frátekin tákni
- Lesfastar

Lykilorð eru eftirfarandi:

- annars
- annarsef
- breyta
- eða
- eflok
- ef
- ekki
- fyrir
- innflutt
- kostur

- lykkja
- lykkjulok
- meðan
- og
- skila
- staðvær
- stef
- stofn
- stofnlok
- úr
- út
- val
- vallok
- þá

Í Fjölni er gerður greinarmunur á hástöfum og lágstöfum (stórum og litlum stöfum) og lykilordin verður að rita með lágstöfum.

Frátekin tákni eru:

() [] { } -> := , ; ..

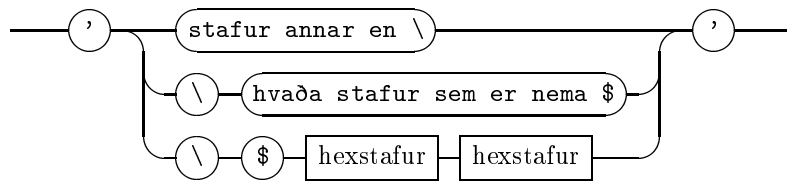
Á milli tákna í Fjölnisforriti mega vera eins margir bilstafir, dálkstafir (e. emph tab) og línubil og verða vill.

Lesfastar eru af þrem gerðum:

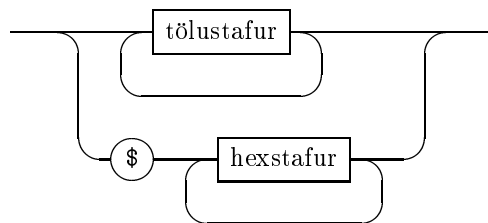
- Fjöldatölu- og heiltölufastar
- Fleytitölufastar
- Strengfastar

Fjöldatölufastar geta haft gildi á bilinu 0 til 65535. Heiltölufastar geta haft gildi á bilinu -32768 til 32767. Fjöldatölu- og heiltölufastar eru ferns konar:

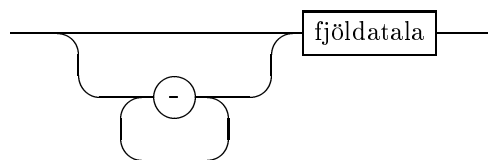
- Fjöldatölufastar í tugakerfi
- Heiltölufastar í tugakerfi
- Fjöldatölufastar í sextándakerfi
- Staffastar

staffasti

Mynd 8.1: Staffasti

fjöldatala

Mynd 8.2: Fjöldatala

heiltala

Mynd 8.3: Heiltala

Allir þessir fastar eru að vissu leyti jafngildir þar eð í keyrslu er ekki unnt að gera greinarmun á þeim í Fjölnisforriti. T.d. er heiltalan -1 sama og fjöldatalan 65535, og staffastar eru geymdir sem fjöldatalan sem er sæti viðkomandi stafs í stafrófinu.

Myndir 8.1, 8.2 og 8.3 lýsa málfræði frumtákna Fjölnis. Málrít þessi hafa þann eiginleika að öll lögleg Fjölnistákn má smíða með því að fylgja línunum eftir, svipað og járnbrautateinum. Farið er í gegnum tvenns konar kassa, ferhyrninga og ávala kassa. Ávölu kassarnir standa fyrir stafi sem koma óbreyttir í viðkomandi forrit. Ferhyrningarnir innihalda nöfn annara máleininga sem í forritinu geta verið, og eru skilgreindir með eigin málríti.

Tölustafir eru '0' til '9'. Hexstafir eru '0' til '9', 'A' til 'F' og 'a' til 'f'. Staffastar eru notaðir til að rita einn ASCII staf. Rita má stýristafi sem staffasta. Þeir eru ritaðir á svipaðan hátt og í strengjum eins og lýst er annars staðar í þessum kafla.

Dæmi um löglega heiltölu-, fjöldatölu- og staffasta:

```
1
12345
-10
$AB12
'a'
'\\'
',''
'\$07'
'\b'
''''
```

Dæmi um ólöglega heiltölu-, fjöldatölu- og staffasta:

```
$G123
'ab'
-40000
123456
"a"
',''
''''
```

Málrít fleytitölufasta er sýnt á mynd 8.4.

Dæmi um löglega fleytitölufasta:

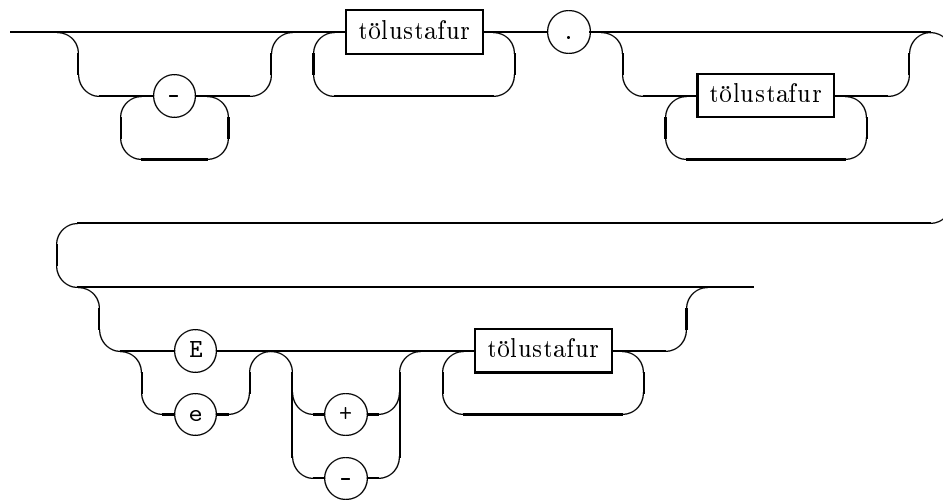
```
1.0
1.E-10
12.34
-1.
--12.34E123
```

Dæmi um ólöglega fleytitölufasta:

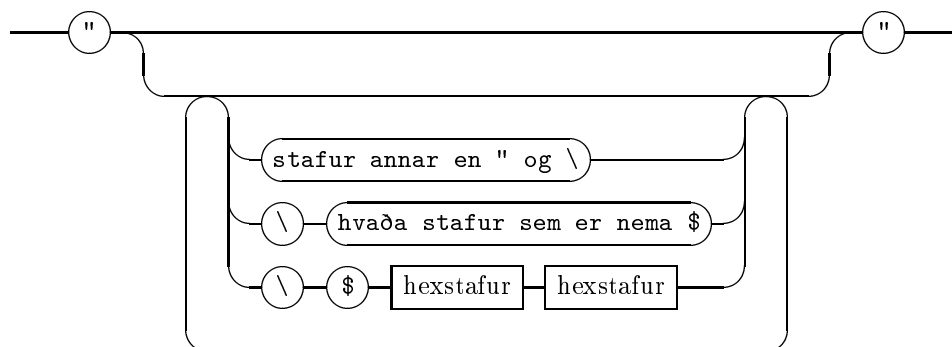
```
.0
1.2E
```

Málrít strengfasta er sýnt á mynd 8.5.

Þegar \ kemur fyrir inni í streng er næsti stafur á eftir oftast tekinn bókstaflega inn í strenginn. Þessa aðferð má nota til að skrifa strengi sem innihalda " eða \. Þá er " táknnað með \" og \ er táknnað með \\. Einnig er \ notað til að gera forritaranum kleift að setja algenga ASCII stýristafi inn í strengi. Tafla 8.1 sýnir táknun þeirra stýristafa sem unnt er að tákna með stafapari þar sem fyrri stafurinn er \:

fleytitala

Mynd 8.4: Fleytitala

strengur

Mynd 8.5: Strengfastar

Stafapar	Stýristafur
\c eða \C	ASCII 13, CR
\b eða \B	ASCII 7, BEL
\e eða \E	ASCII 27, ESC
\f eða \F	ASCII 12, FF
\n eða \N	ASCII 10, LF
\t eða \T	ASCII 9, HT
\0	ASCII 0
\1	ASCII 1
\2	ASCII 2
\3	ASCII 3
\4	ASCII 4
\5	ASCII 5
\6	ASCII 6
\7	ASCII 7
\8	ASCII 8
\9	ASCII 9

Tafla 8.1: Stýristafir

Einnig er unnt að rita hvaða ASCII staf sem er í strenginn með því að skrifa hann á sniðinu `\$xx` þar sem `xx` er tveggja stafa tala í sextándakerfi. T.d. eru `\b` og `\$07` jafngildir stafir, sem og `0` og `\$30`.

Dæmi um löglega strengfasta:

```
"abc"
```

```
"strengur með \" inni í"
```

```
"\\ heitir bakstrik"
```

```
" , "
```

```
"$"
```

```
"\ $FF"
```

Dæmi um ólöglega strengfasta:

```
'abc'
```

```
" " "
```

```
"\"
```

```
"\ $"
```

```
"\ $FG"
```

Hin þriðja gerð lesfasta sem tákna má í Fjölni er tómagildið, sem táknað er með `[]`. Tómagildið stendur fyrir sérstakt gildi í Fjölni, sem er ójafnt öllum öðrum gildum.

Nöfn í Fjölni eru tvenns konar:

- Stefnöfn og breytunöfn
- Aðgerðanöfn

Mállýsing aðgerðanafna er í mynd 8.6:

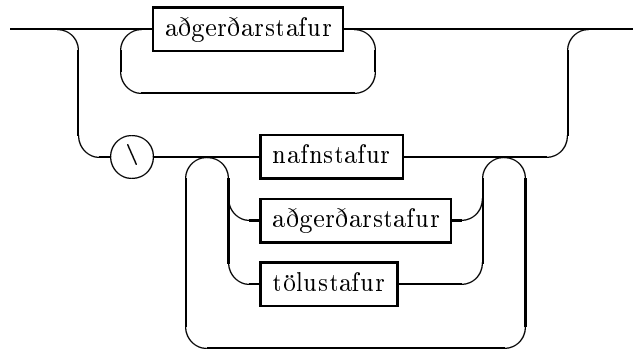
Aðgerðarstafir, sem nota má til að mynda aðgerðir, eru eftirfarandi:

```
+ - * / % ! & =
```

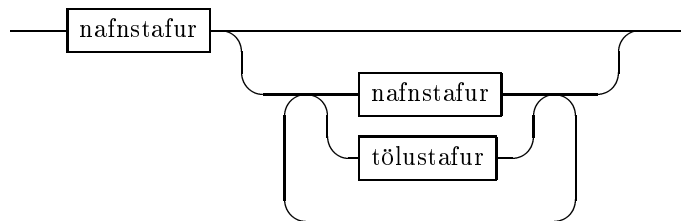
```
? <> | : ^ @
```

Mállýsing stefnafna og breytunafna er í mynd 8.7.

Nafnstafir eru allir stafir aðrir en eftirfarandi:

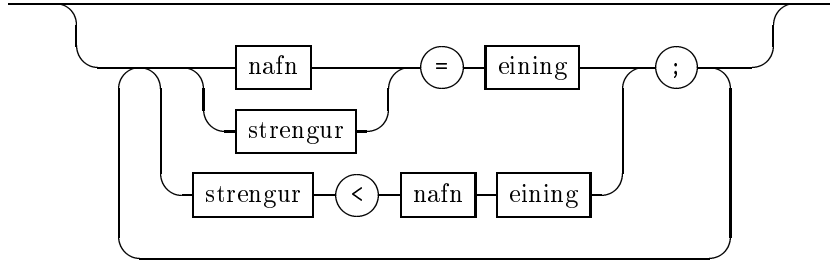
aðgerð

Mynd 8.6: Aðgerðanöfn

nafn

Mynd 8.7: Nöfn

forrit



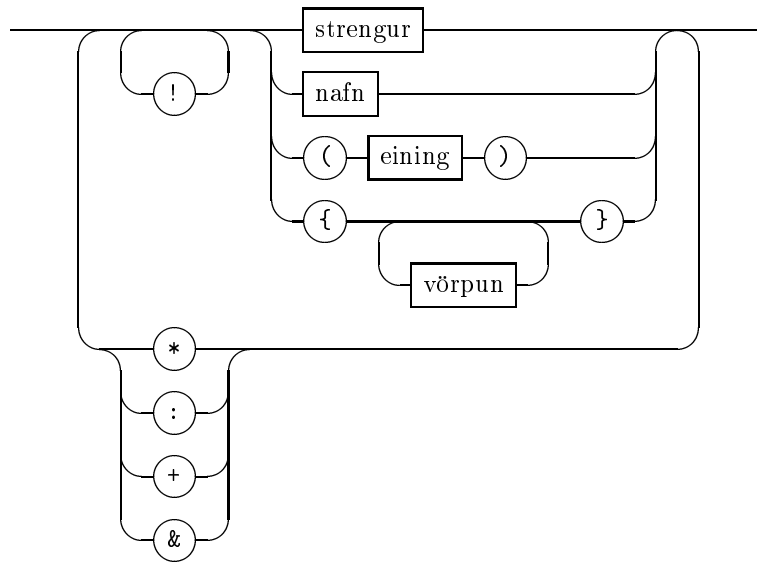
Mynd 8.8: Forrit

```
bil ( ) [ ] { } , ; .
aðgerðarstafir
ASCII stafir með raðnúmer frá 0 til 31
```

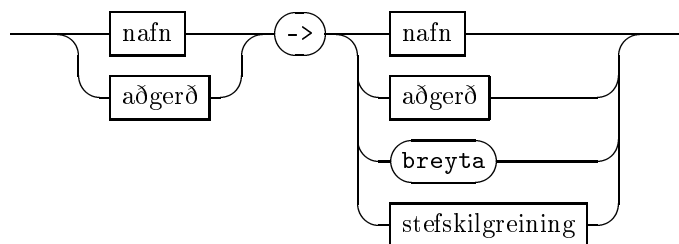
Allir íslenskir bókstafir eru því nafnstafir.

8.3 Málrit

Myndir 8.8, . . . , 8.21 lýsa heildarbyggingu Fjölnisforrita. Málrit þessi skal lesa á sama hátt og málritin sem lýsa framtáknum Fjölnis. Lesa má í gegn um öll lögleg Fjölnisforrit með því að byrja þar sem stendur “forrit” í myndinni hér að neðan og renna með járnbrautateinunum í gegn um myndina og allar þær undirmyndir sem þörf krefur.

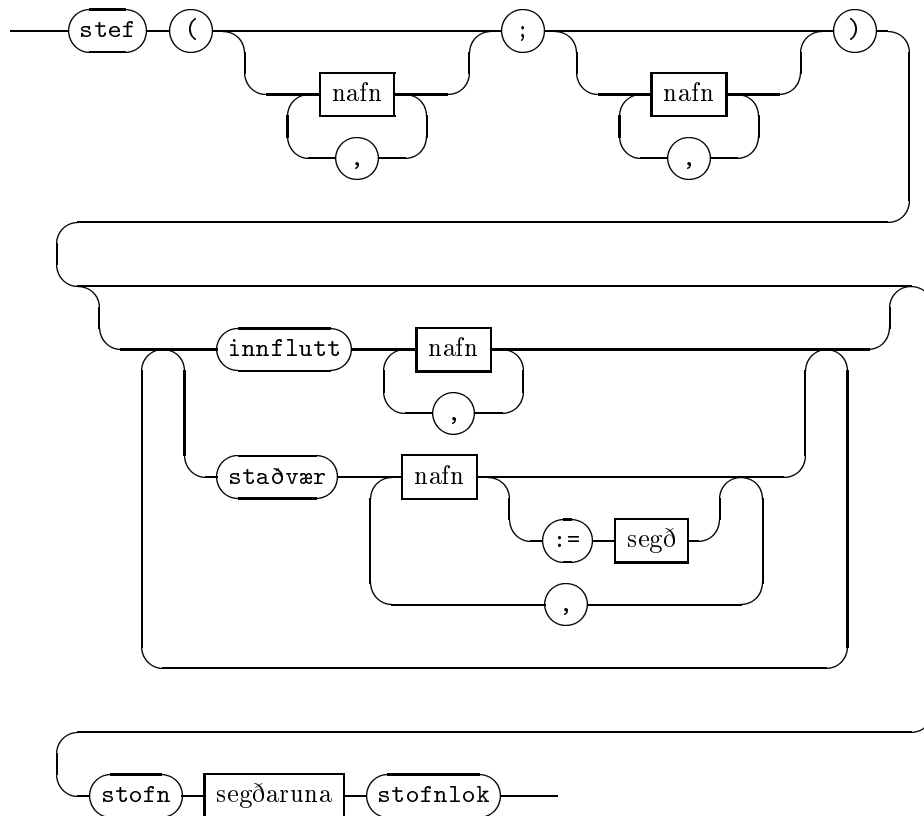
eining

Mynd 8.9: Eining

vörpun

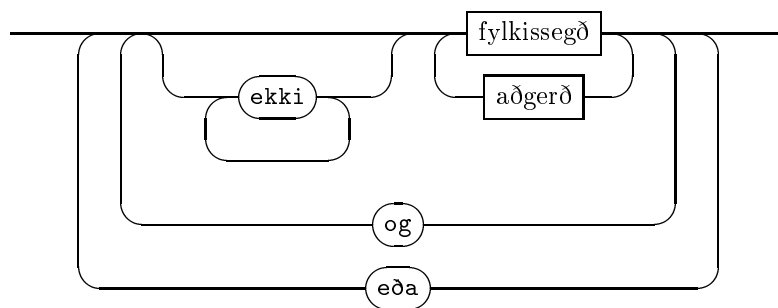
Mynd 8.10: Vörpun

stefskilgreining

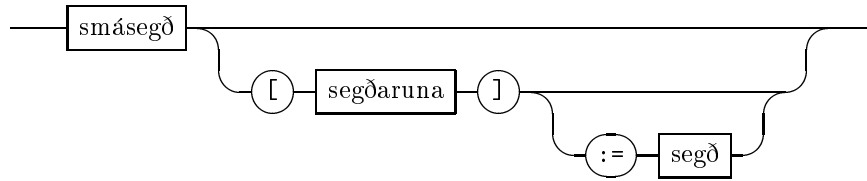


Mynd 8.11: Stefskilgreining

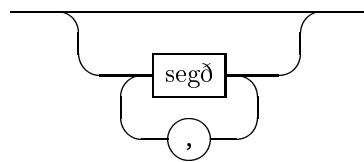
segð



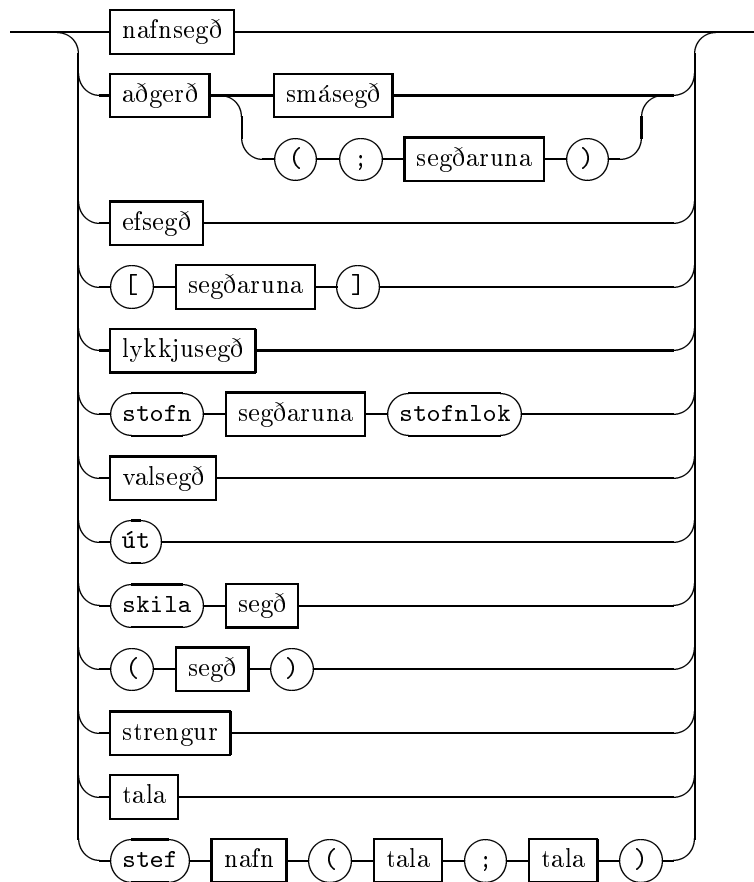
Mynd 8.12: Segð

fylkissegð

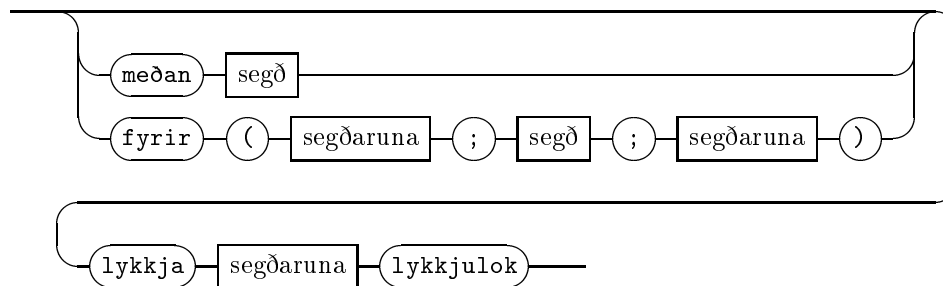
Mynd 8.13: Fylkissegð

segðaruna

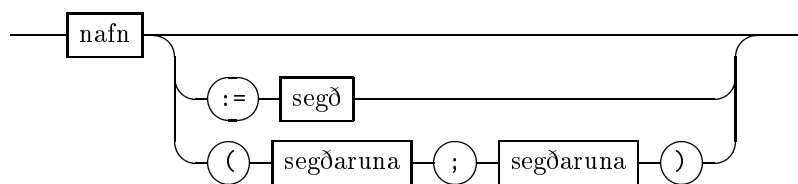
Mynd 8.14: Segðaruna

smásegð

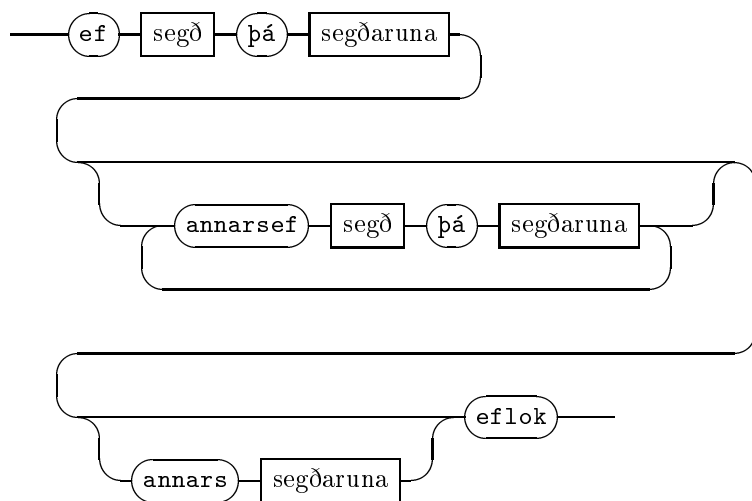
Mynd 8.15: Smásegð

lykkjusegð

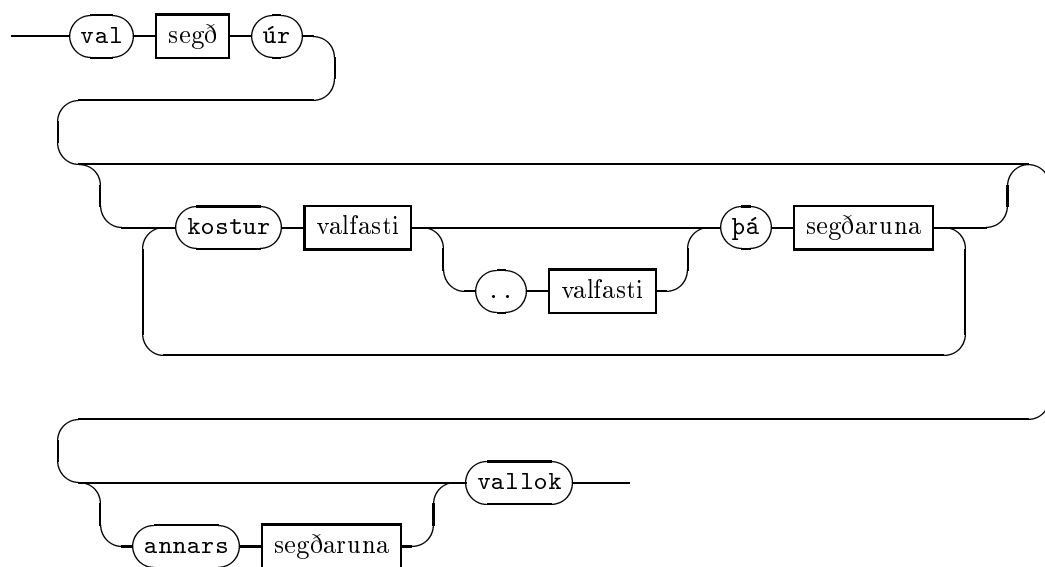
Mynd 8.16: Lykkusegðir

nafnsegð

Mynd 8.17: Nafnsegðir

efsegð

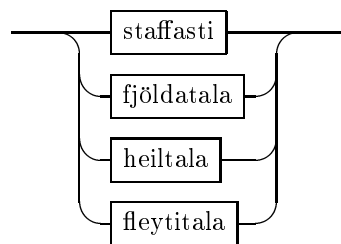
Mynd 8.18: Efsegð

valsegð

Mynd 8.19: Valsegð

valfasti

Mynd 8.20: Valfasti

tala

Mynd 8.21: Tala

Kaflí 9

Eftirmáli

Forritunarmálið Fjölnir er sérstaklega hannað með einingaforritun í huga. Það er byggt á nýjum hugmyndum í einingaforritun, sem fyrst var lýst í heimildum [2,3,4]. Þessar hugmyndir byggjast aftur á grundvallarhugmyndum í stærðfræði óendanlegra trjáa, sem lýst er í heimild [1]. Hugmyndir þessar voru leiðandi ljós í hönnun forritunarmálsins, og reyndist vera ómetanlegt gagn af þeim við að skera úr um vafaatriði í hönnun málsins. Það er þó alls ekki nauðsynlegt fyrir notendur forritunarmálsins að vera kunnugir þessum fræðum.

Kafi 10

Heimildir

- 1 Bruno Courcelle, Fundamental Properties of Infinite Trees, Theoretical Computer Science 25 (1983) 95-169.
- 2 Snorri Agnarsson, Packages as Substitutions, doktorsritgerð, Rensselaer Polytechnic Institute, júní 1985.
- 3 Snorri Agnarsson og M.S. Krishnamoorthy, Towards a Theory of Packages, Proceedings of the ACM SIGPLAN-85 symposium on language issues in programming environments. SIGPLAN Notices, Vol. 20, No. 7.
- 4 Snorri Agnarsson, M.S. Krishnamoorthy og B. David Saunders, An algebraic theory of packages. Proceedings of EUROCAL-85. Lecture Notes in Computer Science, Springer-Verlag 1985.

Hluti II

Einingin GRUNNUR

Einingin GRUNNUR er samsett úr mörgum öðrum einingum. Hverri þessara eininga er lýst í undirkafli í þessum kafla. Grunnurinn er ansi stór eining og inniheldur fleiri stef en flest forrit nota. Það er því oft ástæða til að nota minni grunn en þann sem fylgir Fjölni. Notandi Fjölnis getur smíðað sinn eigin grunn ef hann vill, eða notað beint þær einingar sem grunnurinn er samsettur úr. Í myndinni hér að neðan er sýnt hvernig einingarnar í grunninum tengjast saman, þ.e.a.s. hvernig einingarnar kalla á stef úr hver annarri. Þar má t.d. sjá að ef nota þarf eininguna INN þá þarf einnig að flytja inn einingarnar UT, SNUA, LESALINU og DOSSKRIU. Auk þess þarf nærri alltaf að flytja inn eininguna KJARNI, eða einhverja einingu sem inniheldur þá einingu, t.d. GRUNNUR, þar eð langflestar einingar vísa í einhver kjarnastef. Við köllum eininguna KJARNI kjarnann og eininguna GRUNNUR grunninn og stef í kjarnanum köllum við kjarnastef, og stef í grunninum grunnstef.

Kaflí 11

Einingin AFRITA

Einingin AFRITA inniheldur aðgerð til að taka afrit af gildi.

afrita

Notkun: `gildi1 := afrita(;gildi2)`

Fyrir: `gildi2` er eitthvert gildi, getur verið nánast hvað sem er. Aðeins er þess krafist að ekki séu tilvísanir í hring í gildinu. Slíkar tilvísanir er aðeins unnt að smíða með því að nota stef eins og haussetja, halasetja eða hlunksetja og slík notkun er ekki algeng. Ef um tilvísun í hring er að ræða mun það valda óendanlegri lykkju eða óendanlegri endurkvæmni.

Eftir: `gildi1` fær gildi sem er nákvæmt afrit af `gildi2`. Í `gildi1` eru allir hlunkar (þ.m.t. þör) og strengir nýjar útgáfur af samsvarandi streng eða hlunk í `gildi2`.

Kafli 12

Einingin DOSDISK

disklausK

Notkun: `i := disklausK(;drif)`

Fyrir: `drif` er heiltala eða fjöldatala sem segir til um hvaða diskdrif er að ræða. `drif=0` er sjálfgefið diskdrif. `drif=1` er diskdrif a. `drif=2` er diskdrif b: o.s.frv.

Eftir: `i` inniheldur fjöldatölu sem segir til um hversu mörg kílóþæti eru laus á því diskdrifi sem gefið var upp í `drif`. Ef fjöldatalan sem gefin er upp í `drif` lýsir ekki löglegu diskdrifi á viðkomandi tölvu, inniheldur `i` gildið ósatt eftir kallið. Eins inniheldur `i` gildið ósatt ef einhverjar ólöglegar eða skemmdar diskeiningar eru á viðkomandi diskdrifi.

diskstærð

Notkun: `i := diskstærð(;drif)`

Fyrir: `drif` er heiltala eða fjöldatala sem segir til um hvaða diskdrif er að ræða. `drif=0` er sjálfgefið diskdrif. `drif=1` er diskdrif a, `drif=2` er diskdrif b: o.s.frv.

Eftir: `i` inniheldur fjöldatölu sem segir til um hversu mörg kílóþæti diskdrifið sem gefið var upp í `drif` rúmar í heild.

Ef fjöldatalan sem gefin er upp í `drif` lýsir ekki löglegu diskdrifi á viðkomandi tölvu, inniheldur `i` gildið ósatt eftir kallið. Eins inniheldur `i` gildið ósatt ef einhverjar ólöglegar eða skemmdar diskeiningar eru á viðkomandi diskdrifi.

skiptaumsafn

Notkun: `ok := skiptaumsafn(;safnsemertil)`

Fyrir: `safnsemertil` er strengur sem inniheldur löglega efnisskrárleið fyrir DOS. Dæmi um löglegar leiðir eru `"\\pakkar"`, `"a:pakkar"`, `"a:\\pakkar"`, `"a:\\"`, `"\\"` o.s.frv.

Dæmi um ólöglegar leiðir eru `"a:pakkar\\"`, `"a:"` o.s.frv.

Eftir: Það safn er sjálfgefið, sem gefið er í `safnsemertil`, á sjálfgefnu diskdrifi. Ef `safnsemertil` inniheldur nafn á diskdrifi (er t.d. `"a:\\pakkar"`), er skipt yfir á uppgefið safn á því diskdrifi en ekki er skipt um sjálfgefið diskdrif.

Eftir kallið fær `ok` gildið satt ef uppgefin efnisskrárleið er lögleg og er til á viðkomandi diskdrifi (og skipt hefur verið yfir á `safnsemertil`) en ósatt ef efnisskrárleiðin er ólögleg eða ekki til á viðkomandi diskdrifi. Sálfgefið safn er þá það sama á viðkomandi diskdrifi og var fyrir.

sjálfgefiðsafn

Notkun: `safn := sjálfgefiðsafn(;drif)`

Fyrir: `drif` er heiltala eða fjöldatala sem segir til um hvaða diskadrif er að ræða. `drif=0` er sjálfgefið diskadrif. `drif=1` er diskadrif a. `drif=2` er diskadrif b: o.s.frv.

Eftir: Ef `drif` er löglegt diskadrif fyrir viðkomandi tölvu, fær `safn` gildi strengs sem inniheldur efnisskrárleið að því safni sem sjálfgefið er á diskadrifi `drif`. Sá strengur inniheldur ekki nafn diskadrifsins þ.a. sé sjálfgefið diskadrif "a:" og sjálfgefið safn á því drifi "pakkar" þá gæfi kallið `sjálfgefiðsafn(;1)` niðurstöðuna "pakkar"(en ekki "a: pakkar"). Ef `drif` er ólöglegt diskadrif eða ef einhver önnur villa kemur upp, fær `safn` gildið ósatt.

búatilsafn

Notkun: `ok := búatilsafn(;nýtttsafn)`

Fyrir: `nýtttsafn` er strengur sem inniheldur löglega efnisskrárleið fyrir DOS.

Dæmi um löglegar leiðir væru "
pakkar", "a:pakkar", "a:
pakkar" o.s.frv.

Dæmi um ólöglegar leiðir væru "a:pakkar
, "a:", "a:
, "
"o.s.frv.

Eftir: `Búið` er til það safn sem gefið er í `nýtttsafn` á sjálfgefnu diskadrifi. Ef `nýtttsafn` inniheldur diskadrifsnafn (er t.d. "a: pakkar"), er `búið` til uppgafið safn á því diskadrifi.

Eftir kallið fær `ok` gildið satt ef uppgefin efnisskrárleið er lögleg og safnið er ekki þegar til á viðkomandi diskadrifi (og `búið` hefur verið til safnið `nýtttsafn`) en ósatt ef efnisskrárleiðin er ólögleg eða þegar til á viðkomandi diskadrifi (og ekkert safn hefur verið `búið` til).

eyðasafni

Notkun: `ok := eyðasafni(;safnsemertil)`

Fyrir: `nýtttsafn` er strengur sem inniheldur löglega efnisskrárleið fyrir DOS.

Dæmi um löglegar leiðir væru "
pakkar", "a:pakkar", "a:
pakkar" o.s.frv.

Dæmi um ólöglegar leiðir væru "a:pakkar
, "a:", "a:
, "
"o.s.frv.

Eftir: það safn sem gefið er í `safnsemertil` er fjarlægð á viðkomandi diskadrifi ef `safnsemertil` inniheldur nafn á diskadrifi, annars á sjálfgefnu diskadrifi.

Eftir kallið fær `ok` gildið satt ef uppgefin efnisskrárleið er lögleg og safnið er til staðar á viðkomandi diskadrifi (og `búið` er að eyða safninu `nýtttsafn`) en ósatt ef efnisskrárleiðin er ólögleg eða er ekki til á viðkomandi diskadrifi (og ekkert hefur verið gert).

Athugað skal að ef einhver forrit eða söfn eru til staðar á því safni sem verið er að eyða, fær ok gildið ósatt og ekkert er gert, þar sem ekki er hægt að eyða safni með einhverju á.

erskrátil

Notkun: `ok := erskrátil(;nafn)`

Fyrir: nafn er strengur sem inniheldur löglegt nafn skrár (með efnisskrárleið ef vill).

Eftir: Ef nafn er löglegt nafn á skrá, sú skrá er til og engin villa kemur upp, fær ok gildið satt. Ef skráin er ekki til eða ef einhver villa kemur upp, fær ok gildið ósatt.

opnaskrá

Notkun: `handfang := opnaskrá(;nafn, aðgangur)`

Fyrir: nafn er strengur sem inniheldur löglegt nafn þeirrar skrár sem opna á (með efnisskrárleið ef vill). aðgangur er stafur ('l', 'L', 's', 'S', 'u', 'U', 'v' eða 'V') sem segir til um fyrir hvaða aðgang skráin er opnuð.

Eftir: Ef nafn er löglegt nafn á skrá miðað við aðgangur, er hún opnuð fyrir aðgang aðgangur og handfang fær gildi handfangs (fjöldatala) sem lýsir þessari nýopnuðu skrá, annars fær handfang gildið ósatt. Handfangið er síðan notað fyrir allar síðari aðgerðir á skrána.

nafn getur hvort sem er verið nafn skrár með efnisskrárleið, þá er viðkomandi skrá opnuð á því safni, eða að nafn er eingöngu nafn á skrá, þá er sú skrá opnuð á sjálfgefnu safni.

Les/skrif bendir er bendir sem fylgir hverri opinni skrá og segir til um við hvaða stað í skránni aðgerðirnar lestur og/eða skrift hefjast.

Niðurstaða kallsins á opnaskrá fer eftir því hvaða aðgangur er gefinn og hvort nafn er nafn á skrá sem er til eða ekki:

Ef aðgangur = 'l' eða 'L' er skráin nafn opnuð með lestraraðgangi og allar tilraunir til að skrifa í skrána mistakast. Les/skrif bendir er staðsettur í upphafi skrárinnar. Ef skráin er til og allt er í lagi, fær handfang gildi handfangs sem vísar í skrána. Ef skráin er ekki til, of margar skrár opnar eða einhver önnur villa kemur upp, fær handfang gildið ósatt og skráin hefur ekki verið opnuð.

Ef aðgangur = 's' eða 'S' er skráin nafn opnuð með skriftaraðgangi og allar tilraunir til að lesa úr skránni geta verið mjög varasamar. Les/skrif bendir er staðsettur í upphafi skrárinnar. Ef skráin nafn er þegar til, er henni í raun eytt út og hún síðan búin til aftur þ.e. þegar skrá, sem þegar er til, er opnuð með skriftaraðgangi er hún algerlega núllstillt. Ef skráin er ekki til, er hún búin til og opnuð á sama hátt og áður. Ef allt er í lagi, fær handfang gildi handfangs sem lýsir skránni. Ef einhver villa kemur upp fær handfang gildið ósatt og skráin getur hvort sem er hafa verið búin til eða ekki, en að öllum líkindum er ekki hægt að framkvæma neinar aðgerðir á hana. Kallið `opnaskrá(;nafn,'s')` hefur nákvæmlega sömu áhrif og `kallið búatilskrá(;nafn)`.

Ef aðgangur = 'u' eða 'U' er skráin nafn opnuð með uppfærsluaðgangi og er þá hvort sem er hægt að framkvæma lestur eða skrift. Les/skrif bendir er staðsettur í upphafi skrárinnar. Ef skráin nafn er til og allt er í lagi, er skráin opnuð fyrir uppfærsluaðgang og handfang fær gildi handfangs sem lýsir skránni. Ef skráin er ekki til eða einhver önnur villa kemur upp, fær handfang gildið ósatt og engin skrá hefur verið opnuð. Skrá sem ekki er til en reynt er að opna fyrir uppfærsluaðgang er því ekki búin til.

Ef aðgangur = 'v' eða 'V' er skráin nafn opnuð fyrir viðbótaraðgang og er þá einungis hægt að framkvæma skriftaraðgerð á hana. Les/skrif bendir er staðsettur aftast í skránni (aftan við hennar síðasta bæti). Allar skriftaraðgerðir aftast í skrá, sem opin er fyrir viðbótaraðgang, stækka skrána (bæta við hana). Ef skráin nafn er ekki til, er niðurstaðan nákvæmlega sú sama og að opna hana með skriftaraðgangi ('s') þ.e. skráin er búin til og les/skrif bendir staðsettur fremst í henni. Ef skráin nafn er til, er

hún opnuð og les/skrif bendir færður aftast í hana. Ef allt er í lagi fær handfang gildi handfangs sem lýsir skránni, annars fær handfang gildið ósatt.

búatilskrá

Notkun: `handfang := búatilskrá(;nafn)`

Fyrir: nafn er strengur sem inniheldur löglegt nafn á þeirri skrá sem búa á til. Sjá um opnaskrá.

Eftir: Ef nafn er löglegt nafn á skrá er hún opnuð fyrir skriftaraðgang og allar tilraunir til að lesa úr skránni geta verið mjög varasamar. Les/skrif bendir er staðsettur í upphafi skrárinnar. Ef skráin nafn er þegar til, er henni í raun eytt út og hún síðan búin til aftur þ.e. þegar skrá sem þegar er til er opnuð með skriftaraðgangi er hún algerlega núllstillt. Ef skráin er ekki til, er hún búin til og opnuð á sama hátt og áður. Ef allt er í lagi, fær handfang gildi handfangs (heiltala eða fjöldatala) sem lýsir skránni. Ef einhver villa kemur upp fær handfang gildið ósatt og skráin getur hvort sem er hafa verið búin til eða ekki, en að öllum líkindum er ekki hægt að framkvæma neinar aðgerðir á hana. Kallið `opnaskrá(;nafn,'s')` hefur nákvæmlega sömu áhrif og kallið `búatilskrá(;nafn)`.

lokaskrá

Notkun: `ok := lokaskrá(;handfang)`

Fyrir: handfang er fjöldatala eða heiltala sem er gildi handfangs sem áður hefur fengist úr kalli á opnaskrá eða búatilskrá.

Eftir: Skránni sem lýst er með handfanginu handfang er lokað þ.a. allar les/skrif/loka aðgerðir á handfang eftir það, gefa villu. Ef allt er í lagi og DOS tekst að loka handfang eðlilega, fær ok gildið satt en ósatt ef einhver villa kemur upp. Þær villur sem algengast er að komi upp eru að handfang er talið lýsa einhverju handfangi sem þegar er búid að loka eða að það hefur aldrei verið búid til. Varasamt getur verið að loka slíkum handföngum þar sem DOS hefur 5 frátekin handföng (0..4) sem nauðsynleg eru flestum forritum í keyrslu. Þau eru 0=aðalinntak (standard input), 1=aðalúttak (standard output), 2=Aðalvilluúttak (standard error), 3=ytra inntak/úttak (auxiliary) og 4=aðalprentaraúttak (standard printer). Forrit sem t.d. lokar aðalinntaki (handfangi 2), fær sennilega villu næst þegar það reynir að lesa af aðalinntaki (lyklaborði).

eyðaskrá

Notkun: `ok := eyðaskrá(;nafn)`

Fyrir: nafn er strengur sem inniheldur löglegt nafn (með efnisskrárleið ef vill) á skránni sem eyða á.

Eftir: Skránni sem ber heitið nafn er eytt af diskum svo framarlega sem hún er þar til staðar. Ef engin villa kemur upp og skráin er til staðar á viðkomandi diskadrifi, er henni eytt og ok fær gildið satt, annars fær ok gildið ósatt. Varasamt og í raun ólöglegt er að eyða skrá sem enn er opin (hefur handfang sem lýsir henni) og óútreiknanlegt er hver áhrif það kann að hafa. Í því tilfalli kemur þó í fæstum tilfellum upp villa við kall í eyðaskrá, en síðari les/skrif/loka aðgerðir geta gefið óskiljanlegar villur.

breytanafni

Notkun: `ok := breytanafni(;gamaltnafn, nýttnafn)`

Fyrir: `gamaltnafn` og `nýttnafn` eru hvoru tveggja strengir sem innihalda lögleg nöfn á skrá. `gamaltnafn` er nafn skrár sem þegar er til og `nýttnafn` er nafn á skrá sem ekki er til.

Eftir: Nafni skrárinnar gamaltnafn er breytt yfir í nýtt nafn og ok fær gildið satt ef gamaltnafn er löglegt nafn á skrá sem þegar er til og nýtt nafn er löglegt nafn á skrá sem ekki er til og allt annað er í lagi. Ef eitthvað er ekki í lagi, fær ok gildið ósatt. breytanafni er hægt að nota til að færa skrár milli safna á sama diskadrifi. Ef t.d. ætti að færa skrána "pakki" á sjálfgefnu safni (sem ekki er "pakkar"), yfir á safnið "pakkar", væri hægt að nota skiptaumnafn("pakki","pakkar", væri hægt að nota skiptaumnafn("pakki","pakkar" svo framarlega sem skrá með nafni "pakki" er ekki til á safni "pakkar". Ekki er hægt að nota þessa aðgerð til að flytja skrár á milli diskadrifa þ.a. segðin

```
skiptaumnafn("a:pakki","b:pakki")
```

gæfi ósatt.

sækjaskrárdag

Notkun: ok := sækjaskrárdag(ár,mánuður,dagur;nafn)

Fyrir: nafn er strengur sem inniheldur löglegt nafn á skrá (með efnisskrárleið ef vill). Ekki skiptir máli hvaða gildi ár, mánuður og dagur innihalda.

Eftir: Ef nafn er löglegt nafn á skrá og allt annað er í lagi, fá innút viðföngin ár, mánuður og dagur gildi fjöldatalna sem endurspeglar þeirri dagsetningu sem var í gildi þegar viðkomandi skrá var síðast breytt. Að auki fær ok gildið satt. Ef einhver villa kemur upp, fær ok gildið ósatt og engu er að treysta með gildin í ár, mánuður og dagur.

ár er fjöldatala á bilinu 1980 til 2099. mánuður er fjöldatala á bilinu 1 til 12 þar sem 1=Janúar, 2=Febrúar ... 12=Desember. dagur er fjöldatala á bilinu 1 til 31.

Þessi aðgerð hefur í raun svipuð áhrif og aðgerðin sækjadaghandfangs en er þægilegri í notkun, ef viðkomandi skrá hefur ekki enn verið opnuð.

sækjaskrártíma

Notkun: ok := sækjaskrártíma(k,m,s;nafn)

Fyrir: nafn er strengur sem inniheldur löglegt nafn á skrá (með efnisskrárleið ef vill). Ekki skiptir máli hvaða gildi k, m og s innihalda.

Eftir: Ef nafn er löglegt nafn á skrá og allt annað er í lagi, fá innút viðföngin k, m og s, gildi fjöldatalna sem endurspeglar þá tímasetningu sem var í gildi þegar viðkomandi skrá var síðast breytt. Að auki fær ok gildið satt. Ef einhver villa kemur upp, fær ok gildið ósatt og engu er að treysta með gildin í k, m og s.

k er fjöldatala á bilinu 0 til 23. m er fjöldatala á bilinu 0 til 59. s er fjöldatala á bilinu 0 til 59.

Þessi aðgerð hefur í raun svipuð áhrif og aðgerðin sækjatímahandfangs en er þægilegri í notkun, ef viðkomandi skrá hefur ekki enn verið opnuð.

sækjadaghandfangs

Notkun: ok := sækjadaghandfangs(ár,mánuður,dagur;handfang)

Fyrir: handfang er opið handfang sem lýsir þeirri skrá sem um er að ræða og hefur verið fengið úr áður framkvæmdu kalli á opnaskrá eða búatilskrá. Ekki skiptir máli hvaða gildi ár, mánuður og dagur innihalda.

Eftir: Ef handfang er löglegt opið handfang og allt annað er í lagi, fá innút viðföngin ár, mánuður og dagur gildi fjöldatalna sem endurspeгла þá dagsetningu sem var í gildi þegar viðkomandi skrá var síðast breytt. Að auki fær ok gildið satt. Ef einhver villa kemur upp, fær ok gildið ósatt og engu er að treysta með gildin í ár, mánuður og dagur.

ár er fjöldatala á bilinu 1980 til 2099. mánuður er fjöldatala á bilinu 1 til 12 þar sem 1=Janúar, 2=Febrúar ... 12=Desember. dagur er fjöldatala á bilinu 1 til 31.

sækjatímahandfangs

Notkun: ok := sækjatímahandfangs(k,m,s;h)

Fyrir: h er opið handfang sem lýsir þeirri skrá sem um er að ræða og hefur verið fengið úr áður framkvæmdu kalli á opnaskrá eða búatilskrá. Ekki skiptir máli hvaða gildi k, m og s innihalda.

Eftir: Ef h er löglegt opið handfang og allt annað er í lagi, fá innút viðföngin, k, m og s, gildi fjöldatalna sem endurspeгла þá tímasetningu sem var í gildi þegar viðkomandi skrá var síðast breytt. Að auki fær ok gildið satt. Ef einhver villa kemur upp, fær ok gildið ósatt og engu er að treysta með gildin í k, m og s.

k er fjöldatala á bilinu 0 til 23. m er fjöldatala á bilinu 0 til 59. s er fjöldatala á bilinu 0 til 59.

setjadaghandfangs

Notkun: ok := setjadaghandfangs(;handfang,ár,mánuður,dagur)

Fyrir: handfang er opið handfang sem lýsir þeirri skrá sem um er að ræða og hefur verið fengið úr áður framkvæmdu kalli á opnaskrá eða búatilskrá. ár, mánuður og dagur eru heiltölur eða fjöldatölur sem endurspeгла þeirri dagsetningu sem stimpla á skrána með.

ár er á bilinu 1980 til 2099. mánuður er á bilinu 1 til 12 þar sem 1=Janúar, 2=Febrúar ... 12=Desember. dagur er á bilinu 1 til fjölda daga í viðkomandi mánuði. Ef um er að ræða febrúar, verður að taka til greina hvort viðkomandi ár er hlaupár eða ekki.

Eftir: Ef handfang er löglegt opið handfang og allt annað er í lagi, er skráin sem lýst er með handfang stimpluð með þeirri dagsetningu sem gefin er í ár, mánuður og dagur og ok fær gildið satt. Ef einhver villa kemur upp, fær ok gildið ósatt.

Það flokkast undir að villa hafi komið upp ef gildin sem gefin eru í ár, mánuður og dagur eru ekki innan þeirra marka sem löglegt er. Þar með talið er að gefa upp 29 daga í Febrúar ef uppgengið ár er ekki hlaupár. Slíkt verður þó ekki til þess að ok fái gildið ósatt, þar sem DOS gerir enga athugasemd, en dagsetning skrárinnar er er þar með orðin röng.

setjatímahandfangs

Notkun: ok := setjatímahandfangs(;h,k,m,s)

Fyrir: h er opið handfang sem lýsir þeirri skrá sem um er að ræða og hefur verið fengið úr áður framkvæmdu kalli á opnaskrá eða búatilskrá. k, m og s eru heiltölur eða fjöldatölur sem endurspeгла þeirri tímasetningu sem stimpla á skrána með.

Klukkustundir er á bilinu 0 til 23. Mínútur er á bilinu 0 til 59. Sekúndur er á bilinu 1 til 59.

Eftir: Ef h er löglegt opið handfang og allt annað er í lagi, er skráin sem lýst er með h stimpluð með þeirri tímasetningu sem gefin er í k, m og s og ok fær gildið satt. Ef einhver villa kemur upp, fær ok gildið ósatt.

Það flokkast undir að villa hafi komið upp ef gildin sem gefin eru í **k**, **m** og **s** eru ekki innan þeirra marka sem löglegt er. Slíkt verður þó ekki til þess að **ok** fái gildið ósatt, þar sem DOS gerir enga athugasemd, en tímasetning skrárinnar er er þar með orðin röng.

bráðabirgðarskrá

Notkun: `nafn := bráðabirgðarskrá(;leið)`

Fyrir: leið er strengur sem inniheldur löglega efnisskrárleið yfir á það safn sem innihalda á skrána. Efnisskrárleiðin þarf ekki nauðsynlega að enda á "
".

Eftir: Búin er til ný skrá á safninu sem efnisskrárleiðin leið vísar á, sem tryggt er að ekki sé þegar til staðar á því safni. Ef leið er strengur að lengd 0, er skráin búin til á aðalsafni sjálfgefins diskadrifs (sömu áhrif og ef leið hefði verið "

"). Ef leið inniheldur diskadrifsnafn, er skráin búin til á því diskadrifi. Ef allt er í lagi og engin villa kemur upp, fær nafn gildi strengs, sem inniheldur fullt nafn þeirrar skrár þ.e. efnisskrárleiðina leið að viðbættu þessu einkvæma nafni sem DOS býr til. Ef einhver villa kemur upp (t.d. ef leið er ólögleg efnisskrárleið), fær nafn gildið ósatt og er þá ekkert vitað um það, hvort einhver skrá hafi verið búin til eða ekki, né hvað hún þá heitir.

Þessi leið til að fá nafn á skrá uppgæfið frá DOS, er mjög hentug fyrir forrit, sem þurfa á því að halda að búa til skrá á meðan á keyrslu stendur, sem er svo eytt áður en keyrslu lýkur.

Athuga skal að DOS eyðir ekki sjálfkrafa skráum sem búnar eru til með þessum hætti. Það verður notandinn að gera sjálfur.

Varast ber að búa bráðabirgðarskrár til á sjálfgefnu safni (hægt með því að hafa leið = "."). Ástæða þess er sú að um leið og sjálfgefið safn breytist, vísar ". "til annars safns og þar með er skráin týnd.

færaskrárbendi

Notkun: `ok := færaskrárbendi(k1,v1;h,aðferð,k2,v2)`

Fyrir: **h** er opið handfang sem áður hefur fengist úr kalli á búatilskrá eða opnaskrá. **aðferð** er heiltala eða fjöldatala; 0, 1 eða 2. **k2** og **v2** eru fjöldatölur. Ekki skiptir máli hvaða gildi **k1** og **v1** innihalda.

Eftir: Færir les/skrif bendinn til í skránni sem lýst er með handfanginu **h** þ.a. síðari lestur eða skrift í þá skrá, hefst á þeim stað í skránni. **k2** og **v2** mynda til samans stóra (32 bita) fjöldatölu og segja til um yfir í hvaða stöðu á að færa bendinn. **aðferð** segir til um út frá hvaða stöðu tilfærslan miðast. Ef allt gengur eðlilega fyrir sig og engin villa kemur upp, fær **ok** gildið satt og **k1** og **v1** mynda til samans stóra (32 bita) fjöldatölu sem segir til um hver staða les/skrif bendis er eftir tilfærsluna, talið frá upphafi skrár. Ef einhver villa kemur upp, fær **ok** gildið ósatt og engu er að treysta með upplýsingarnar í **k1** og **v1**.

Ef **aðferð** = 0, þá miðast tilfærslan út frá byrjun skrár og bendir er færður í stöðu **k1:v1** (stór fjöldatala) talið frá byrjun skrár.

Ef **aðferð** = 1, þá miðast tilfærslan út frá núverandi stað les/skrif bendis í skránni og bendir er færður **k1:v1** (stór fjöldatala) aftur fyrir þann stað.

Ef **aðferð** = 2, þá miðast tilfærslan út frá endi skrár og les/skrif bendir er færður **k1:v2** aftur fyrir endi skrár.

Eitt einfalt en þægilegt dæmi um notkun væri kallið

`færaskrárbendi(hátt,lágt;handfang,1,0,0)`

Eftir þetta kall innihalda hátt:lágt stóra fjöldatölu sem segir til um hver staða les/skrif bendis er í skránni, án þess að færa hann nokkuð til.

Annað dæmi væri kallið

```
færaskrárbendi(hátt,lágt;handfang,2,0,0)
```

Eftir þetta kall, innihalda hátt:lágt stóra fjöldatölu sem segir til um hver stærð viðkomandi skrár er.

Kallið

```
færaskrárbendi(hátt,lágt;handfang,2,0,2000)
```

myndi stækka skrána um 2000 bæti, svo framarlega sem hún væri opin fyrir skrift. Ef skráin væri ekki opin fyrir skrift, kæmi upp villa.

skráraðgangur

Notkun: aðgangur := skráraðgangur(;nafn)

Fyrir: nafn er strengur sem inniheldur löglegt fullt nafn á skrá (þ.e. efnisskrárleið að viðbættu nafni skrár) sem til staðar er á uppgefnu safni.

Eftir: Ef nafn er löglegt fullt nafn á skrá sem til er á uppgefnu safni, fær aðgangur gildi fjöldatölu sem segir til um af hvaða tagi aðgangur að viðkomandi skrá er. Ef nafn er ólöglegt, ekki til eða einhver önnur villa kemur upp, fær aðgangur gildið ósatt.

Í DOS er geymd ein fjöldatala fyrir hvert nafn (sem getur verið nafn skrár, safns eða disks) sem lýsir því af hvaða tagi aðgangur að viðkomandi nafni er og hvaða aðgerðir má framkvæma á það. Þessi fjöldatala samanstendur af 16 bitum sem hver um sig segir sína sögu. Þessir bitar eru eftirfarandi:

Biti 0: Ef þessi biti er á, er eingöngu hægt að nota skrána til lestrar, ekki er hægt að framkvæma skrift (skráin er "read only"- aðeins til lestrar). Ef bitinn er af, er bæði hægt að framkvæma lestur og skrift á þessari skrá.

Biti 1: Ef þessi biti er á, er skráin falin (hidden) og kemur ekki fram við venjulegar aðgerðir við að lista út nöfn á disk, t.d. DIR skipun í DOS.

Biti 2: Ef þessi biti er á, er skráin kerfisskrá (system file) sem hefur mjög svipuð áhrif og þegar biti 1 er á. Venjulega er það þó þannig, að ef biti 2 er á, þá er biti 1 það líka.

Biti 3: Ef þessi biti er á, er viðkomandi nafn nafn disksins sem vísað er á. Sú "skrá" geymir engar aðrar nothæfar upplýsingar en nafnið. Á hverjum disk getur einungis verið eitt nafn með bita 3 á og það nafn getur einungis verið í aðalefnisskrá viðkomandi disks.

Biti 4: Ef þessi biti er á er viðkomandi nafn nafn á safni (directory). Slík nöfn er ekki hægt að skrifa í eða lesa úr eins og venjulegar skrár þar sem þau innihalda upplýsingar sem nauðsynlegar eru DOS til að vita hvað er geymt á viðkomandi safni.

Biti 5: Þessi biti er kallaður geymslubiti. Hann er notaður af nokkrum afritunarforritum s.s. BACKUP og RESTORE í DOS. Slík afritunarforrit nota sér það að í hvert skipti sem skrifað er í skrá (og þá líka þegar hún er búin til) er geymslubiti settur á. Þegar afritunarforritin taka afrit af slíkri skrá, taka þau geymslubita af. Næst þegar þau taka afrit, er þessi sama skrá einungis tekin með ef geymslubiti er á sem er líka til merkis um að skrifað hefur verið í skrána í millitíðinni.

Þessum bitum er svo hægt að blanda saman nokkurn vegin að vild. T.d. væri nafn með bita 1 og 4 á en alla aðra af, falið safn. Nafn með bita 1,2 og 4 á, væri falið kerfissafn o.s.frv. T.d. geymir DOS tvær faldar kerfisskrár (þ.e. með bita 1 og 2 á) á aðalsafni hvers kerfisdisks.

breytaskráraðgangi

Notkun: `ok := breytaskráraðgangi (;nafn,nýraðgangur)`

Fyrir: nafn er strengur sem inniheldur löglegt fullt nafn á skrá (þ.e. efnisskrárleið að viðbættu nafni skrár) sem til staðar er á uppgefnu safni. nýraðgangur er fjöldatala sem lýsir því hver aðgangur viðkomandi skrár á að verða.

Eftir: Breytir aðgangi (sjá skráraðgangur) að skránni nafn yfir í nýraðgangur. Ef skráin nafn finnst á uppgefnu safni og löglegt er að breyta aðgangi hennar yfir í nýraðgangur, er það gert og ok fær gildið satt. Ef skráin er ekki til, nýraðgangur er ólöglegur eða einhver önnur villa kemur upp, fær ok gildið ósatt.

safnlisti

Notkun: `listi := safnlisti (;leitargildi,aðgangur)`

Fyrir: leitargildi er strengur sem inniheldur löglegan leitarstreng fyrir DOS. aðgangur er heiltala eða fjöldatala sem lýsir því hvaða nöfn geta verið í þessum safnlista.

Eftir: Ef leitargildi er löglegur leitarstrengur fyrir DOS, fær listi gildi lista af strengjum sem hver um sig inniheldur nafn (án efnisskrárleiðar) sem passar við leitargildi og hefur skráraðgang aðgangur (sjá skráraðgangur). Ef leitargildi er ólöglegur leitarstrengur fyrir DOS, engin nöfn sem passa við hann hafa skráraðgang aðgangur eða einhver önnur villa kemur upp, fær listi gildið ósatt.

Kallið safnlisti("a:

.*",0) myndi skila lista nafna allra venjulegra skráa á aðalsafni diskadrifs a:

Kallið safnlisti("*.\"",0) myndi skila lista nafna allra venjulegra skráa á sjálfgefnu diskadrifi.

Athugað skal að með því setja á einhvern ákveðinn bita í skráraðgangi, fylgja með í listanum öll þau nöfn sem hafa þann bita af þ.a. kallið safnlisti("*.\"",1) gefur lista yfir allar faldar skrár á sjálfgefnu diskadrifi og safni, auk þess sem með í listanum fylgja nöfn allra venjulegra skráa. Eins er með kallið safnlisti("*.\"",16). Það gefur lista yfir öll söfn, auk þess sem með fylgja nöfn allra venjulegra skráa. Kallið safnlisti("*.\"",63) gefur því lista yfir allt sem væri að finna á sjálfgefnu safni sjálfgefins diskadrifs.

efnisskrársafnlisti

Notkun: `listi := efnisskrársafnlisti (;leitargildi,aðgangur)`

Fyrir: leitargildi er strengur sem inniheldur löglegan leitarstreng fyrir DOS. aðgangur er fjöldatala sem lýsir því hvaða nöfn geta verið í þessum safnlista.

Eftir: Það sama og í safnlisti að viðbættu því að hvert nafn í listanum listi, hefur skeytt framan á sig efnisskrárleið þess safns sem hún fannst á og er þar með orðið fullt nafn á skrá.

nákvæmursafnlisti

Notkun: `listi := nákvæmursafnlisti (;leitargildi,aðgangur)`

Fyrir: leitargildi er strengur sem inniheldur löglegan leitarstreng fyrir DOS. aðgangur er heiltala eða fjöldatala sem lýsir því hvaða nöfn geta verið í þessum safnlista.

Eftir: Það sama og í safnlisti að frátöldu því að hvert stak í listanum listi er nú listinn [nafn,aðgangur] (í stað þess að vera bara strengurinn nafn í safnlisti) þar sem nafn er strengur sem inniheldur nafn skrár (án efnisskrárleiðar) og aðgangur er sá aðgangur sem nafn hefur.

Þetta væri t.d. hægt að nota til að finna allar faldar skrár á sjálfgefnu safni sjálfgefins diskadrifs. Þá væri kallað

```
listi := nákvæmursafnlisti("*.*",2)
```

og svo rennt eftir listanum sem kemur út úr kallinu og öll nöfn fjarlægð sem ekki hafa aðgangur = 2.

söfn

Notkun: `listi := söfn(;leitargildi)`

Fyrir: leitargildi er strengur sem inniheldur löglegan leitarstreng fyrir DOS.

Eftir: Þau sömu og ef kallað væri á nákvæmursafnlisti(;leitargildi, 63), síðan rennt eftir listanum sem þar kemur út og skilin einungis eftir þau nöfn sem hafa bita 4 á (sjá skráraðgangur). Þ.e. listi fær gildi lista af strengjum sem hvert um sig er nafn safns sem til staðar er á því diskdrifi og safni sem leitargildi vísar til. Ef engin slík finnst eða einhver önnur villa kemur upp, fær listi gildið ósatt.

sjálfgefiðdrif

Notkun: `drif := sjálfgefiðdrif(;`

Fyrir: Ekkert..

Eftir: drif fær gildi fjöldatölu sem segir til um hvaða diskdrif er sjálfgefið diskdrif. drif = 0 er "a:", drif=1 er "b:" o.s.frv.

veljadrif

Notkun: `ok := veljadrif(;drif)`

Fyrir: drif er heiltala eða fjöldatala sem lýsir löglegu diskdrifi fyrir viðkomandi tölvu.

Eftir: Ef drif lýsir löglegu diskdrifi fyrir viðkomandi tölvu, fær ok gildið satt og sjálfgefið diskdrif er orðið það diskdrif sem drif lýsir. Ef drif er ekki löglegt diskdrif fyrir viðkomandi tölvu eða einhver önnur villa kemur upp, fær ok gildið ósatt. Drifi "a:" er lýst með 0, drifi "b:" með 1 o.s.frv.

Kafli 13

Einingin DOSSKRIU

skrálesabæti

Notkun: bæti := skrálesabæti(;handfang)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir lestraraðgang og lýsir þeirri skrá sem lesa á úr. Það verður að hafa fengist úr kalli á opnaskrá (ekki búatilskrá þar sem það handfang væri þá einungis opið fyrir skrift) eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að lesa úr (sjá um lokaskrá).

Eftir: Lesið er eitt bæti úr skránni sem lýst er með handfang frá þeim stað sem les/skrif bendir er og les/skrif bendir er færður aftur um 1. bæti fær gildi fjöldatölu sem er það bæti sem lesið var ef allt er í lagi og engin villa kemur upp. Ef einhver villa kemur upp (t.d. reynt að lesa aftur fyrir endi skrár, handfang er ekki löglegt o.s.frv) fær bæti gildið ósatt.

skráskrifabæti

Notkun: ok:= skráskrifabæti(;handfang,bæti)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir skriftaraðgang og lýsir þeirri skrá sem skrifa á í. Það verður að hafa fengist úr kalli á opnaskrá (fyrir skrift) eða búatilskrá eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að skrifa í (sjá um lokaskrá). bæti er fjöldatala sem er það bæti sem skrifa á.

Eftir: lágbæti fjöldatölu þæti (\lágbæti þæti) er skrifað í skránni sem lýst er með handfang frá þeim stað sem les/skrif bendir er og les/skrif bendir er færður aftur um 1. ok fær gildið satt ef engin villa kemur upp. Ef einhver villa kemur upp (t.d. reynt að skrifa í handfang sem ekki er opið fyrir skrift, handfang er ekki löglegt o.s.frv) fær ok gildið ósatt.

skrálesaorð

Notkun: orð:= skrálesaorð(;handfang)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir lestraraðgang og lýsir þeirri skrá sem lesa á úr. Það verður að hafa fengist úr kalli á opnaskrá (ekki búatilskrá þar sem það handfang væri þá einungis opið fyrir skrift) eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að lesa úr (sjá um lokaskrá).

Eftir: Lesið er eitt orð (tvö bæti) úr skránni sem lýst er með handfang frá þeim stað sem les/skrif bendir er og les/skrif bendir er færður aftur um 2. orð fær gildi fjöldatölu sem er það orð sem lesið var ef allt

er í lagi og engin villa kemur upp. Ef einhver villa kemur upp (t.d. reynt að lesa aftur fyrir enda skrár, handfang er ekki löglegt o.s.frv.) fær orð gildið ósatt.

Eitt orð samanstendur af tveimur bætum sem geymd eru með gildislægra bætið (lágbætið) á undan og það gildishærra (hábbætið) á eftir. Þegar orð er lesið úr handfangi eru því lesin tvö bæti. Fyrst það gildislægra og það látið vera gildislægri hluti útkomunnar. Síðan er það gildishærra lesið og það látið vera gildishærri hluti útkomunnar.

skraskrifaorð

Notkun: `ok:= skraskrifaorð(;handfang,orð)`

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir skriftaraðgang og lýsir þeirri skrá sem skrifa á í. Það verður að hafa fengist úr kalli á opnaskrá (fyrir skrift) eða búatilskrá eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að skrifa í (sjá um lokaskrá). orð er fjöldatala sem er það orð sem skrifa á.

Eftir: Orðið orð er skrifað í skrána sem lýst er með handfang frá þeim stað sem les/skrif bendir er og les/skrif bendir er færður aftur um 2. ok fær gildið satt ef engin villa kemur upp. Ef einhver villa kemur upp (t.d. reynt að skrifa í handfang sem ekki er opið fyrir skrift, handfang er ekki löglegt o.s.frv) fær ok gildið ósatt.

Eitt orð samanstendur af tveimur bætum sem geymd eru með gildislægra bætið á undan og það gildishærra á eftir. Þegar orð er skrifað í handfang eru því skrifuð tvö bæti. Fyrst gildislægri hluti orð og síðan gildishærri hluti orð (sjá um skrálesaorð).

skrálesastreng

Notkun: `fjöldi := skrálesastreng(;handfang,strengur,lengd)`

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir lestraraðgang og lýsir þeirri skrá sem lesa á úr. Það verður að hafa fengist úr kalli á opnaskrá (ekki búatilskrá þar sem það handfang væri þá einungis opið fyrir skrift) eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að lesa úr (sjá um lokaskrá). lengd er fjöldatala og er minni en 256. lengd segir til um hversu mörg bæti á í mesta lagi að lesa inn. strengur er strengur með strengstærð a.m.k. lengd.

Eftir: Inn í strengur eru lesin allt að lengd bæti úr skránni sem lýst er með handfang frá þeim stað sem les/skrif bendir er og les/skrif bendir er færður aftur um það mörg bæti. Ef engin villa kemur upp, fær fjöldi gildi fjöldatölu sem segir til um hversu mörg bæti voru lesin inn í strenginn strengur. Eftir lesturinn inniheldur strengur streng að lengd fjöldi. Ef einhver villa kemur upp, fær fjöldi gildið ósatt.

Ef lengd er stærri en 256 eða ef lengd er stærri en strengstærð strengur er ekkert lesið inn og fjöldi fær gildið ósatt.

Í flestum tilfellum verður gildið á fjöldi það sama og lengd. Það er þó ekki algilt. Ef færri en lengd bæti eru til staðar í því handfangi sem verið er lesa úr, verður gildið á fjöldi lægra en gildið á lengd.

Ef handfang lýsir aðalinntaki, og aðalinntak er lyklorðið, er einungis lesin lína og lína í einu. Þá verður fjöldi í flestum tilfellum lægri en lengd.

Það er til merkis um að búíð sé að lesa aftur í enda skrár, ef fjöldi fær gildið 0.

skraskrifastreng

Notkun: `ok := skraskrifastreng(;handfang,strengur)`

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir skriftaraðgang og lýsir þeirri skrá sem skrifa á í. Það verður að hafa fengist úr kalli á opnaskrá (fyrir skrift) eða búatilskrá eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að skrifa í (sjá um lokaskrá). strengur er strengur sem skrifa á í skrána.

Eftir: Tekin er lengd(;strengur) (þ.e. það bæti sem geymt er í sæti 0 í strengur) og það látið segja til um þann fjölda bæta sem skrifa á. Þau bæti eru síðan skrifuð úr strengnum strengur út í handfangið handfang og les/skrif bendir færður aftur um jafnmörg bæti. ok fær gildið satt ef náðist að skrifa umbeðinn fjölda bæta og engin villa kemur upp. Ef einhver villa kemur upp, fær ok gildið ósatt.

Það er jafnan merki þess að diskur sé fullur ef upp kemur sú staða að ok er ósatt.

skrálesastreng

Notkun: strengur := skrálesastreng(;handfang)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir lestraraðgang og lýsir þeirri skrá sem lesa á úr. Það verður að hafa fengist úr kalli á opnaskrá (ekki búatilskrá þar sem það handfang væri þá einungis opið fyrir skrift) eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að lesa úr (sjá um lokaskrá).

Eftir: Lesið er næsta bæti úr handfanginu sem handfang lýsir og les/skrif bendir færður aftur um 1. Það bæti er látið segja til um hve mörg bæti á að lesa inn í strenginn. Síðan er strengur af rétttri lengd búinn til og lesið inn í hann áður nefndur fjöldi bæta og les/skrif bendir færður aftur um þann fjölda bæta. Ef allt er í lagi, fær strengur gildi strengs sem inniheldur innlesnu bætinn með það bæti sem fyrst var lesið inn í sæti 0 (þ.e. það segir til um lengd strengsins). Ef einhver villa kemur upp, fær strengur gildið ósatt.

Það að strengur fær gildið ósatt, er oftast afleiðing þess að reynt er að lesa aftur fyrir endi skrár. Það gerist þá í því tilfelli þegar það bæti sem lesið er inn fyrst og notað er til að segja til um hversu mörg bæti á að lesa inn í strenginn, segir ranglega til um þann fjölda. Þ.e. sá fjöldi er hærri en fjöldi þeirra bæta sem eftir eru í skránni.

Strengi, sem skrifaðir hafa verið með skráskrifastreng, er hægt að lesa aftur inn með þessari aðgerð.

skráskrifastreng

Notkun: ok := skráskrifastreng(;handfang,strengur)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir skriftaraðgang og lýsir þeirri skrá sem skrifa á í. Það verður að hafa fengist úr kalli á opnaskrá (fyrir skrift) eða búatilskrá eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að skrifa í (sjá um lokaskrá). strengur er strengur sem skrifa á í skrána.

Eftir: Tekin er lengd(;strengur) (þ.e. það bæti sem geymt er í sæti 0 í strengur) og það látið segja til um þann fjölda bæta sem skrifa á. Það bæti er skrifað út í handfangið handfang og les/skrif bendir færður aftur um 1 bæti. Þar á eftir er sá fjöldi bæta úr strengnum strengur og les/skrif bendir færður aftur um jafn mörg bæti. ok fær gildið satt ef allt gengur eðlilega fyrir sig, en ósatt annars.

Það er jafnan merki þess að diskur sé fullur ef upp kemur sú staða að ok fær gildið ósatt.

skrálesageymslu

Notkun: fjöldi := skrálesageymslu(;handfang,strengur,lengd)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir lestraraðgang og lýsir þeirri skrá sem lesa á úr. Það verður að hafa fengist úr kalli á opnaskrá (ekki búatilskrá þar sem það handfang væri þá einungis opið fyrir skrift) eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að lesa úr (sjá um lokaskrá). lengd er fjöldatala. lengd segir til um hversu mörg bæti á í mesta lagi að lesa inn. strengur er strengur með strengstærð a.m.k. lengd.

Eftir: Inn í strengur eru lesin allt að lengd bæti úr skránni sem lýst er með handfang frá þeim stað sem les/skrif bendir er og les/skrif bendir er færður aftur um það mörg bæti. Þar sem strengur getur verið lengri en 255 stafir, er ekki lengur hægt að nota sæti 0 til að tákna lengd hans (þar sem bæti getur einungis haft gildin 0 til 255). Fyrsta bætið sem lesið er inn, fer því inn í sæti 0 í strengnum strengur. Ef engin villa kemur upp, fær fjöldi gildi fjöldatölu sem segir til um hversu mörg bæti voru lesin inn í strenginn strengur, annars fær fjöldi gildið ósatt.

Ef lengd er stærri en strengstærð strengur er ekkert lesið og fjöldi fær gildið ósatt.

Í flestum tilfellum verður gildið á fjöldi það sama og lengd. Það er þó ekki algilt. Ef færri en lengd bæti eru til staðar í því handfangi sem verið er lesa úr, verður gildið á fjöldi lægra en gildið á lengd.

Ef handfang lýsir aðalinntaki, og aðalinntak er lykilorðið, er einungis lesin lína og lína í einu. Þá verður fjöldi í flestum tilfellum lægri en lengd.

Það er til merkis um að búíð sé að lesa aftur í enda skrár, ef fjöldi fær gildið 0.

skráskrifageymslu

Notkun: ok := skráskrifastreng(;handfang,strengur,lengd)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir skriftaraðgang og lýsir þeirri skrá sem skrifa á í. Það verður að hafa fengist úr kalli á opnaskrá (fyrir skrift) eða búatilskrá eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að skrifa í (sjá um lokaskrá). lengd er fjöldatala sem segir til um hversu mörg bæti á að skrifa. strengur er strengur sem skrifa á í skrána og hefur strengstærð a.m.k. lengd.

Eftir: lengd fyrstu bætinn í strengnum strengur (það fyrsta úr sæti 0) eru skrifuð út í handfangið handfang og les/skrif bendir færður aftur um jafn mörg bæti. ok fær gildið satt ef náðist að skrifa umbeðinn fjölda bæta og engin villa kemur upp. Ef einhver villa kemur upp, fær ok gildið ósatt.

Ef lengd er stærri en strengstærð strengur, er ekkert skrifað og ok fær gildið ósatt.

Það er jafnan merki þess að diskur sé fullur ef upp kemur sú staða að ok er ósatt.

skráskrifaminni

Notkun: ok := skráskrifaminni(;handfang,kafli,vistfang,lengd)

Fyrir: handfang er handfang (sem er fjöldatala) sem opið er fyrir skriftaraðgang og lýsir þeirri skrá sem skrifa á í. Það verður að hafa fengist úr kalli á opnaskrá (fyrir skrift) eða búatilskrá eða þá að það er eitt af þeim handföngum sem DOS setur upp í upphafi keyrslu og er hægt að skrifa í (sjá um lokaskrá). lengd er fjöldatala sem segir til um hversu mörg bæti á að skrifa. kafli og vistfang mynda til samans bendi út í minni tölvunnar (á forminu kafli:vistfang) sem segir til um hvar vitja á þeirra bæta sem skrifa á.

Eftir: lengd fyrstu bætinn, á þeim stað sem kafli:vistfang vísar á, eru skrifuð út í handfangið handfang. ok fær gildið satt ef náðist að skrifa umbeðinn fjölda bæta og engin villa kemur upp. Ef einhver villa kemur upp, fær ok gildið ósatt.

Það er jafnan merki þess að diskur sé fullur ef upp kemur sú staða að ok er ósatt.

Ef eingöngu eru skrifuð út Fjölnisgildi (strengir, hlunkar o.þ.h.) er engin þörf á að nota þessa aðgerð. Þá eru aðgerðirnar `skraskrifastreng` og `skraskrifaurstreng` nægilega öflugar.

Sem dæmi um hvernig nota má þessa aðgerð mætti taka:

```
s := "Halló þú þarna",
skraskrifaminni(
    ;h
    ,\kafli s
    ,\vistfang s+1
    ,s \strengsækjabæti 0
)
```

Þetta er hægt að gera á einfaldari hátt með:

```
s := "Halló þú þarna",
skraskrifastreng(;h,s)
```


Kafli 14

Einingin DOSSTIU

prentastaf

Notkun: `prentastaf(;stafur)`

Fyrir: `stafur` er fjöldatala á bilinu 0 til 255.

Eftir: Stafurinn `stafur` er skrifaður út á aðalprentara. Þar sem DOS gefur ekki neinn möguleika á því að vita hvort einhver villa hefur komið upp, er það ekki heldur gert hér. Sjá þó um `prentastreng` um betri aðferð.

prentastreng

Notkun: `prentastreng(;s)`

Fyrir: `s` er strengur sem getur innihaldið hvað sem er.

Eftir: Innihald strengsins `s` er skrifað út á aðalprentara. `prentastreng` notar `prentastaf` til að skrifa strenginn staf fyrir staf. Sjá því umfjöllun um `prentastaf`.

Athugað skal, að þar sem DOS gefur engar upplýsingar um hvort villa kemur upp eða ekki ef þetta kall er notað, getur oft verið þægilegra að nota aðrar aðgerðir sem það gera. Þar er sérstaklega mælt með því að nota "handfangaköll". Ef t.d. tölvan er sett þannig upp að prentari er tengdur samhliða við tölvuna (parallel) og við aðalsamhliðatengið (PRN) er hægt að nota sér það á eftirfarandi hátt:

```
prentari := opnaskrá("PRN",'s'),
ef prentari þá
  str := "stafur",
  ef ekki skráskrifastreng(;prentari,str) þá
    skrifa("Villa við skrift á prentara")
  eflok,
  ef ekki lokaskrá(;prentari) þá
    skrifa("Villa við frágang prentara")
  eflok
annars
  skrifa("Get ekki notað prentara")
eflok
```

náistafánstopps

Notkun: stafur := náistafánstopps(;)

Fyrir: Engin skilyrði.

Eftir: Ef einhver stafur er tilbúinn í aðalinntaki, er sá stafur lesinn og **stafur** fær gildi fjöldatölu sem inniheldur innlesna stafinn í gildislægra bætínu. Ef enginn stafur er til staðar í aðalinntaki, fær **stafur** gildið ósatt.

Venjulega þegar verið er að lesa úr aðalinntaki, athugar DOS hvort notandi hefur ýtt á Ctrl-Break eða Ctrl-PrtSc og framkvæmir þá viðeigandi ráðstafandir (stöðvar forritið eða skrifar skjá út á prentara). Það er aftur á móti ekki gert ef þessi aðgerð er notuð við innlestur.

Þegar lesið er inn með þessari aðgerð, notar DOS "útvíkkuð gildi" til að greina á milli þess hvort ýtt hefur verið á venjulegan lykil eða aðgerðarlykil. Ef ýtt er á venjulegan lykil, verður útkoman einungis gildi þess stafs sem lykillinn gefur. Ef aftur á móti er ýtt á aðgerðarlykil, þarf tvö köll í þessa aðgerð til að fá á hreint hvaða lykil er um að ræða. Hvort ýtt er á aðgerðarlykil eða ekki er hægt að sjá út frá því að ef stafur fær gildið 0, hefur verið ýtt á aðgerðarlykil og annað kall þarf til að komast að því hvaða lykill það var. Eftirfarandi tafla sýnir hvaða aðgerðarlykil um er að ræða, miðað við það gildi sem stafur fær úr seinna kallinu:

Ef stafur er Þá hefur verið ýtt á

3 NUL (þ.e. alt-0)

15 Shift-Tab

16-25 Alt-Q, W, E, R, T, Y, U, I, O, P

30-38 Alt-A, S, D, F, G, H, J, K, L

44-50 Alt-Z, X, C, V, B, N, M

59-68 F1-F10

71 Home

72 Ör upp

73 PgUp

75 Ör til vinstri

77 Ör til hægri

79 End

80 Ör niður

81 PgDn

82 Ins

83 Del

84-93 F11-F20 (Shift F1-Shift-F10)

94-103 F21-F30 (Ctrl-F1-Ctrl-F10)

104-113 F31-F40 (Alt-F1-Alt-F10)

114 Ctrl-PrtSc

115 Ctrl-Ör til vinstri

116 Ctrl-Ör til hægri

117 Ctrl-End

118 Ctrl-PgDn
 119 Ctrl-Home
 120-131 Alt-1,2,3,4,5,6,7,8,9,0,-,=
 132 Ctrl-PgUp

skrifastafánstopps

Notkun: skrifastafánstopps(;stafur)

Fyrir: stafur er fjöldatala sem inniheldur stafinn sem skrifa á í gildislægra bätinu.

Eftir: Stafurinn stafur er skrifaður beint í aðalúttak. Þar sem DOS gefur ekki til kynna hvort einhver villa hefur komið upp, er það ekki heldur hægt hér.

Venjulega þegar verið er að skrifa í aðalúttak, athugar DOS hvort notandi hefur ýtt á Ctrl-Break eða Ctrl-PrtSc og framkvæmir þá viðeigandi ráðstafandir (stöðvar forritið eða skrifar skjá út á prentara). Það er aftur á móti ekki gert ef þessi aðgerð er notuð við skrift.

lesastafánbergmáls

Notkun: stafur := lesastafánbergmáls(;)

Fyrir: Ekkert.

Eftir: Lesinn er einn stafur úr aðalinntaki (beðið er eftir að hann sé sleginn inn, ef hann er ekki þar til staðar) og stafur fær gildi fjöldatölu með innlesinn staf í gildislægra bätinu.

Venjulega þegar stafir eru lesnir af aðalinntaki (ef aðalinntak er lyklaborð), eru innlesnir stafir birtir á skjá jafnóðum. Það gerist ekki ef þessi aðgerð er notuð þ.e. ekkert birtist á skjá.

Sjá einnig um útvíkkuð gildi í náistafánstopps.

lesastafánstopps

Notkun: stafur := lesastafánstopps(;)

Fyrir: Ekkert.

Eftir: Lesinn er einn stafur úr aðalinntaki (beðið er eftir að hann sé sleginn inn, ef hann er ekki þar til staðar) og stafur fær gildi fjöldatölu með innlesinn staf í gildislægra bätinu.

Venjulega þegar stafir eru lesnir af aðalinntaki (ef aðalinntak er lyklaborð), eru innlesnir stafir birtir á skjá jafnóðum. Það gerist ekki ef þessi aðgerð er notuð þ.e. ekkert birtist á skjá.

Venjulega þegar verið er að skrifa í aðalúttak, athugar DOS líka hvort notandi hefur ýtt á Ctrl-Break eða Ctrl-PrtSc og framkvæmir þá viðeigandi ráðstafandir (stöðvar forritið eða skrifar skjá út á prentara). Það er aftur á móti ekki gert ef þessi aðgerð er notuð við skrift.

Sjá einnig um útvíkkuð gildi í náistafánstopps.

erinntak

Notkun: sanngildi := erinntak(;)

Fyrir:

Ekkert.

Eftir: sanngildi fær gildið satt ef einhver stafur er tilbúinn til innlestrar úr aðalinntaki, annars fær sanngildi gildið ósatt.

Kafi 15

Einingin DOSYMISL

sækjadag

Notkun: sækjadag(ár,mánuður,mánaðardagur,vikudagur;)

Fyrir: Ekkert.

Eftir: Sækir þá dagsetningu sem í gildi er á tölvunnu og stingur fjöldatölum inn í innútvíðföngin ár, mánuður, mánaðardagur og vikudagur sem endurspeglar þeirri dagsetningu.

ár verður fjöldatala á bilinu 1980 til 2099.

mánuður verður fjöldatala á bilinu 1 til 12 þar sem 1=janúar, 2=febrúar ... 12=desember.

mánaðardagur verður fjöldatala á bilinu 1 til 31.

vikudagur verður fjöldatala á bilinu 0 til 6 þar sem 0=sunnudagur, 1=mánudagur ... 6=laugardagur.

setjadag

Notkun: ok := setjadag(;ár,mánuður,mánaðardagur)

Fyrir: ár, mánuður og mánaðardagur innihalda fjöldatölur sem til samans mynda löglega dagsetningu.

ár er fjöldatala á bilinu 1980 til 2099.

mánuður er fjöldatala á bilinu 1 til 12 þar sem 1=janúar, 2=febrúar ... 12=desember.

mánaðardagur er fjöldatala á bilinu 1 til 31.

Eftir: Ef ár, mánuður og mánaðardagur mynda til samans löglega dagsetningu, er dagsetning tölvunnar stillt þannig og ok fær gildið satt. Ef þau mynda aftur á móti ólöglega dagsetningu, eða einhver önnur villa kemur upp fær ok gildið ósatt og dagsetningu tölvunnar hefur að öllum líkindum ekkert verið breytt.

sækjatíma

Notkun: sækjatíma(klukkustundir,mínútur,sekúndur;)

Fyrir: Ekkert.

Eftir: Sækir þá tímasetningu sem í gildi er á tölvunnu og stingur fjöldatölum inn í innútvíðföngin klukkustundir, mínútur og sekúndur sem endurspeglar þeirri tímasetningu.

klukkustundir verður fjöldatala á bilinu 0 til 23.

mínútur verður fjöldatala á bilinu 0 til 59.

sekúndur verður fjöldatala á bilinu 0 til 59.

setjatíma

Notkun: `ok := setjatíma(;klukkustundir,mínútur,sekúndur)`

Fyrir: klukkustundir, mínútur og sekúndur innihalda fjöldatölur sem til samans mynda löglega tímasetningu.

klukkustundir er fjöldatala á bilinu 0 til 23.

mínútur er fjöldatala á bilinu 0 til 59.

sekúndur er fjöldatala á bilinu 0 til 59.

Eftir:

Ef klukkustundir, mínútur og sekúndur mynda til samans löglega tímasetningu, er tímasetning tölvunnar stillt þannig og ok fær gildið satt. Ef þau mynda aftur á móti ólöglega tímasetningu, eða einhver önnur villa kemur upp fær ok gildið ósatt og tímasetningu tölvunnar hefur að öllum líkindum ekkert verið breytt.

setjaöryggiá

Notkun: `setjaöryggiá(;`

Fyrir: Ekkert.

Eftir: Setur DOS skriftaröryggi á, hvort sem það var á fyrir eða ekki.

Þegar skriftaröryggi er á, les DOS aftur hvert einasta bæti sem það skrifar út á disk, ber það saman við það bæti sem átti að skrifa og athugar hvort þau eru eins. Ef þau eru ekki eins, þá hefur komið upp villa, annars er allt í lagi.

Þegar skriftaröryggi er af (ekki á), fer enginn slíkur samanburður fram og DOS einungis skrifar út á disk það sem um er beðið. Það er því heilmikið öryggi í því að hafa skriftaröryggi á, en um leið dregur það mjög úr skriftarhraða (nær tvöfalt).

takaöryggiaf

Notkun: `takaöryggiaf(;`

Fyrir: Ekkert.

Eftir: Tekur DOS skriftaröryggi af, hvort sem það var á fyrir eða ekki.

Þegar skriftaröryggi er á, les DOS aftur hvert einasta bæti sem það skrifar út á disk, ber það saman við það bæti sem átti að skrifa og athugar hvort þau eru eins. Ef þau eru ekki eins, þá hefur komið upp villa, annars er allt í lagi.

Þegar skriftaröryggi er af (ekki á), fer enginn slíkur samanburður fram og DOS einungis skrifar út á disk það sem um er beðið. Það er því heilmikið öryggi í því að hafa skriftaröryggi á, en um leið dregur það mjög úr skriftarhraða (nær tvöfalt).

útgáfunúmer

Notkun: útgáfunúmer (aðalnúmer, undirnúmer;)

Fyrir: Ekkert.

Eftir: aðalnúmer og undirnúmer fá gildi fjöldatalna sem endurspegla þeirri útgáfu af DOS sem í gangi er á tölvunni. Ef t.d. útgáfa 3.10 er í gangi, fær aðalnúmer gildið 3 og undirnúmer gildið 10.

Þessa aðgerð getur verið mjög þægilegt að hafa, þegar framkvæma þarf einhverjar aðgerðir sem ekki voru til (eða voru til í annarri mynd) í eldri útgáfum stýrikerfisins. Þessi aðgerð er þá notuð til að segja til um hvort viðkomandi aðgerðir ganga eða ekki.

ígrípsstaður

Notkun: ígrípsstaður (kafla, vistfnag; ígríp)

Fyrir: ígríp inniheldur fjöldatölu á bilinu 0 til 255 sem segir til um það ígrípsnúmer sem verið er að athuga. Ekki skiptir máli hvaða gildi eru í kafla og vistfang.

Eftir: Fyrir þá sem til þekkja er þessi aðgerð notuð til að komast að því, hvar í minni sá forritsstubbur er sem afgreiðir viðkomandi ígrípið (interrupt) ígríp. Innútvíðföngin kafla (segment) og vistfang (offset) fá gildi fjöldatalna sem til samans mynda bendi út í minni á viðkomandi forritsstubb.

staðastopps

Notkun: staða := staðastopps(;)

Fyrir: Ekkert.

Eftir: staða fær gildið satt ef stopp er á, annars ósatt.

Þegar stopp er á er í sumum tilfellum hægt að stoppa forrit í keyrslu (þó það sé nú ekki alltaf þannig í raun) en þegar stopp er af á ekki að vera hægt að stoppa forrit í keyrslu nema þau séu að lesa/skrifa úr/í aðalinntak með aðgerðum sem bjóða upp á að athugað sé hvort ýtt hafi verið á Ctrl-Break. Flestar aðgerðir athuga hvort það hefur verið gert. Sjá þó um náistafánstopps, skrifastafánstopps, lesastafánbergmáls og lesastafánstopps.

stillastopp

Notkun: stillastopp(;staða)

Fyrir: staða inniheldur gildið satt eða ósatt, eftir því hvort setja á stopp á eða taka það af.

Eftir: Ef staða inniheldur gildið satt, er stopp sett á, annars er stopp tekið af.

Þegar stopp er á er í sumum tilfellum hægt að stoppa forrit í keyrslu (þó það sé nú ekki alltaf þannig í raun) en þegar stopp er af á ekki að vera hægt að stoppa forrit í keyrslu nema þau séu að lesa/skrifa úr/í aðalinntak með aðgerðum sem bjóða upp á að athugað sé hvort ýtt hafi verið á Ctrl-Break. Flestar aðgerðir athuga hvort það hefur verið gert. Sjá þó um náistafánstopps, skrifastafánstopps, lesastafánbergmáls og lesastafánstopps.

Kafi 16

Einingin FLEYFÖLL

Einingin FLEYFÖLL inniheldur gagnleg föll sem vinna með fleytitölur. Hún vísar einingis í falin kjarnastef.

abs

Notkun: `s := \abs x`

Fyrir: `x` er fleytitala.

Eftir: `s` inniheldur fleytitölu sem er $|x|$.

asin

Notkun: `s := \asin x`

Fyrir: `x` er fleytitala milli -1.0 og 1.0 að báðum meðtöldum.

Eftir: `s` inniheldur fleytitölu sem er horn í radíönum þ.a. $\sin s = x$.

atan2

Notkun: `s := atan2(;x,y)`

Fyrir: `x` og `y` eru fleytitölur.

Eftir: `s` inniheldur horn vigrans (x,y) . T.d. skilar `atan2(;1.0,0.0)` horninu 0.0 og `atan2(;0.0,1.0)` skilar horninu $\pi/2$.

cos

Notkun: `s := \cos x`

Fyrir: `x` er fleytitala sem er horn í radíönum.

Eftir: `s` inniheldur fleytitölu sem er kósínus hornsins `x`.

exp

Notkun: $s := \backslash \exp x$

Fyrir: x er fleytitala.

Eftir: s inniheldur fleytitölu sem er e^x .

ln

Notkun: $s := \backslash \ln x$

Fyrir: x er jákvæð fleytitala.

Eftir: s inniheldur fleytitölu sem er $\log_e x$.

max

Notkun: $s := \max(;)$

Fyrir: Ekkert.

Eftir: s inniheldur stærstu mögulega fleytitölu.

min

Notkun: $s := \min(;)$

Fyrir: Ekkert.

Eftir: s inniheldur minnstu mögulega jákvæða fleytitölu.

radian

Notkun: $s := \backslash \text{radian } x$

Fyrir: x er fleytitala sem stendur fyrir horn í gráðum.

Eftir: s inniheldur fleytitölu sem er sama horn í radíönum.

rót

Notkun: $s := \backslash \text{rót } x$

Fyrir: x er fleytitala sem er ekki neikvæð.

Eftir: s inniheldur kvaðratrót x .

sin

Notkun: $s := \backslash \sin x$

Fyrir: x er fleytitala sem er horn í radíönum.

Eftir: s inniheldur fleytitölu sem er sínus hornsins x .

Kafli 17

Einingin INN

Einingin INN inniheldur stef sem lesa ýmiss konar gildi af aðalinntaki. Einingin notar biðminni (e. *buffer*) og er því ekki ráðlegt að nota aðrar aðferðir samtímis til að lesa frá aðalinntaki.

lesa

Notkun: `s := lesa(;)`

Fyrir: Ekkert.

Eftir: `s` inniheldur gildi sem lesin var úr aðalinntaki. Læsileg gildi eru eftirfarandi:

- Heiltölur milli -32768 og 32767, á sama sniði og í Fjölni
- Fleytitölur á sama sniði og í Fjölni
- Listar á sniðinu `[g1,...,gn]` þar sem `g1` til `gn` eru læsileg gildi.

Milli lesinna gilda og einstakra gilda í lista geta verið bilstafir, dálkstafir (`tab`) og línulok. Ef villa verður í lestri verður skrifuð melding þess efnis og keyrsla forritsins stöðvuð.

lesalínu

Notkun: `s := lesalínu(;)`

Fyrir: Ekkert.

Eftir: Les eina línu úr aðal inntaki, sé endir inntaks ekki fundinn, þó aldrei fleiri en 255 stafir. `s` fær gildi strengs sem inniheldur þá línu sem lesin var. Í sæti 0 í `s` er að finna fjölda lesinna stafa (þ.e. lengd línunnar).

Ef komið er í enda inntaks, sem markast af því að lesið hefur verið ASCII gildið 26 (`Ctrl-z`) eða ASCII gildið 0 (`NULL`), fær `s` gildið ósatt.

Endir línu er venjulega markaður af vendistaf (*carriage return*, ASCII gildi 13) og línuskipti staf (e. *linefeed*, ASCII gildi 10). Lesalínu vinnur rétt þó endir línu sé eingöngu markaður með línuskipti staf. Þessi tvö gildi verða þó aldrei hluti innihaldsins í `s`.

Ef línan sem verið er að lesa inn, er lengri en 255 stafir, eru einungis lesnir 255 stafir, þeim skilað í strengnum `s` og afgangurinn er látinn bíða næsta kalls á lesalínu.

lesastaf

Notkun: `s := lesastaf(;`

Fyrir: Ekkert.

Eftir: `s` inniheldur staf sem lesinn var úr línu úr aðal inntaki. Næst þegar kallað er á stefið skilar það næsta staf úr línunni. Ef farið er út fyrir enda línunnar fær `s` gildið 13 (vendistafur). Ef enginn stafur var lesin úr aðal inntaki þá skilar stefið sanngildinu ósatt.

næstistafur

Notkun: `s := næstistafur(;`

Fyrir: Ekkert.

Eftir: `s` er gildi stafs úr línu í aðal inntaki. Ef komið er út fyrir enda línunnar fær `s` gildi vendistafs (13). Ef engin stafur er til staðar fær `s` gildið ósatt. Þegar kallað er næst á stefið þá skilar það sama staf nema að færð hafi verið staðsetningin innan línunnar með því að kalla á stefið lesastaf.

Kafli 18

Einingin KJARNI

Stef sem eiga við fjöldataölur

```
!  
%  
&  
*  
+  
-  
/  
<  
<=|  
<>|  
=|  
>|  
>=  
|  
||  
ertala  
fdeiling  
fmargfeldi  
fmismunur  
fsumma  
háþæti  
hhliðra  
lágbæti  
minnka  
stækka  
vhliðra
```

Stef sem eiga við fleytitölur

```
***  
+++  
---  
///
```

```

<<<
<=<=<=
<><><>
===
>>>
>=>=>=
brot
erfleyt
erfleytneikvæð
erfleytnúll
fjötilfleyt
fleytmínus
fleytitala
fleyttilfjöl
formerki
veldi

```

Stef sem eiga við heiltölur

```

!
%%
&
**
++
--
//
<<
<=<=
<><>
==
>>
>=>=
|
||
ertala
hábæti
hdeiling
hhliðra
hmargfeldi
hmismunur
hsumma
lágæti
vhliðra

```

Stef sem eiga við hlið

```

innæti
útbæti

```

Stef sem eiga við hlunka

erhlunkur
 hlunksetja
 hlunkstærð
 hlunksækja
 hlunkur

Stef sem eiga við pör

:
 erpar
 halasetja
 hali
 haus
 haussetja

Stef sem eiga við stegildi

erstef
 innfjöldi
 innútfjöldi

Stef sem eiga við strengi

erstrengur
 kafli
 strengflytja
 strengfylla
 strengsetjabæti
 strengsetjaorð
 strengstærð
 strengsækjabæti
 strengsækjaorð
 strengur
 vistfang

Stef sem eiga við tómagildið

ertóm

Stef sem eiga við hlunka og strengi

fylkissetja1
 fylkissækja1

Stef sem vinna beint á minni

bætfylla
flytja
fylla
kíkjabæti
kíkjaorð
orðfylla

Stef til ýmissar notkunar

brjóta
hætta
ígrip10
laus_K
lesastaf
msdos
RSskilti
safna
skalsafna
skrifastaf

Kjarnastef í stafrófsröð

!

Notkun: $s := ! x$

Fyrir: x er fjölda- eða heiltala.

Eftir: s inniheldur töluna sem kemur út þegar bitunum sem tákna x er snúið við ($0 \rightarrow 1$ og $1 \rightarrow 0$).

%

Notkun: $s := x \% y$

Fyrir: x og y eru fjöldatölur.

Eftir: s inniheldur afganginn þegar x er deilt með y .

%%

Notkun: $s := x \% \% y$

Fyrir: x og y eru heiltölur.

Eftir: s inniheldur afganginn þegar x er deilt með y .

&

Notkun: $s := x \& y$

Fyrir: x og y eru fjölda- eða heiltölur.

Eftir: s inniheldur töluna sem kemur út þegar x og y eru "og-uð"saman bita fyrir bita.

Notkun: $s := x * y$

Fyrir: x og y eru fjöldatölur.

Eftir: s inniheldur margfeldi x og y.

Notkun: $s := x ** y$

Fyrir: x og y eru heiltölur.

Eftir: s inniheldur margfeldi x og y.

Notkun: $s := x *** y$

Fyrir: x og y eru hvorutveggja fleytitölur.

Eftir: s er fleytitala sem er margfeldi x og y.

+

Notkun: $s := x + y$

Fyrir: x og y eru fjöldatölur.

Eftir: s inniheldur summu x og y.

++

Notkun: $s := x ++y$

Fyrir: x og y eru heiltölur.

Eftir: s inniheldur summu x og y.

+++

Notkun: $s := x +++ y$

Fyrir: x og y eru hvorutveggja fleytitölur.

Eftir: s er fleytitala sem er summa x og y.

-

Notkun: $s := x - y$

Fyrir: x og y eru fjöldatölur.

Eftir: s inniheldur mismun x og y.

—

Notkun: $s := x -- y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur mismun x og y .

—

Notkun: $s := x --- y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s er fleytitala sem er mismunur x og y .

/

Notkun: $s := x / y$ **Fyrir:** x og y eru fjöldatölur.**Eftir:** s inniheldur kvóta x og y . Ef y er 0 þá stöðvast keyrsla forritsins, DOS kerfið tekur stjórnina og skrifar út villumeldingu. (e. *divide overflow*).

//

Notkun: $s := x // y$ **Fyrir:** x og y eru heiltölur, y má ekki vera núll.**Eftir:** s inniheldur kvóta x og y . Ef y er 0 þá stöðvast keyrsla forritsins, DOS kerfið tekur stjórnina og skrifar út villumeldingu. (e. *divide overflow*).

///

Notkun: $s := x /// y$ **Fyrir:** x og y eru fleytitölur. y má ekki vera núll og ef x er ekki núll þá má kvóti x og y ekki vera of stór (stærri en $1.79 \cdot 10^{308}$ að tölugildi) og ekki of lítill (minni en $1.11 \cdot 10^{-308}$ að tölugildi).**Eftir:** s er fleytitala sem er kvóti x og y .

:

Notkun: $s := x : y$ **Fyrir:** x og y eru gildi af hvaða tagi sem er.**Eftir:** Úthlutað er nýju pari og x er geymt í hausnum og y í halanum í parinu. s verður þetta par. Ef ekki er hægt að úthluta parinu vegna þess að allt minni er upp urið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.

<

Notkun: $s := x < y$ **Fyrir:** x og y eru fjöldataölur.

Eftir:

 s inniheldur sanngildi fyrir samanburðinn $x < y$, þ.e. satt þá og því aðeins að x sé minna en y , annars ósatt.

«

Notkun: $s := x \ll y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \ll y$, þ.e. satt þá og því aðeins að x sé minna en y , annars ósatt.

«<

Notkun: $s := x \lll y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \lll y$, þ.e. satt þá og því aðeins að x sé minna en y , annars ósatt.

<=

Notkun: $s := x \leq y$ **Fyrir:** x og y eru fjöldataölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \leq y$, þ.e. satt þá og því aðeins að x er minna en eða sama og y , annars ósatt.

<=<=

Notkun: $s := x \leq\leq y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \leq\leq y$, þ.e. satt þá og því aðeins að x er minna en eða sama og y , annars ósatt.

<=<=<=

Notkun: $s := x \leq\leq\leq y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \leq\leq\leq y$, þ.e. satt þá og því aðeins að x sé minna en eða sama og y , annars ósatt.

<>

Notkun: $s := x <> y$ **Fyrir:** x og y eru fjöldatölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x <> y$, þ.e. satt þá og því aðeins að x sé ekki sama og y , annars ósatt.

<><>

Notkun: $s := x <><> y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x <><> y$, þ.e. satt þá og því aðeins að x sé ekki sama og y , annars ósatt.

<><><>

Notkun: $s := x <><><> y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x <><><> y$, þ.e. satt þá og því aðeins að x sé ójafnt y , annars ósatt.

=

Notkun: $s := x = y$ **Fyrir:** x og y eru fjöldatölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x = y$, þ.e. satt þá og því aðeins að x sé sama og y , annars ósatt.

==

Notkun: $s := x == y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x == y$, þ.e. satt þá og því aðeins að x sé sama og y , annars ósatt.

===

Notkun: $s := x >=>=> y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x === y$, þ.e. satt þá og því aðeins að x sé sama og y , annars ósatt.

>

Notkun: $s := x > y$ **Fyrir:** x og y eru fjöldatölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x > y$, þ.e. satt þá og því aðeins að x sé stærra en y , annars ósatt.

»

Notkun: $s := x \gg y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \gg y$, þ.e. satt þá og því aðeins að x sé stærra en y , annars ósatt.

»»

Notkun: $s := x \ggg y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \ggg y$, þ.e. satt þá og því aðeins að x sé stærra en y , annars ósatt.

>=

Notkun: $s := x \geq y$ **Fyrir:** x og y eru fjöldatölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \geq y$, þ.e. satt þá og því aðeins að x sé stærra en eða sama og y , annars ósatt.

>=>=

Notkun: $s := x \geq\geq y$ **Fyrir:** x og y eru heiltölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \geq\geq y$, þ.e. satt þá og því aðeins að x sé stærra en eða sama og y , annars ósatt.

>=>=>=

Notkun: $s := x \geq\geq\geq y$ **Fyrir:** x og y eru hvorutveggja fleytitölur.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \geq\geq\geq y$, þ.e. satt þá og því aðeins að x sé stærra en eða sama og y , annars ósatt.

|

Notkun: $s := x \mid y$ **Fyrir:** x og y eru fjölda- eða heiltölur.**Eftir:** s inniheldur töluna sem kemur út þegar bætinn sem tákna x og y eru "eð-uð"saman bita fyrir bita.

||

Notkun: $s := x \mid y$ **Fyrir:** x og y eru fjölda- eða heiltölur.**Eftir:** s inniheldur misgildis-eða af x og y . Þ.e. hver einstakur bita s er 1 þá og því aðeins að annar samsvarandi bita í x og y hafi verið 1.

brjóta

Notkun: $s := \backslash\text{brjóta } x$ **Fyrir:** x er hvaða gildi sem er.**Eftir:** Stefið veldur ígripi 3. Gildinu. x er skilað í s ef notandinn leyfir keyrslu að halda áfram eftir ígripið. Stefið er notað þegar keyrt er undir DEBUG.

brot

Notkun: $s := \backslash\text{brot } x$ **Fyrir:** x er fleytitala ójöfn núlli.**Eftir:** s inniheldur brothluta fleytitölnnar x . Sjá lýsingu stefnsins fleytitala.

bætfylla

Notkun: $\text{bætfylla}(;g,x,y,l)$ **Fyrir:** g,x,y,l eru fjöldataölur.**Eftir:** x stendur fyrir kafla og y stendur fyrir vistfang. Stefið veldur því að bætunum frá og með $x:y$ til og með $x:(y+l-1)$ er gefið gildið g .

erfleyt

Notkun: $s := \backslash\text{erfleyt } x$ **Fyrir:** x er eitthvert gildi.**Eftir:** s fær gildið satt þá og því aðeins að x sé fleytitala, annars ósatt.

erfleytneikvæð

Notkun: $s := \backslash\text{erfleytneikvæð } x$ **Fyrir:** x er fleytitala.**Eftir:** s inniheldur sanngildi fyrir samanburðinn $x \ll 0$, þ.e. satt þá og því aðeins að x sé minna en 0, annars ósatt.

erfleytnúll

Notkun: $s := \backslash\text{erfleytnúll } x$

Fyrir: x er fleytitala.

Eftir: s inniheldur sanngildi fyrir samanburðinn $x == 0.0$, þ.e. satt þá og því aðeins að x sé fleytitalan 0.0, annars ósatt.

erhlunkur

Notkun: $s := \backslash\text{erhlunkur } x$

Fyrir: x er eitthvert gildi.

Eftir: s fær gildið satt þá og því aðeins að x sé hlunkur, annars ósatt.

erpar

Notkun: $s := \backslash\text{erpar } x$

Fyrir: x er eitthvert gildi.

Eftir: s fær gildið satt þá og því aðeins að x sé par, annars ósatt.

erstef

Notkun: $s := \backslash\text{erstef } x$

Fyrir: x er eitthvert gildi.

Eftir: s fær gildið satt þá og því aðeins að x sé stefgildi, annars ósatt.

erstrengur

Notkun: $s := \backslash\text{erstrengur } x$

Fyrir: x er eitthvert gildi.

Eftir: s fær gildið satt þá og því aðeins að x sé strengur, annars ósatt.

ertala

Notkun: $s := \backslash\text{ertala } x$

Fyrir: x er eitthvert gildi.

Eftir: s fær gildið satt þá og því aðeins að x sé fjölda- eða heiltala, annars ósatt.

ertóm

Notkun: $s := \backslash\text{ertóm } x$

Fyrir: x er eitthvert gildi.

Eftir: s fær gildið satt þá og því aðeins að x sé tómagildið, annars ósatt.

fdeiling

Notkun: `fdeiling(a,b;x,y,z)`

Fyrir: x, y og z eru fjöldatölur.

Eftir: ab (32 bita tala) er fjöldatöluútkoman þegar z er deilt í xy .

fjöltilfleyt

Notkun: `s := \fjöltilfleyt x`

Fyrir: x er fjöldatala.

Eftir: Stefið gefur s gildi fleytitölu sem samsvarar fjöldatölunni x .

fleytitala

Notkun: `s := fleytitala(;f,b,v)`

Fyrir: f er fjöldatala, annað hvort 0 eða 1. b er einhver fjöldatala. v er heiltala milli -1023 og 1023 að báðum meðtöldum.

Eftir: s er fleytitalan $(-1)^f(1 + b/65536)2^v$. Komast má að formerki, broti og veldi fleytitölu (f, b og v) með því að nota stefin formerki, brot, og veldi. Eftirfarandi jöfnur gilda:

```
\formerki fleytitala(;f,b,v) = f
\brot fleytitala(;f,b,v) = b
\veldi fleytitala(;f,b,v) = v
```

fleytmínus

Notkun: `s := \fleytmínus x`

Fyrir: x er fleytitala.

Eftir: Stefið gefur s gildi fleytitölnnar x með öfugu formerki.

fleyttilfjöl

Notkun: `s := \fleyttilfjöl x`

Fyrir: x er fleytitala.

Eftir: Stefið gefur s gildi fjöldatölu sem samsvarar fleytitölunni x .

flytja

Notkun: `flytja(;x,y,p,q,l)`

Fyrir: x, y, p, q, l eru fjöldatölur.

Eftir: x og p standa fyrir kafla, y og q standa fyrir vistföng. Stefið veldur því að l bæti frá og með $x:y$ eru flutt til $p:q$.

fmargfeldi**Notkun:** `fmargfeldi(a,b;x,y)`**Fyrir:** `x,y` eru fjöldatölur.**Eftir:** `ab` (32 bita tala) er margfeldi `x` og `y`.**fmismunur****Notkun:** `fmismunur(a,b;x,y)`**Fyrir:** `x,y` eru fjöldatölur.**Eftir:** `a` er mismunur `x` og `y`. `b` er 0 ef `x >=y` annars 1.**formerki****Notkun:** `s := \formerki(;x)`**Fyrir:** `x` er fleytitala ójöfn núlli.**Eftir:** Sjá lýsingu á stefinu fleytitala.**fsumma****Notkun:** `fsumma(a,b;x,y)`**Fyrir:** `x` og `y` eru fjöldatölur.**Eftir:** `ab` (32 bita tala) er summa `x` og `y`.**fylkissetja1****Notkun:** `x := fylkissetja1(;f,i,g)` eða
`x := f[i] := g`**Fyrir:** `f` er strengur eða hlunkur, `i` er sæti í `f`, `g` er hvaða gildi sem er ef `f` er hlunkur, bætigildi ef `f` er strengur.**Eftir:** Sæti `i` í `f` inniheldur `g`, `x` inniheldur `g`.**fylkissækja1****Notkun:** `x := fylkissækja1(;f,i)` eða
`x := f[i]`**Fyrir:** `f` er strengur eða hlunkur, `i` er sæti í `f`.**Eftir:** `x` inniheldur gildið í sæti `i` í `f`. Ef `f` var strengur þá er um bæti að ræða.**fylla****Notkun:** `fylla(;g,x,y,l)`**Fyrir:** `g,x,y,l` eru fjöldatölur.**Eftir:** `x` stendur fyrir kafla og `y` stendur fyrir vistfang. Stefið veldur því að bætunum frá og með `x:y` til og með `x:(y+l-1)` er gefið gildið `g`.

halasetja**Notkun:** `halasetja(;x,y)`**Fyrir:** x er par, y er eitthvað gildi.**Eftir:** Stefið veldur því að gildið y er sett í hala parsins x . Ef x er ekki par þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**hali****Notkun:** `s := \hali x`**Fyrir:** x er par.**Eftir:** s inniheldur hala parsins x . Ef x er ekki par þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**haus****Notkun:** `s := \haus x`**Fyrir:** x er par.**Eftir:** s inniheldur haus parsins x . Ef x er ekki par þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**haussetja****Notkun:** `haussetja(;x,y)`**Fyrir:** x er par, y er eitthvað gildi.**Eftir:** Stefið veldur því að gildið y er sett í haus parsins x . Ef x er ekki par þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**hábæti****Notkun:** `s := \hábæti x`**Fyrir:** x er fjölda- eða heiltala.**Eftir:** s inniheldur hábæti tölunnar x , þ.e. fyrra bætið sem táknar töluna.**hdeiling****Notkun:** `hdeiling(a,b;x,y,z)`**Fyrir:** x,y og z eru heiltölur.**Eftir:** ab (32 bita tala) er heiltöluútkoman þegar z er deilt í xy (32 bita tala).**hhliðra****Notkun:** `s := x \hhliðra y`**Fyrir:** x,y eru fjölda- eða heiltölur.**Eftir:** s inniheldur útkomuna þegar x er hliðrað til hægri um y bita.

hlunksetja**Notkun:** hlunksetja(x,y,g)**Fyrir:** x er hlunkur, y er fjöldatala og g er eitthvað gildi.**Eftir:** Stefið veldur því að gildið g er sett í y-ta sæti í hlunknum x. Ef x er ekki hlunkur þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**hlunkstærð****Notkun:** s := \hlunkstærð x**Fyrir:** x er hlunkur.**Eftir:** s inniheldur fjölda sæta í hlunknum x.**hlunksækja****Notkun:** s := x \hlunksækja y**Fyrir:** x er hlunkur og y er fjöldatala.**Eftir:** s inniheldur gildið í y-ta sæti í hlunknum x. Ef x er ekki hlunkur þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**hlunkur****Notkun:** s := \hlunkur x**Fyrir:** x er fjöldatala.**Eftir:** s er nýr hlunkur með a.m.k. x sætum, með öll sætin tóm. Ef ekki er hægt að úthluta hlunknum vegna þess að allt minni er upp urið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**hmargfeldi****Notkun:** hmarginfeldi(a,b;x,y)**Fyrir:** x,y eru heiltölur.**Eftir:** ab (32 bita tala) er margfeldi x og y.**hmismunur****Notkun:** hmismunur(a,b;x,y)**Fyrir:** x,y eru heiltölur.**Eftir:** a er mismunur x og y. b er 0 ef $x \geq y$, annars 1.**hsumma****Notkun:** hsumma(a,b;x,y)**Fyrir:** x og y eru heiltölur.**Eftir:** ab (32 bita tala) er summa x og y.

hætta**Notkun:** `hætta(;x)`**Fyrir:** `x` er heiltala.**Eftir:** Stefið veldur því að keyrsla forritsins stöðvast og tölunni `x` er skilað sem niðurstöðulykli (e. *exit code*) út í stýrikerfið.**innbæti****Notkun:** `s := \innbæti x`**Fyrir:** `x` er fjöldatala.**Eftir:** `s` inniheldur bæti sem lesið var úr hliði (e. *port*) `x`.**innfjöldi****Notkun:** `s := \innfjöldi x`**Fyrir:** `x` er stefgildi.**Eftir:** `s` inniheldur fjölda inn breyta fyrir stefgildið `x`.**innútfjöldi****Notkun:** `s := \innútfjöldi x`**Fyrir:** `x` er stefgildi.**Eftir:** `s` inniheldur fjölda innút breyta fyrir stefgildið `x`.**ígríp10****Notkun:** `ígríp10(ax,bx,cx,dx;)`**Fyrir:** `ax,bx,cx,dx` eru tölubreytur.**Eftir:** Stefið veldur því að gildin í breytunum eru sett í gistir AX, BX, CX, DX, því næst er kallað á BIOS kerfið með ígrípi (e. *interrupt*) 10. Gildin í gistunum eftir ígrípið eru síðan sett í samsvarandi breytur eftir kallið.**kafli****Notkun:** `s := \kafli x`**Fyrir:** `x` er strengur eða hlunkur.**Eftir:** `s` inniheldur tölu sem stendur fyrir kafla-hluta (e. *segment*) vistfangsins fyrir `x`.**kíkjabæti****Notkun:** `s := x \kíkjabæti y`**Fyrir:** `x,y` eru fjöldatölur.**Eftir:** `x` stendur fyrir kafla og `y` stendur fyrir vistfang. `s` inniheldur gildi bætisins í vistfangi `x:y`.

kíkjaorð

Notkun: `s := x \kíkjaorð y`

Fyrir: `x,y` eru fjöldataölur.

Eftir: `x` stendur fyrir kafla og `y` stendur fyrir vistfang. `s` inniheldur gildi orðsins í vistfangi `x:y`.

laus_K

Notkun: `s := laus_K(;)`

Fyrir:

Eftir: `s` inniheldur stærð ónotaðs minnis, mælt í K-um (1K = 1024 bæti).

lágbæti

Notkun: `s := \lágbæti x`

Fyrir: `x` er fjölda- eða heiltala.

Eftir: `s` inniheldur lágbæti tölunnar `x`, þ.e. seinna bætið sem táknar töluna.

lesastaf

Notkun: `s := lesastaf(;)`

Fyrir:

Eftir: `s` inniheldur tölu sem táknar staf er lesinn var frá lyklaborði.

minnka

Notkun: `s := \minnka x`

Fyrir: `x` er fjöldatala.

Eftir: `s` inniheldur gildið `x - 1`.

msdos

Notkun: `s :=msdos(ax,bx,cx,dx,bp,si,di,ds,es,flögg;)`

Fyrir: `ax,bx,cx,dx,bp,si,di,ds,es` og `flögg` eru breytur.

Eftir: Stefið veldur því að gildin í breytunum eru sett í gistur AX, BX, CX, DX, BP, SI, DI, og kafla DS, ES. Því næst er kallað á DOS kerfið með ígripi (e. *interrupt*) 21. Gildin í gistunum, köflunum og flöggin eftir ígripið eru síðan sett í samsvarandi breytur eftir kallið. Breyturnar eru ávallt af taginu tölur eftir kallið, óháð því hvaða tagi þær voru af fyrir kallið á stefið.

orðfylla

Notkun: `orðfylla(;g,x,y,l)`

Fyrir: `g,x,y` og `l` eru fjöldataölur.

Eftir: `x` stendur fyrir kafla og `y` stendur fyrir vistfang. stefið veldur því að orðin frá og með `x:y` til og með `x:(y+2(l-1))` fá gildið `g`.

RSskilti

Notkun: rofi1 := RSskilti(;rofi2)

Fyrir: rofi2 er eitthvert gildi.

Eftir: Stef þetta stýrir hvort gefin er vísbending á skjá þegar ruslasöfnun á sér stað. Í upphafi keyrslu hvers forrits er því þannig háttáð að meðan rusli er safnað eru settir stafirnir RS í efra horn skjásins vinstra megin. Þessari skipan mála má breyta með því að kalla á RSskilti með viðfangi sem er ósatt. Þá mun engin vísbending verða gefin þegar safnað er. Upphaflega skipan mála (þ.e.a.s. þá sem gildir í byrjun keyrslu) má fá aftur með því að kalla á RSskilti með sönnu viðfangi. Eftir kallið

```
rofi1 := RSskilti(;rofi2)
```

inniheldur rofi1 gildi sem segir til um skipan mála fyrir kallið, þ.a. þá skipan mála má fá aftur með kallinu

```
RSskilti(;rofi1)
```

safna

Notkun: safna(;

Fyrir: Ekkert.

Eftir: Kallið veldur því að ruslasöfnun á sér stað.

skalsafna

Notkun: rofi1 := skalsafna(;rofi2)

Fyrir: rofi2 er eitthvert gildi.

Eftir: Stef þetta stýrir ruslasöfnunum. Í upphafi keyrslu hvers forrits er ruslasöfnunum þannig háttáð að þær eru framkvæmdar eftir því sem þörf krefur. Í hvert skipti sem kallað er á kjarnastef sem úthlutar minni (þau eru strengur, hlunkur og :) er ruslasafnað ef þörf þykir á. Þessari skipan mála má breyta með því að kalla á skalsafna með viðfangi sem er ósatt. Þá mun aldrei verða ruslasafnað heldur verður minnið einfaldlega étið upp smátt og smátt nema kallað sé á grunnstefið safna. Upphaflega skipan mála (þ.e.a.s. þá sem gildir í byrjun keyrslu) má fá aftur með því að kalla á skalsafna með sönnu viðfangi. Eftir kallið

```
rofi1 := skalsafna(;rofi2)
```

inniheldur rofi1 gildi sem segir til um skipan mála fyrir kallið, þ.a. þá skipan mála má fá aftur með kallinu

```
skalsafna(;rofi1)
```

skrifastaf

Notkun: s := skrifastaf(;x)

Fyrir: x er 8 bita fjöldatala (0 - 255)

Eftir: Stefið veldur því að stafurinn, sem x táknar, er skrifaður í aðalúttak (e. *standard output*).

strengflytja

Notkun: `strengflytja(;s1,p1,s2,p2,n)`

Fyrir: `s1` og `s2` eru strengir. `p1` og `p2` og `n` eru fjöldatölur þ.a. `p1 ... p1 + n - 1` eru sæti í `s1` og `p2 ... p2 + n - 1` eru sæti í `s2`.

Eftir: Búið er að afrita bætinn í sætum `p1 ... p1 + n - 1` í `s1` í sæti `p2 ... p2 + n - 1` í `s2`.

strengfylla

Notkun: `strengsetjabæti(;gildi,strengur,staður,fjöldi)`

Fyrir: `strengur` er strengur, `gildi` er fjöldatala, `staður` og `fjöldi` eru fjöldatölur þ.a. `staður...staður+fjöldi-1` eru sæti í strengur.

Eftir: Sætin `staður...staður+fjöldi-1` í strengur hafa verið fyllt með bætinu í lágbæti `gildi`.

strengsetjabæti

Notkun: `strengsetjabæti(;x,y,g)`

Fyrir: `x` er strengur, `y` og `g` eru fjöldatölur.

Eftir: Stefið veldur því að `g` er sett í `y`-ta sæti í strengnum `x`. Ef `x` er ekki strengur þegar, kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.

strengsetjaorð

Notkun: `strengsetjaorð(;x,y,g)`

Fyrir: `x` er strengur, `y` er fjöldatala og `g` er fjölda- eða heiltala.

Eftir: Stefið veldur því að `g` er sett í `y`-ta sæti í strengnum `x`. Ef `x` er ekki strengur, þegar kallað er á stefið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.

strengstærð

Notkun: `s := \strengstærð x`

Fyrir: `x` er strengur.

Eftir: `s` inniheldur fjölda bæta í strengnum `x`.

strengsækjabæti

Notkun: `s := x \strengsækjabæti y`

Fyrir: `x` er strengur, `y` er fjöldatala.

Eftir: `s` inniheldur bætið í `y`-ta sæti strengsins `x`.

strengsækjaorð

Notkun: `s := x \strengsækjaorð y`

Fyrir: `x` er strengur, `y` er fjöldatala.

Eftir: `s` inniheldur orðið `y`-ta sæti strengsins `x`.

strengur**Notkun:** `s := \strengur x`**Fyrir:** `x` er fjöldatala.**Eftir:** `s` inniheldur nýjan streng af lágmarksstærð `x` bæti. Ef ekki er hægt að úthluta strengnum vegna þess að allt minni er upp urið, skrifast villumelding þess efnis á skjáinn og keyrsla forritsins stöðvast.**stækka****Notkun:** `s := \stækka x`**Fyrir:** `x` er fjöldatala.**Eftir:** `s` inniheldur gildið `x + 1`.**útbæti****Notkun:** `útbæti (; x, y)`**Fyrir:** `x` og `y` eru fjöldatölur.**Eftir:** Stefið veldur því að bætið `y` er skrifað í hlið (e. *port*) `x`.**veldi****Notkun:** `s := \veldi x`**Fyrir:** `x` er fleytitala ójöfn núlli.**Eftir:** Sjá lýsingu á stefnu fleytitala.**vhliðra****Notkun:** `s := x \vhliðra y`**Fyrir:** `x, y` eru fjölda- eða heiltölur.**Eftir:** `s` inniheldur útkomuna þegar tölunni `x` er hliðrað til vinstri um `y` bita.**vistfang****Notkun:** `s := \vistfang x`**Fyrir:** `x` er strengur eða hlunkur.**Eftir:** `s` inniheldur tölu sem stendur fyrir vistfangs-hluta (e. *offset*) vistfangsins fyrir `x`.

Kaflí 19

Einingin LESALINU

Þessi eining inniheldur eitt stef sem les línu af aðalinntaki.

lesalínu

Notkun: lína := lesalínu(;)

Fyrir: Ekkert.

ATH! Þar sem lesalínu notar biðminni við lestur af aðalinntaki, er ekki ráðlegt að nota aðrar aðgerðir við lestur úr aðalinntaki á sama tíma.

Eftir: Les eina línu úr aðal inntaki, sé endir inntaks ekki fundinn, þó aldrei fleiri en 255 stafir. lína fær gildi strengs sem inniheldur þá línu sem lesin var. Í sæti 0 í lína er að finna fjölda lesinna stafa (þ.e. lengd línunnar).

Ef komið er í enda inntaks, sem markast af því að lesið hefur verið ascii gildið 26 (Ctrl-z) eða ascii gildið 0 (NULL), fær lína gildið ósatt.

Endir línu er venjulega markaður af ascii gildunum 13 (carriage return) og 10 (linefeed) en lesalínu virkar þó endir línu sé eingöngu markaður með ascii gildinu 10. Þessi tvö gildi verða þó aldrei hluti innihaldsins í lína.

Ef línan sem verið er að lesa inn, er lengri en 255 stafir, eru einungis lesnir 255 stafir, þeim skilað í strengnum lína og afgangurinn látinn bíða næsta kalls á lesalínu.

Kaflí 20

Einingin SKRIFALIN

Þessi eining inniheldur eitt stef sem skrifar streng á aðalúttak.

skrifalínu

Notkun: `ok := skrifalínu(;strengur)`

Fyrir: strengur er strengur.

Eftir: Tekin er lengd(;strengur) (þ.e. bætið í sæti 0) og það látið segja til um þann fjölda bæta sem skrifa á. Þau bæti eru síðan skrifuð út í aðalúttak. Að lokum er svo bendir færður í næstu línu á eftir (með því að skrifa ascii gildin 13 og 10).

Ef tekst að skrifa þetta allt án þess að nokkur villa komi upp, fær ok gildið satt annars fær ok gildið ósatt.

Kaflí 21

Einingin SNUA

Einingin SNUA inniheldur aðgerð til að snúa við lista.

snúa

Notkun: `listi1 := snúa(;listi2)`

Fyrir: `listi2` er eiginlegur listi (tómur eða ekki) sem getur innihaldið hvað sem er. Listi er eiginlegur ef unnt er að fá tómagildið sem útkomu með því að taka hala nægilega oft.

Eftir: `listi1` fær gildi lista sem er þannig að búið er að snúa við röð staka í listanum `listi2` þ.a. fyrsta stakið í `listi2` verður það síðasta í `listi1` o.s.frv. Ef `listi2` er tómur (þ.e. []) þá verður `listi1` það líka.

Sem dæmi mætti taka að `snúa([1,2,3,4])` skilar `[4,3,2,1]`.

Kafli 22

Einingin STRENGIR

Einingin Strengir inniheldur nokkrar aðgerðir sem vinna á strengjum.

erstafurístreng

Notkun: `staður := erstafurístreng(;strengur, stafur)`

Fyrir: Strengur er strengur og stafur er stafur (þ.e. fjöldatala á bilinu 0 til 255).

Eftir: Leitað er í strengnum strengur af stafnum stafur, eins langt og lengd strengur (þ.e. bætið í sæti 0) segir til um. Ef stafurinn stafur finnst a.m.k. einu sinni í strengur, fær staður gildi fjöldatölu sem er sætisnúmerið þar sem stafur finnst fyrst í strengur. Ef stafur finnst ekki í strengur, fær staður gildið ósatt.

lengd

Notkun: `lengd := lengd(;strengur)`

Fyrir: Strengur er strengur.

Eftir: lengd fær gildi fjöldatölu sem er bætið sem geymt er í sæti 0 í strengur (þ.e. lengd hans).

strengskeyta

Notkun: `str3 := strengskeyta(;str1, str2)`

Fyrir: str1 og str2 eru strengir og samanlögð lengd þeirra er minni eða jöfn 255.

Eftir: str3 fær gildi strengs sem er útkoman þegar strengnum str1 er skeytt framan á strenginn str2. Innihaldi str1 og str2 er ekkert breytt.

hlutstrengur

Notkun: `hluti := hlutstrengur(;strengur, byrja, fjöldi)`

Fyrir: Strengur er strengur. byrja og fjöldi eru fjöldatölur.

Eftir: Ef byrja er stærri en lengd strengur, fær hluti gildið ósatt. Ef byrja að viðbættum fjöldi - 1 er stærri en lengd strengur, fær hluti líka gildið ósatt. Að öðrum kosti fær hluti gildi strengs sem er hlutstrengur úr strengur, frá og með sæti byrja til og með sæti byrja+fjöldi-1 þ.e. hluti verður strengur með lengd fjöldi.

strengurístreng

Notkun: `strengurístreng(;frástrengur,tilstrengur,staður)`

Fyrir: frástrengur og tilstrengur eru strengir. staður er fjöldatala.

Eftir: Ef staður er stærri en lengd tilstrengur er ekkert gert. Ef staður að viðbættri lengd frástrengur -
ler stærri en lengd tilstrengur er ekkert heldur gert. Að öðrum kosti er innihald frástrengur flutt yfir
í tilstrengur þ.a. fyrsta bætíð í sæti 1 í frástrengur fer í sæti staður í tilstrengur, bætíð í sæti 2 í
frástrengur fer í sæti staður + 1 í tilstrengur o.s.frv. Innihaldi frástrengur er ekkert breytt.

Kaflí 23

Einingin UT

nýlína

Notkun: nýlína(;))

Fyrir: Ekkert.

Eftir: Stefið veldur því að vendistafur (e. *carriage return*) og línuskipti stafur (e. *linefeed*) eru skrifaðir út í aðal úttak (e. *standard output*). Þannig að það sem skrifað er næst út í aðal úttak, lendir fremst í næstu línu fyrir neðan.

skrifa

Notkun: skrifa(;x)

Fyrir: x er gildi sem er listi, tómagildið, heiltala, strengur, fleytitala eða hlunkur.

Eftir: Stefið veldur því að gildið x er skrifuð út í aðal úttak (e. *standard output*), ef það er einhver af þeim gildum sem taldar voru upp hér að framan. Ef x er ekki slíkt gildi þá skrifast út stafurinn ?. Hlunkar eru skrifaðir umluktir slaufusvigum.

skrifafjöl

Notkun: skrifafjöl(;x)

Fyrir: x er fjöldatala.

Eftir: Stefið veldur því að fjöldatalan x er skrifuð út í aðal úttak (e. *standard output*).

skrifastreng

Notkun: skrifastreng(;x)

Fyrir: x er strengur. Bætið í 0. sæti í strengnum, segir til um hversu langur strengurinn er.

Eftir: Ef x er strengur, þá er hann skrifaður út í aðal úttak (e. *standard output*). Strengurinn er skrifaður án umlykjandi gæsalappa. Ef x er ekki strengur þá er stafurinn ? skrifaður út í aðal úttak.

Hluti III
Viðaukar

Kafli 24

Forritið Fjalla

Forritið Fjalla er hjálparforrit fyrir þá sem forrita í Fjölni. Forritið les einingar sem búnar hafa verið til með Fjölni og skrifar út ýmsar upplýsingar um viðkomandi einingar. Forritið er keyrt í DOS umhverfinu með því að gefa skipanalínu á forminu:

```
A>FJALLA[/rofar] ein [/rofar ein]...
```

Nánari lýsing á rofar og ein fer hér á eftir.

Rofar í Fjalla Rofunum er stjórnað í skipanalímunni eins og í Fjölni. Í upphafi eru rofarnir u og i sjálfgefnir.

Rofi u: Atriði sem einingin flytur út eru skrifuð út. Ef einingin flytur út einhverjar faldar grunnaðgerðir (sjá umfjöllun um eininguna FELAGRUN) eru þær ekki skrifaðar út þegar u rofinn er á, nema að g rofinn sé einnig á. Atriðin geta verið hvort heldur sem er, stef eða víðværar breytur.

Rofi i: Atriði sem einingin flytur inn eru skrifuð út. Ef einingin flytur inn einhverjar grunnaðgerðir eru þær ekki skrifaðar út þegar i rofinn er á, nema að g rofinn sé einnig á. Atriðin geta verið hvort heldur sem er, stef eða víðværar breytur.

Rofi g: Ef einingin flytur inn og/eða út einhverjar faldar aðgerðir eru þær skrifaðar út.

Rofi t: Minnstilvísanir eru skrifaðar út fyrir öll þau atriði sem lýst er.

Rofi r: Upplýsingar um fjölda og heildarstærð strengja talin í bætum eru skrifaðar út.

Rofi b: Upplýsingar um fjölda óbundinna víðværra breytna er skrifaður út. Þær breytur sem uppnefna aðra breytu teljast sem ein breyta með breytunni sem þær uppnefna.

Rofi s: Upplýsingar um stærð minnisblokkar einingarinnar talin í bætum eru skrifaðar út.

Rofarnir eru virkir frá því að þeir eru gerðir virkir í skipanalímunni allt til enda hennar, ef ekki er slökkt á þeim áður. Hægt er að slökkva á rofum með því að setja '-' táknið fyrir aftan þá.

Dæmi: A>Fjalla/si- ein1 /u-i ein2.ein

Í þessu dæmi skrifast út útflutt atriði, uppnefningar og stærð minnisblokkar fyrir ein1. Fyrir ein2 skrifast út innflutt atriði, uppnefningar og stærð minnisblokkar.

Einingar í fjalla Hægt er að tiltaka eina einingu í senn, sem forritið á að vinna á, með eða án eftirnafns.

Dæmi: A>Fjalla ein1 ein2.ein

Hér skrifast út útflutt, innflutt atriði og uppnefningar fyrir einingarskrána ein1.ein Fyrir ein2.ein skrifast út það sama.

Dæmi um úttak úr Fjalla Ef kallað er á forritið þannig:

```
A>FJALLA/gtrbs innut2
```

þá má búast við úttaki á þennan veg:

```
Fjöldnir Íslensk útgáfa 1.00 Raðnúmer 0000001
Copyright (c) 1986, 1987, 1988 Öll réttindi áskilin
Snorri Agnarsson, Jón Harðarson, Páll Björnsson
Lýsi einingunni innut2.EIN
```

Útflutningur:

```
lesa [stef 0 0] [0]
lesalínu [stef 0 0] [0]
```

Uppnefningar:

```
snúa -> snúningur
```

Innflutningur:

```
(+) [stef 0 0] [1952 1952]
_eftirmáli0 máli [3798]
_formáli0 máli [3170]
*** [stef 0 2] [2710]
```

Breytur:

```
Eru 2 talsins
```

Strengir:

```
Eru 1 og eru 129 bæti alls á lengd.
```

Stærð þulu í minni: 3904

Fyrsta línan segir til um útgáfu forritsins og raðnúmer þess. Næstu tvær línur eru síðan til upplýsingar um forritið. Því næst kemur lína sem segir til um hvaða einingu er verið að lýsa.

Útflutningar eru því næst taldir upp í stafrófsröð. Hver lína í upptalningunni er þ.a. fyrst kemur nafn atriðisins, því næst kemur tagið á atriðinu, ef það er stef þá segir fyrri talan til um fjölda innút breytna og seinni talan segir til um fjölda inn breytna. Þar á eftir kemur tala sem segir til um vistfang stefnsins.

Upptalning á uppnefningum kemur því næst. Hver lína sýnir uppnefninguna eins og hún er í einingunni.

Upptalning á innflutningum kemur þar á eftir, formið á upptalningunni er eins og fyrir útflutninginn.

Upplýsingar um fjölda víðværra breytna er þar á eftir.

Þar á eftir er upptalning á fjölda strengja og samanlögð lengd þeirra gefin upp.

Að lokum stærð minnisblokkarinnar undir þau stef sem einingin inniheldur.

Kaflí 25

Forritið Hvarer

Forritið Hvarer er hjálparforrit fyrir þá sem forrita í Fjölni. Forritið les einingar sem búnar hafa verið til með Fjölni og leitar að atriðum meðal útfluttra og/eða innfluttra atriða eininganna. Forritið skrifar í lok keyrslu út á skjá lista yfir einingar sem viðkomandi atriði fundust í inn- og/eða útflutningi. Forritið er keyrt í DOS umhverfinu með því að gefa skipanalínu á forminu:

```
A>HVARER atriði
```

rofar einingar

Nánari lýsing á atriði, rofar og einingar fer hér á eftir.

Atriði í Hvarer

Nöfn þeirra út- og/eða innflutninga sem leita á að, eru tilgreind í atriði. Eru nöfnin slegin inn eins og þau koma fyrir í einingunum sem leita á í. Ein undantekning er þó sú, að er sleppt framan af nafninu, ef það er til staðar. Þar sem við á geta atriði átt bæði við nöfn á stefjum og viðværum breytum.

Rofar í Hvarer

Rofar í Hvarer forritinu eru tvenns konar: i og u. Rofinn u er sjálfgefinn.

Rofi Áhrif

u Leitað að atriði/atriðum í útfluttum atriðum eininga.

i Leitað að atriði/atriðum í innfluttum atriðum eininga.

Einingar í Hvarer

Einingar sem leita á í eru tilteknaðar með nöfnum. Ekki þarf að gefa upp eftirnafn eininganna. Einnig er hægt að tilgreina ákveðið safn eininga sem leita á í. Til dæmis væri hægt að skrifa: `\ein\b*.ein`. Þá er leitað í öllum einingum með nafni sem byrjar á stafnum b og með eftirnafninu .ein í möppunni ein sem liggur út frá rótinni.

Dæmi: `A>Hvarer lesa + taka \iu- ein1 ein2 f*.ein`

Hér er leitað að lesa, + og taka í innfluttum atriðum eininganna ein1, ein2 og allra þeirra einingaskráa sem hafa nafn sem samsvarar f*.ein.

Dæmi um úttak úr Hvarer

Ef kallað er á forritið þannig:

```
A>Hvarer --- - + \i ein\grunnur ein\inn
```

þá má búast við úttaki á þennan veg:

```
Fjölnir Íslensk útgáfa 1.00 Raðnúmer 0000001
```

```
Copyright (c) 1986, 1987, 1988 Öll réttindi áskilin
```

```
Snorri Agnarsson, Jón Harðarson, Páll Björnsson
```

```
Leita að: [+ - ---]
```

```
í: [ein\inn ein\grunnur]
```

```
Eftirtöld atriði fundust í útflutningi:  
    "+"   í eftirtöldum einingum:  
           [ein\grunnur.ein]  
    "-"   í eftirtöldum einingum:  
           [ein\grunnur.ein]
```

```
Eftirtöld atriði fundust í innflutningi:  
    "+"   í eftirtöldum einingum:  
           [ein\inn.ein]
```

Fyrsta línan segir til um útgáfu foritsins og raðnúmer þess. Næstu tvær línur eru síðan til upplýsingar um forritið. Í fjórðu línunni er síðan tekið fram hverju er verið að leita að í einingunum. Í fimmtu línu er tekið fram í hvaða einingum skal leita að atriðunum. Þar á eftir koma síðan upplýsingar um í hvaða einingum atriðin fundust, ef þau fundust þá yfirleitt. Ef viðkomandi atriði finnst, hvort sem er í útflutningi eða innflutningi, þá kemur lína sem segir hvert atriðið er og þar á eftir upptalning á einingunum sem atriðið fannst í.

Kafli 26

Einingin BRÖT

Einingin BRÖT inniheldur aðgerðir á brot. Einingin er fjölnota þar eð notandinn getur ákveðið hvers konar gildi eru notuð í stuðla (teljara og nefnara) með því að tengja viðeigandi stuðlaaðgerðir einingunni.

Útflutningar

*

Notkun: $x := y * z$

Fyrir: y og z eru brot.

Eftir: z er brotið sem er margfeldi y og z .

+

Notkun: $x := y + z$

Fyrir: y og z eru brot.

Eftir: z er brotið sem er summa y og z .

-

Notkun: $x := y - z$

Fyrir: y og z eru brot.

Eftir: z er brotið sem er mismunur y og z .

/

Notkun: $x := y / z$

Fyrir: y og z eru brot.

Eftir: z er brotið sem er kvóti y og z .

<

Notkun: $x := y < z$ **Fyrir:** y og z eru brot.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

<=

Notkun: $x := y <= z$ **Fyrir:** y og z eru brot.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

<>

Notkun: $x := y <> z$ **Fyrir:** y og z eru brot.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

=

Notkun: $x := y = z$ **Fyrir:** y og z eru brot.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

>

Notkun: $x := y > z$ **Fyrir:** y og z eru brot.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

>=

Notkun: $x := y >= z$ **Fyrir:** y og z eru brot.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.**brot****Notkun:** $x := y \backslash \text{brot } z$ **Fyrir:** y og z eru stuðlar, y er ekki núll.**Eftir:** x er brotið y/z . Tryggt er að fyrir alla stuðla y og z þ.a. z er ekki núll gildir að $y/z = (\backslash \text{teljari } x) / (\backslash \text{nefnari } x)$ og að $\backslash \text{nefnari } x > 0$ og að stuðlarnir $\backslash \text{teljari } x$ og $\backslash \text{nefnari } x$ hafa engan þátt sameiginlegan.

nefnari**Notkun:** $x := \backslash\text{nefnari } y$ **Fyrir:** y er brot.**Eftir:** z er nefnari fullstytts brots jafnt y . Sjá einnig brot.**teljari****Notkun:** $x := \backslash\text{teljari } y$ **Fyrir:** y er brot.**Eftir:** z er teljari fullstytts brots jafnt y . Sjá einnig brot.**Innflutningar**

Auk falinna kjarnastefja er vísað í eftirfarandi stef úr einingunni BROT.

*

Notkun: $x := y * z$ **Fyrir:** y og z eru stuðlar.**Eftir:** x er stuðullinn sem er margfeldi y og z .

+

Notkun: $x := y + z$ **Fyrir:** y og z eru stuðlar.**Eftir:** x er stuðullinn sem er summa y og z .

-

Notkun: $x := y - z$ **Fyrir:** y og z eru stuðlar.**Eftir:** x er stuðullinn sem er mismunur y og z .

/

Notkun: $x := y / z$ **Fyrir:** y og z eru stuðlar.**Eftir:** x er stuðullinn sem er kvótíy og z .

<

Notkun: $x := y < z$ **Fyrir:** y og z eru stuðlar.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

<>

Notkun: $x := y <> z$ **Fyrir:** y og z eru stuðlar.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

=

Notkun: $x := y = z$ **Fyrir:** y og z eru stuðlar.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.

>

Notkun: $x := y > z$ **Fyrir:** y og z eru stuðlar.**Eftir:** z er niðurstaða samanburðarins, satt eða ósatt.**gcd****Notkun:** $x := y \text{ gcd } z$ **Fyrir:** y og z eru stuðlar.**Eftir:** z er stuðullinn sem er stærsti sameiginlegi þáttur y og z .**núll****Notkun:** $x := \text{núll}()$ **Fyrir:** Ekkert.**Eftir:** x er stuðullinn núll.

Kafli 27

Einingarnar FELAGRUN og FELAKJAR

Einingarnar FELAGRUN og FELAKJAR hafa nokkra sérstöðu. Þær eru notaðar til þess að festa innflutt stef sem grunnstef og kjarnastef, þ.e.a.s. stef úr einingunum GRUNNUR og KJARNI. Ef t.d. einingin "A" vísar í grunnstef "+" og "-" þá má smíða nýja einingu "B" með einingarveitingunni

```
"B" = "A" * "FELAGRUN" ;
```

Þessi nýja eining vísar nú ekki lengur í "+" og "-" heldur í "(+)" og "(-)". Einingin "GRUNNUR" inniheldur stef með þessum nöfnum, og eru það sömu stef og "+" og "-". Þar eð aðeins einingarnar GRUNNUR og KJARNI innihalda "(+)" og "(-)" og ekki er mögulegt að uppnefna "(+)" og "(-)" er tryggt að stef þessi tengjast grunnstefjunum eða kjarnastefjunum, sem reyndar eru þau sömu.

Einingin FELAGRUN er reyndar ekkert annað en uppnefning á öllum grunnstefjum sem gefur þeim nöfn með svigum, þ.e.a.s.:

```
"FELAGRUN" =  
{  
!      -> (!)  
%      -> (%)  
.      .  
.      .  
.      .  
vhliðra -> (vhliðra)  
vistfang -> (vistfang)  
} ;
```

Slíka einingu er ekki hægt að skrifa í Fjölni þar eð svigar eru ekki leyfilegir í nöfnum. Svipað gildir um eininguna FELAKJAR.

Kafli 28

Einingar fyrir fylki

Einingarnar HFYDKI og SFYDKI innihalda stef fyrir fylki með talnavísun. Fylki þau sem HFYDKI býður upp á eru hraðvirkari en þau sem SFYDKI býður, en nota meira minni. Báðar gerðir fylkja stækka sjálfkrafa þegar bætt er í þau og leyfa vísa frá 0 til 65535.

Athugið að hver og einn getur hannað sín eigin fylki og þarf alls ekki endilega að nota þessar einingar. T.d. má oft notast við hlunka eða strengi sem fylki, og einnig má auðveldlega smíða fylkjaeiningu með hjálp einingarinnar TAFLA.

Einingar þessar vísa aðeins í kjarnastef.

28.1 Einingin HFYDKI

fylki

Notkun: `f := fylki`

Fyrir: Ekkert.

Eftir: `f` inniheldur fylki.

fylkissetja1

Notkun: `s := fylkissetja1(;f,i,g)`

Fyrir: `f` er fylki, `i` er tala (heiltala eða fjöldatala), `g` er eitthvert gildi.

Eftir: Sætið `i` í fylkinu `f` inniheldur gildið `g`. `s` inniheldur gildið `g`.

fylkissækja1

Notkun: `s := fylkissækja1(;f,i)`

Fyrir: `f` er fylki, `i` er tala (heiltala eða fjöldatala).

Eftir: `s` inniheldur stakið úr sæti `i` í fylkinu `f`. Ef ekkert stak hefur verið sett í sæti `i` þá inniheldur `s` tómagildið, `||`.

28.2 Einingin SFYLKI

fylki

Notkun: `f := fylki`

Fyrir: Ekkert.

Eftir: `f` inniheldur fylki.

fykkissetja1

Notkun: `s := fykkissetja1(;f,i,g)`

Fyrir: `f` er fylki, `i` er tala (heiltala eða fjöldatala), `g` er eitthvert gildi.

Eftir: Sætið `i` í fylkinu `f` inniheldur gildið `g`. `s` inniheldur gildið `g`.

fykkissækja1

Notkun: `s := fykkissækja1(;f,i)`

Fyrir: `f` er fylki, `i` er tala (heiltala eða fjöldatala).

Eftir: `s` inniheldur stakið úr sæti `i` í fylkinu `f`. Ef ekkert stak hefur verið sett í sæti `i` þá inniheldur `s` tómagildið, `[]`.

max

Notkun: `s := max(;f)`

Fyrir: `f` er fylki.

Eftir: `s` inniheldur stærsta fjöldatöluvísi í `f`. Sætið `s` í `f` inniheldur gildi annað en tómagildið, en öll sæti þar fyrir ofan eru tóm.

min

Notkun: `s := min(;f)`

Fyrir: `f` er fylki.

Eftir: `s` inniheldur minnsta fjöldatöluvísi í `f`. Sætið `s` í `f` inniheldur gildi annað en tómagildið, en öll sæti þar fyrir neðan eru tóm.

Kafli 29

Einingin HEXEININ

Þessi eining inniheldur stef sem koma að góðum notum þegar skrifa þarf heiltölur og fleytitölur út í sextánda-kerfi.

Útflutningar

hexbæti

Notkun: `str := hexbæti(;tala)`

Fyrir: tala er fjöldatala.

Eftir: `str` fær gildi strengs sem hefur lengd 3 og inniheldur "\$xy" þar sem `xy` er fjöldatalan `tala` (eða `lágbæti(;tala)` ef `tala` er > 255) táknuð á hex-sniði. T.d. fengi `str` gildið "\$1B" ef kallað væri

```
str := hexbæti(;27)
```

hexorð

Notkun: `str := hexorð(;tala)`

Fyrir: tala er fjöldatala.

Eftir: `str` fær gildi strengs sem hefur lengd 5 og inniheldur "\$wxyz" þar sem `wxyz` er fjöldatalan `tala` táknuð á hex-sniði. T.d. fengi `str` gildið "\$001B" ef kallað væri

```
str := hexbæti(;27)
```

hexstafur

Notkun: `stafur := hexstafur(;tala)`

Fyrir: tala er fjöldatala á bilinu 0 til 15.

Eftir: `stafur` fær gildi fjöldatölu sem inniheldur í lágbæti sínu stafinn sem notaður er til að tákna tala á hexformi. T.d. fengi `stafur` gildið 'A' ef kallað væri `stafur := hexstafur(;10)`.

Innflutningar

Auk kjarnastefja er vísað í eftirfarandi stef úr einingunni HEXEININ.

afrita

Notkun: `str2 := afrita(;str1)`

Fyrir: `str1` er strengur.

Eftir: `str2` hefur gildi strengs sem er afrit af strengnum `str1`.

Stef sem uppfyllir þessa lýsingu má finna í einingunni AFRITA og í einingunni GRUNNUR.

Kafli 30

Einingin ISLRÖÐ

Eining þessi inniheldur stef til að bera saman stafi og strengi samkvæmt íslenskri stafrófsröð. Úr henni er einungis vísað í falin kjarnastef.

íslenskriröð

Notkun: `svar := íslenskriröð(;stafur1,stafur2)`

Fyrir: `stafur1` og `stafur2` eru bæði fjöldatölur á bilinu 0 til 255.

Eftir: Þetta stef gefur einkvæma röðun fyrir íslenska stafrófið. `svar` fær gildið `satt` ef `stafur1` er á undan `stafur2` (eða `stafur1 = stafur2`) í íslenska stafrófinu, annars fær `svar` gildið `ósatt` (þ.e. ef `stafur1` er á eftir `stafur2` í íslenska stafrófinu).

íslsamanb

Notkun: `svar := íslsamanb(;strengur1,strengur2)`

Fyrir: `strengur1` og `strengur2` eru strengir.

Eftir: `svar` fær gildi stafs (þ.e. fjöldatölu) sem segir til um innbirðis röð `strengur1` og `strengur2` í íslenskri stafrófsröð. `svar` fær gildið `'<'` ef `strengur1` er á undan `strengur2` í röðinni. `svar` fær gildið `'='` ef `strengur1` og `strengur2` eru eins (þ.e. á sama stað í röðinni). `svar` fær gildið `'>'` ef `strengur2` er á undan `strengur1` í röðinni.

Kafli 31

Einingin LYKLAR

Þessi eining inniheldur stef sem koma að góðum notum þegar þörf er á nákvæmum upplýsingum um innslátt frá lyklaborði.

skiptistaða

Notkun: `s := skiptistaða(;`

Fyrir: Ekkert.

Eftir: `s` er bæti sem táknar stöðu skiptihnappa á lyklaborðinu. Bitar í `s` settir (1) gefa upplýsingar um skiptiástand samkvæmt eftirfarandi töflu:

Biti settur Ástand

0 ýtt á hægri skiptihnapp (shift)

1 ýtt á vinstri skiptihnapp

2 ýtt á Ctrl hnapp

3 ýtt á Alt hnapp

4 Scroll lock á

5 Num lock á

6 Caps lock á

7 Insert á

varýttástaf

Notkun: `s := varýttástaf(;`

Fyrir: Ekkert.

Eftir: Stefið athugar hvort einhver stafur er í biðminni (e. *buffer*) lyklaborðsins, ef svo er þá er sá stafur lesinn og `s` fær fjöldatölugildi. Ef enginn stafur bíður í biðminninu fær `s` gildið ósatt. Ef stafur var lesinn er hann táknður með tveimur bætum sem geymd eru í `\hábæti s` og `\lágþæti s`. Ef `\lágþæti s` er ekki 0, þá hefur verið ýtt á venjulegan staf og hefur `\lágþæti s` gildi sem er samsvarandi stafnum í ASCII táknrófinu. Ef `\lágþæti s` er 0 þá hefur hinsvegar verið ýtt á aðgerðartakka á lyklaborðinu. `\hábæti s` inniheldur þá gildi sem segir til um hvaða aðgerðartakka var ýtt á. Taflan hér að neðan sýnir gildi `\hábæti s` miðað við hvaða takka var ýtt á.

Ef `\hábæti s` er: Þá hefur verið ýtt á:

3 NULL (þ.e. Alt-0)
 15 Shift-Tab
 16-25 Alt-Q,W,E,R,T,Y,U,I,O,P
 30-38 Alt-A,S,D,F,G,H,J,K,L
 44-50 Alt-Z,X,C,V,B,N,M
 59-68 F1-F10
 71 Home
 72 Ör upp
 73 PgUp
 75 Ör til vinstri
 77 Ör til hægri
 79 End
 80 Ör niður
 81 PgDn
 82 Ins
 83 Del
 84-93 F11-F20 (Shift F1-Shift F10)
 94-103 F21-F30 (Ctrl F1-Ctrl F10)
 104-113 F31-F40 (Alt F1-Alt F10)
 114 Ctrl-PrtScr
 115 Ctrl-Ör til vinstri
 116 Ctrl-Ör til hægri
 117 Ctrl-End
 118 Ctrl-PgDn
 119 Ctrl-Home
 120-131 Alt-1,2,3,4,5,6,7,8,9,0,-,=
 132 Ctrl-PgUp

ýttástaf

Notkun: `s := ýttástaf(;)`

Fyrir: Ekkert.

Eftir: Ef einhver stafur er í biðminni (e. *buffer*) lyklaborðsins er sá stafur lesinn. Ef enginn stafur er í biðminninu þá er beðið eftir því að ýtt sé á staf á lyklaborðinu. Að öðru leyti gildir lýsingin hér að framan fyrir stefið varýttástaf.

Kaflí 32

Einingin QUICKSOR

Einingin QUICKSOR inniheldur stef sem raðar fylkjum. Aðferðin sem notuð er er svokölluð Quicksort aðferð sem fundin var upp af C.A.R. Hoare. Einingin er fjölnota þar eð úr henni er vísað í samanburðaraðgerðir "=" og « "sem notandi einingarinnar getur tengt við hvaða samanburðaraðgerðir sem hann vill. Einnig má tengja hvers konar fylkjaaðgerðir sem notandinn vill.

Útflutningar

raða

Notkun: raða(;f,min,max)

Fyrir: fer fylki, min og max eru fjöldatölur.

f inniheldur a.m.k. stökin f[min] til f[max-1].

Eftir: Svæðinu f[min] til f[max-1] hefur verið raðað í vaxandi röð.

Innflutningar

Úr einingunni QUICKSOR er vísað í stefin fylkissækja1 og fylkissetja1 (sjá kaflann um fylkissegðir). Auk þess er vísað í eftirfarandi stef:

<

Notkun: s := x < y

Fyrir: x og y eru stök af því tagi sem fylkið f inniheldur.

Eftir: s inniheldur sanngildi samanburðarins.

=

Notkun: s := x = y

Fyrir: x og y eru stök af því tagi sem fylkið f inniheldur.

Eftir: s inniheldur sanngildi samanburðarins.

Kaflí 33

Einingin MARGLID

Einingin MARGLID inniheldur stef til að reikna með margliður. Einingin er fjölnota þar eð notandi hennar getur sjálfur ákveðið hvers konar stuðla hann notar með því að tengja viðeigandi aðgerðir á stuðla. Margliðurnar er geymdar sem listar stuðla. Margliðan $a_0x^0 + \dots + a_nx^n$ er þannig geymd sem listinn $[a_0, \dots, a_n]$. Það skilyrði er sett að stuðullinn a_n sé ekki núll. Margliðan núll er því geymd sem tómi listinn $[]$.

Útflutningar

*

Notkun: $s := x * y$

Fyrir: x og y eru margliður.

Eftir: s inniheldur margliðuna sem er margfeldi x og y .

+

Notkun: $s := x + y$

Fyrir: x og y eru margliður.

Eftir: s inniheldur margliðuna sem er summa x og y .

-

Notkun: $s := x - y$

Fyrir: x og y eru margliður.

Eftir: s inniheldur margliðuna sem er mismunur x og y .

=

Notkun: $s := x = y$

Fyrir: x og y eru margliður.

Eftir: s inniheldur sanngildi samanburðarins, satt eða ósatt.

skrifamargliðu**Notkun:** skrifamargliðu(;x)**Fyrir:** x er margliða.**Eftir:** Margliðan x hefur verið skrifuð á læsilegan hátt á aðalúttak.**Innflutningar**

*

Notkun: $s := x * y$ **Fyrir:** x og y eru stuðlar.**Eftir:** s inniheldur stuðulinn sem er margfeldi x og y.

+

Notkun: $s := x + y$ **Fyrir:** x og y eru stuðlar.**Eftir:** s inniheldur stuðulinn sem er summa x og y.

-

Notkun: $s := x - y$ **Fyrir:** x og y eru stuðlar.**Eftir:** s inniheldur stuðulinn sem er mismunur x og y.

=

Notkun: $s := x = y$ **Fyrir:** x og y eru stuðlar.**Eftir:** s inniheldur sanngildi samanburðarins, satt eða ósatt.**núllstuðull****Notkun:** $s := \text{núllstuðull}()$ **Fyrir:** Ekkert.**Eftir:** s inniheldur stuðulinn núll.**skrifastuðul****Notkun:** skrifastuðul(;x)**Fyrir:** x er stuðull.**Eftir:** Stuðullinn x hefur verið skrifaður á læsilegan hátt á aðalúttak.

Kafli 34

Einingin MENGI

Einingin MENGI inniheldur stef til að vinna með röðuð mengi. Einingin er fjölnota þar eð notandinn getur valið hvers konar stök geta verið í mengjunum með því að tengja viðeigandi samanburðaraðgerðir.

Útflutningar

*

Notkun: $s := x * y$

Fyrir: x og y eru mengi.

Eftir: s inniheldur sniðmengi x og y .

+

Notkun: $s := x + y$

Fyrir: x og y eru mengi.

Eftir: s inniheldur sammengi x og y .

++

Notkun: $s := x ++ y$

Fyrir: x er mengi, y er stak.

Eftir: s inniheldur sammengi x og y .

-

Notkun: $s := x - y$

Fyrir: x og y eru mengi.

Eftir: s inniheldur mengjamismun x og y .

\leq

Notkun: $s := x \leq y$

Fyrir: x og y eru mengi.

Eftir: s inniheldur satt ef x er hlutmengi y , annars ósatt.

$=$

Notkun: $s := x = y$

Fyrir: x og y eru mengi.

Eftir: s inniheldur satt ef x er jafnt y , annars ósatt.

min

Notkun: $s := \min x$

Fyrir: x er mengi, ekki tómt.

Eftir: s inniheldur minnsta stak x .

rest

Notkun: $s := \text{rest } x$

Fyrir: x er mengi.

Eftir: s inniheldur mengi sem inniheldur allt nema minnsta stak x . Ef x er tómt þá er s tómt.

tómt

Notkun: $s := \text{tómt}()$

Fyrir: Ekkert

Eftir: s inniheldur tómt mengi.

Innflutningar

$<$

Notkun: $s := x < y$

Fyrir: x og y eru stök.

Eftir: s inniheldur satt ef x er minna en y , annars ósatt.

$=$

Notkun: $s := x = y$

Fyrir: x og y eru stök.

Eftir: s inniheldur satt ef x er jafnt y , annars ósatt.

Kaflí 35

Einingin RADA

Einingin RADA inniheldur stef til að raða listum. Einingin er fjölnota þar eð notandinn getur sjálfur ákveðið hvers konar röð notuð er með því að tengja hina innfluttu samanburðaraðgerð viðeigandi stefi.

Útflutningar

mergja

Notkun: `s := x \mergja y`

Fyrir: `x` og `y` eru raðaðir listar þannig að t.d. ef `x = [x1, . . . xn]` þá gildir fyrir öll `i` þ.a. `i < n` að `xiíröðxi+1`.

Eftir: `s` inniheldur mergjun listanna `x` og `y`, þ.e.a.s. `s` inniheldur öll stök úr `x` og `y` í röð.

raða

Notkun: `s := \raða x`

Fyrir: `x` er listi sem inniheldur stök af því tagi sem stefið íröð vinnur á.

Eftir: `s` inniheldur lista sömu staka og `x`, í röð.

skipta

Notkun: `skipta(y,z;x)`

Fyrir: `x` er listi.

Eftir: `y` inniheldur u.þ.b. helming stakanna úr `x` og `z` inniheldur hinn helminginn.

Innflutningar

íröð

Notkun: `s := x \íröð y`

Fyrir: `x` og `y` eru einhver gildi af því tagi sem íröð er ætlað að vinna á.

Eftir: `s` inniheldur satt eða ósatt eftir því hvort gildin eru í vaxandi röð.

Athugasemdir: íröð skyldi tengja við samanburðaraðgerð sem vinnur svipað og `<` eða `ó`.

Kaflí 36

Einingin SKJAR

Hér eru nokkur stef sem nota ígrip10 í BIOS kerfinu til að stjórna aðgerðum á skjánum. Í allri meðhöndlun á skjánum er litið á hann sem hnitakerfi þ.a. stafurinn efst í vinstra horninu, hefur hnitin (1,1) (lína 1 og dálkur 1). Skjáblaðsíður (e. *display page*) eru númeraðar 0 til 7 fyrir 40 dálka línur en 0 til 3 fyrir 80 dálka línur.

depilgerð

Notkun: depilgerð(byrjalínu, endalínu; blaðsíða)

Fyrir: blaðsíða er fjöldatala sem segir til um á hvaða skjáblaðsíðu skal finna gerð depilsins (e. *cursor*).

Eftir: byrjalínu og endalínu innihalda fjöldatölur sem segja til um gerð depilsins á blaðsíðunni blaðsíða (sbr. stefið setjadedil hér að neðan).

depilstaða

Notkun: depilstaða(lína, dálkur; blaðsíða)

Fyrir: blaðsíða er fjöldatala sem segir til um á hvaða skjáblaðsíðu skal finna staðsetningu depilsins.

Eftir: lína og dálkur innihalda fjöldatölur sem segja til um staðsetningu depilsins á skjáblaðsíðunni blaðsíða.

depilstaðsetja

Notkun: depilstaðsetja(; lína, dálkur, blaðsíða)

Fyrir: lína, dálkur og blaðsíða eru fjöldatölur.

Eftir: Depillinn er staðsettur á skjáblaðsíðunni blaðsíða eftir staðsetningunni sem lína og dálkur gefa.

fillameðstaf

Notkun: fillameðstaf(; stafur, fjöldi, blaðsíða)

Fyrir: stafur er stafur. fjöldi og blaðsíða eru fjöldatölur.

Eftir: Stefið skrifar út fjöldi stafi af stafnum stafur út í skjáblaðsíðuna blaðsíða frá þeim stað sem depillinn er staðsettur í þeirri skjáblaðsíðu.

hreinsaskjá**Notkun:** hreinsaskjá(;)**Fyrir:** Skjáblaðsíða sú sem hreinsa á, er sú sem er virk við kallið.**Eftir:** Stefið hreinsar skjáblaðsíðuna sem er virk við kallið á stefið. Depillinn er staðsettur í lokin í efra vinstra horn skjásins (hnit 1,1).**ritastaf****Notkun:** ritastaf(;stafur,blaðsíða)**Fyrir:** stafur er stafur. blaðsíða er fjöldatala.**Eftir:** Stefið skrifar stafinn stafur út á skjáblaðsíðuna blaðsíða með eiginleika þeim sem settur var með stefinu setjaeiginleik sem er lýst nánar hér að neðan. Ef ekki hefur verið settur eiginleiki með því stefi, þá er stafurinn skrifaður með eiginleikanum \$07 (þ.e. hvítum stöfum á svörtum fleti). Athuga ber, að sé stafur skrifaður með þessu stefi þá er depillinn ekki færður áfram.**ritastafi****Notkun:** ritastafi(;c,n,blaðsíða)**Fyrir:** c er stafur. n og blaðsíða eru fjöldatölur.**Eftir:** Stefið skrifar n stafa af stafnum c út á skjáblaðsíðuna blaðsíða með eiginleika þeim sem settur var með stefinu setjaeiginleik sem er lýst nánar hér að neðan. Ef ekki hefur verið settur eiginleiki með því stefi, þá eru stafirnir skrifaðir með eiginleikanum \$07 (þ.e. hvítum stöfum á svörtum fleti). Athuga ber að, séu stafir skrifaðir með þessu stefi þá er depillinn ekki færður áfram.**setjablaðsíðu****Notkun:** setjablaðsíðu(;blaðsíða)**Fyrir:** blaðsíða er fjöldatala.**Eftir:** Eftir að stefið er framkvæmt er skjáblaðsíðan blaðsíða virk.**setjadepil****Notkun:** setjadepil(;afeðaá,byrjalínu,endalínu)**Fyrir:** afeðaá, byrjalínu og endalínu eru fjöldatölur.**Eftir:** Stefið stjórnar gerð depilsins (e. *cursor*). Ef afeðaá hefur gildið 0 við kallið þá kviknar á deplinum, ef afeðaá er eitthvað annað gildi þá slökknar á honum. Venjulega sést depillinn sem ein eða tvær skannlínur (e. *scan lines*) á skjánum. Hægt er að breyta stærð depilsins með því að setja önnur gildi í byrjalínu og endalínu (byrjalínu og endalínu segja til um skannlínur fyrir depilinn). Lita/grafík tengildi (e. *color graphics adapter, CGA*) getur sýnt átta skannlínur, númeraðar efst frá 0 niður í 7. Einlita tengildi og EGA geta sýnt depil með 14 skannlínur, númeraðar efst frá 0 niður í 13. Ef byrjalínu er minni en endalínu, kemur venjulegur heill depill upp. Ef byrjalínu er stærri en endalínu þá sést depillinn í tveimur hlutum. Ef byrjalínu er gefið gildið 32 þá hverfur depillinn.

setjæiginleika

Notkun: setjæiginleika(;x)

Fyrir: x er fjöldatala.

Eftir: Stefið sér um að geyma eiginleikann x, sem gefin var upp í kallinu. Næst þegar stefin ritastaf eða ritastafi eru keyrð þá er gildið notað sem eiginleiki stafsins/stafanna sem skrifaðir eru. Til þess að finna æskilegan eiginleika er eftirfarandi aðferðir notaðar:

Einlitur skjár (e. *monochrome*):

1. Veljið eitt af eftirfarandi:

Svartir stafir á svörtum fleti (þ.e. autt): x := 0

Hvítir stafir á svörtum fleti (þ.e. venjulegt einkenni): x := 7

Svartir stafir á hvítum fleti: x := 112

2. Leggja 1 við x ef stafirnir eiga vera undirstrikaðir.

3. Leggja 8 við x ef stafirnir eiga að vera upplýstir (e. *high intensity*)

4. Leggja 128 við x ef stafirnir eiga að blikka.

Litaskjár:

1. Veljið einn lit á stafi og einn lit á flöt úr eftirfarandi töflu og leggið saman. Gefið x það gildi sem þá kemur út.

stafir flötur

svart 0 svart 0

grátt 8 blátt 16

blátt 1 grænt 32

ljósblátt 9 blágrænt 48

grænt 2 rautt 64

ljósgrænt 10 blárautt 80

blágrænt 3 brúnt 96

ljóscyan 11 hvítt 112

rautt 4

ljósrautt 12

blárautt 5

ljósblárautt 13

brúnt 6

gult 14

hvítt 7

hvítt (upplýst) 15

2. Leggið 128 við x, ef stafirnir eiga að blikka.

skjástaða

Notkun: `skjástaða(skjástaða,línulengd,skjáblaðsíða;)`

Fyrir: Ekkert.

Eftir: Stefið skilar eftirfarandi upplýsingum um stöðu skjásins, þegar kallað er: skjástaða (e. *current display mode*), línulengd (fjöldi stafa í hverri línu á skjánum) og virk skjáblaðsíða (e. *active display page*).

skrunaniður

Notkun: `skrunaniður(;n,y1,x1,y2,x2)`

Fyrir: `n`, `y1`, `x1`, `y2` og `x2` eru fjöldatölur.

Eftir: Stefið veldur því að rétthyrningur sem afmarkast af efra vinstra horni af (`x1,y1`) og neðra hægra horni af (`x2,y2`) er skrunaður (e. *scroll*) niður um `n` línur. Settar eru auðar línur í stað þeirra sem skrunaðar eru. Ef fjölina hefur gildið 0 við kall þá eru allar línur skrunaðar niður í rétthyrningnum, þ.e. allt er þurrkað út í honum.

skrúnaupp

Notkun: `skrúnaupp(;n,y1,x1,y2,x2)`

Fyrir: `n`, `y1`, `x1`, `y2` og `x2` eru fjöldatölur.

Eftir: Stefið veldur því að rétthyrningur sem afmarkast af efra vinstra horni af (`x1,y1`) og neðra hægra horni af (`x2,y2`) er skrúnaður (e. *scroll*) upp um `n` línur. Settar eru auðar línur í stað þeirra sem skrúnaðar eru. Ef `n` hefur gildið 0 við kall þá eru allar línur skrúnaðar upp í rétthyrningnum, þ.e. allt er þurrkað út í honum.

þurrkaendalínu

Notkun: `þurrkaendalínu(;y,x)`

Fyrir: `y` og `x` eru fjöldatölur.

Eftir: Stefið veldur því að þurrkað er út úr línunni `x` frá og með dálknum `y` allt til enda línunnar.

þurrkaendaskjá

Notkun: `þurrkaendaskjá(;y,x)`

Fyrir: `y` og `x` eru fjöldatölur.

Eftir: Stefið veldur því að þurrkað er út af skjánum frá og með dálknum `x` í línunni `y` allt til neðra hægra horns skjásins.

Kaflí 37

Einingin SLEMBI

Einingin SLEMBI inniheldur tvö stef sem smíða slembitölur. Einungis er vísað í falin kjarnastef úr þessari einingu.

fleytslembi

Notkun: `s := fleytslembi(;)`

Fyrir: Ekkert.

Eftir: `s` inniheldur slembitölu sem er fleytitala milli 0.0 og 1.0 að þeirri fyrri meðtalinni, en ekki þeirri seinni.

slembi

Notkun: `s := slembi(;)`

Fyrir: Ekkert.

Eftir: `s` inniheldur slembitölu sem er fjöldatala milli 0 og 65535 að báðum meðtöldum.

Kafli 38

Einingin STORFJOL

Einingin STORFJOL inniheldur stef sem gera notanda kleift að reikna með stórar fjöldatölur. Venjulegar fjöldatölur milli 0 og 65535 eru undirmengi í mengi stórra fjöldatalna, geymdar á venjulegan hátt. Stærri tölur eru geymdar í strengjum. Engin fjöldatala er neikvæð, ef nauðsynlegt er að vinna með neikvæðar tölur ætti að nota venjulegar heiltölur sem eru milli -32768 og 32767 eða nota mætti stórar heiltölur (sjá eininguna STORHEIL). Úr þessari einingu er einungis vísað í falin kjarnastef.

%

Notkun: $s := x \% y$

Fyrir: x og y eru stórar fjöldatölur.

Eftir: s inniheldur stóru fjöldatöluna sem er afgangurinn þegar y er deilt í x . Fyrir allar stórar fjöldatölur x og y er tryggt að $0 \leq x \% y < y$ og $y * (x / y) + (x \% y) = x$.

*

Notkun: $s := x * y$

Fyrir: x og y eru stórar fjöldatölur.

Eftir: s inniheldur stóru fjöldatöluna sem er margfeldi x og y .

-

Notkun: $s := x - y$

Fyrir: x og y eru stórar fjöldatölur og $x \geq y$.

Eftir: s inniheldur stóru fjöldatöluna sem er mismunur x og y .

/

Notkun: $s := x / y$

Fyrir: x og y eru stórar fjöldatölur.

Eftir: s inniheldur stóru fjöldatöluna sem er kvótinn þegar y er deilt í x . Fyrir allar stórar fjöldatölur x og y er tryggt að $0 \leq x / y$.

<

Notkun: $s := x < y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

<=

Notkun: $s := x <= y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

<>

Notkun: $s := x <> y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

=

Notkun: $s := x = y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

>

Notkun: $s := x > y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

>=

Notkun: $s := x >= y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

^

Notkun: $s := x \wedge y$ **Fyrir:** x og y eru stórar fjöldataölur.**Eftir:** s inniheldur stóru fjöldataöluna sem er x í veldinu y .

fdeiling

Notkun: `fdeiling(q,r;x,y)`

Fyrir: x og y eru stórar fjöldataölur.

Eftir: q inniheldur stóru fjöldataöluna x/y og r inniheldur stóru fjöldataöluna $x\hat{E}$

fgcd

Notkun: `s := x \fgcd y`

Fyrir: x og y eru stórar fjöldataölur.

Eftir: s inniheldur stóru fjöldataöluna sem er stærsti sameiginlegi deilir x og y .

skrifafjöl

Notkun: `skrifafjöl(;x)`

Fyrir: x er stór fjöldata.

Eftir: x hefur verið skrifað á aðalúttak sem tugatala.

Kafli 39

Einingin STORHEIL

Einingin STORHEIL inniheldur stef sem gera notanda kleift að reikna með stórar heiltölur. Venjulegar heiltölur milli -32768 og 32767 eru undirmengi í mengi stórra heiltalna, geymdar á venjulegan hátt. Stærri tölur eru geymdar í strengjum.

Þessi eining vísar í stef úr einingunni STORFJOL og muna þarf að tengja þá einingu einnig ef þessi eining er notuð.

Útflutningar

%%

Notkun: $s := x \% y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur stóru heiltöluna sem er afgangurinn þegar y er deilt í x .

**

Notkun: $s := x ** y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur stóru heiltöluna sem er margfeldi x og y .

–

Notkun: $s := x -- y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur stóru heiltöluna sem er mismunur x og y .

//

Notkun: $s := x // y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur stóru heiltöluna sem er kvótinn þegar y er deilt í x .

«

Notkun: $s := x \ll y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

<=<=

Notkun: $s := x \leq y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

<><>

Notkun: $s := x \neq y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

==

Notkun: $s := x = y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

>=>=

Notkun: $s := x \geq y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

»

Notkun: $s := x \gg y$

Fyrir: x og y eru stórar heiltölur.

Eftir: s inniheldur niðurstöðuna úr samanburðinum, satt eða ósatt.

hdeiling

Notkun: $fdeiling(q,r;x,y)$

Fyrir: x og y eru stórar heiltölur.

Eftir: q inniheldur stóru heiltöluna x / y og r inniheldur stóru heiltöluna x

hgcd

Notkun: `s := x \hgcd y`

Fyrir: `x` og `y` eru stórar heiltölur.

Eftir: `s` inniheldur stóru heiltöluna sem er stærsti sameiginlegi deilir `x` og `y`.

skrifaheil

Notkun: `skrifaheil(;x)`

Fyrir: `x` er stór heiltala.

Eftir: `x` hefur verið skrifað á aðalúttak sem tugatala.

Innflutningar

Einingin `STORHEIL` notar eftirtalin stef úr einingunni `STORFJOL`:

Kafli 40

Einingin TAFLA

Einingin TAFLA inniheldur stef til að vinna með töflur. Töflurnar eru söfn færsla þar sem hver færsla hefur lykil og gildi. Aðeins ein færsla í hverri töflu getur innihaldið tiltekinn lykil. Einingin er fjölnota þar eð notandinn getur ákveðið hvers konar gildi hann notar sem lykla með því að tengja viðeigandi samanburðaraðgerð á lykla.

Útflutningar

eyða

Notkun: `eyða(;tafla,lykill)`

Fyrir: tafla er tafla. lykill er lykilgildi.

Eftir: Ef færsla með lyklinum lykill finnst í töflunni tafla er henni eytt úr töflunni.

fletja

Notkun: `s := fletja(;tafla)`

Fyrir: tafla er tafla.

Eftir: s inniheldur lista allra færsla í töflunni tafla. Hvert stak í s inniheldur tvö stök, lykil og meðfylgjandi gildi. T.d. ef taflan t inniheldur tvo lykla, a og b, sem hafa meðfylgjandi gildi x og y þá skilar fletja(;t) listanum `[[a,x],[b,y]]`.

innsetja

Notkun: `innsetja(;tafla,lykill,gildi)`

Fyrir: tafla er tafla. lykill er eitthvert lykilgildi. gildi er einhvert gildi.

Eftir: Stefið veldur því að bætt er í töfluna tafla færslu með lyklinum lykill og meðfylgjandi gildi gildi. Ef lykillinn lykill er til fyrir í töflunni er gildi sett í stað gamla gildisins í viðkomandi færslu.

leita

Notkun: `s := leita(gildi;tafla,lykill)`

Fyrir: tafla er tafla. lykill er lykill.

Eftir: Ef lykillinn finnst í töflunni fær s sanngildið satt og gildinu sem fylgir lyklinum er skilað í gildi. Ef lykillinn finnst ekki þá fær s sanngildið ósatt.

lyklamengi

Notkun: $s := \text{lyklamengi}(\text{;tafla})$

Fyrir: tafla er tafla.

Eftir: s er listi allra lykla í töflunni tafla.

lyklasniðmengi

Notkun: $s := \text{lyklasniðmengi}(\text{;tafla1,tafla2})$

Fyrir: tafla1 og tafla2 eru töflur.

Eftir: s er listi allra lykla sem sameiginlegar eru með töflunum tafla1 og tafla2.

samsetja

Notkun: $s := \text{samsetja}(\text{;tafla1,tafla2})$

Fyrir: tafla1 og tafla2 eru töflur.

Eftir: s inniheldur nýja töflu sem inniheldur færslur beggja taflnanna. Ef einhverjir lykjar eru sameiginlegir báðum töflunum þá er gildið úr færslunni í tafla1 ráðandi í nýju töflunni.

tóm

Notkun: $s := \text{tóm}(\text{;})$

Fyrir: Ekkert.

Eftir: s inniheldur nýja tóma töflu. Þessi nýja tafla inniheldur engar færslur.

Innflutningar

samanburður

Notkun: $s := \text{samanburður}(\text{;lykill1,lykill2})$

Fyrir: lykill1 og lykill2 eru lykjar

Eftir: s inniheldur stafinn '<', '=' eða '>' eftir því vernig röð lyklnanna er. Samanburður þessi verður ávallt að skila einu þessara þriggja stafa, og gilda verða venjulegar reglur um samanburðaraðgerðir, t.d. ef $x < y$ og $y < z$ þá verður að gilda $x < z$.

Kafli 41

Einingin TAYLOR

Einingin TAYLOR inniheldur stef til að reikna með Taylor raðir. Eða réttara sagt með MacLaurin raðir þar eð reiknað er með að grunnpunkturinn sé núll. Einingin er fjölnota þar eð notandinn getur sjálfur ákveðið hvers konar stuðlar eru í röðunum með því að tengja viðeigandi aðgerðir á stuðla.

Útflutningar

*

Notkun: $s := x * y$

Fyrir: x og y eru Taylor raðir.

Eftir: s inniheldur Taylor röðina sem er margfeldi x og y . Lengd s er sú styttri af lengdum x og y .

+

Notkun: $s := x + y$

Fyrir: x og y eru Taylor raðir.

Eftir: s inniheldur Taylor röðina sem er summa x og y . Lengd s er sú styttri af lengdum x og y .

-

Notkun: $s := x - y$

Fyrir: x og y eru Taylor raðir.

Eftir: s inniheldur Taylor röðina sem er mismunur x og y . Lengd s er sú styttri af lengdum x og y .

/

Notkun: $s := x / y$

Fyrir: x og y eru Taylor raðir.

Eftir: s inniheldur Taylor röðina sem er kvóti x og y . Lengd s er sú styttri af lengdum x og y .

cos**Notkun:** $s := \backslash\cos n$ **Fyrir:** n er fjöldatala.**Eftir:** s inniheldur Taylor röðina fyrir kósínus af lengd n .**diffra****Notkun:** $s := \backslash\text{diffra } f$ **Fyrir:** f er Taylor röð.**Eftir:** s inniheldur Taylor röðina sem er diffurkvótinn af f .**exp****Notkun:** $s := \backslash\exp n$ **Fyrir:** n er fjöldatala.**Eftir:** s inniheldur Taylor röðina fyrir e^x af lengd n .**fasti****Notkun:** $s := k \backslash\text{fasti } n$ **Fyrir:** k er stuðull, n er fjöldatala.**Eftir:** s inniheldur Taylor röðina fyrir fastann k af lengd n . M.ö.o.: $s = kz^0 + 0z^1 + \dots + 0z^{n-1} + O(z^n)$.**fyrsti****Notkun:** $s := \backslash\text{fyrsti } x$ **Fyrir:** x er Taylor röð, ekki tóm, þ.e.a.s. ekki $O(1)$.**Eftir:** s inniheldur fyrsta stuðul í Taylor röðinni. Ef $x = a_0z^0 + \dots + a_nz^n + O(z^{n+1})$ þá inniheldur s stuðulinn a_0 .**gildi****Notkun:** $s := x \backslash\text{gildi } t$ **Fyrir:** x er Taylor röð, t er stuðull.**Eftir:** s inniheldur gildi Taylor raðarinnar í punktinum t . Ef $x = a_0z^0 + \dots + a_nz^n + O(z^{n+1})$ þá inniheldur s stuðulgildið $a_0t^0 + \dots + a_nt^n$.**rest****Notkun:** $s := \backslash\text{rest } x$ **Fyrir:** x er Taylor röð, ekki tóm, þ.e.a.s. ekki $O(1)$.**Eftir:** s inniheldur halann af Taylor röðinni. Ef $x = a_0z^0 + \dots + a_nz^n + O(z^{n+1})$ þá inniheldur s Taylor röðina $a_1z^0 + \dots + a_nz^{n-1} + O(z^n)$.

samskeyta**Notkun:** `s := x \samskeyta y`**Fyrir:** x og y eru Taylor raðir.**Eftir:** s inniheldur fallasamskeytingu Taylor raðanna. Ef $x = a_0z^0 + \dots + a_nz^n + O(z^{n+1})$ og $y = b_0z^0 + \dots + b_mz^m + O(z^{m+1})$ þá inniheldur s Taylor röðina sem reiknuð er út frá formúlunni $a_1y^0 + \dots + a_ny^{n-1} + O(z^n)$.**sin****Notkun:** `s := \sin n`**Fyrir:** n er fjöldatala.**Eftir:** s inniheldur Taylor röðina fyrir sínus af lengd n.**skrifataylor****Notkun:** `skrifataylor(;x)`**Fyrir:** x er Taylor röð.**Eftir:** Taylor röðin x hefur verið skrifuð á læsilegan hátt á aðalúttak.**tegra****Notkun:** `s := \tegra x`**Fyrir:** x er Taylor röð.**Eftir:** s inniheldur Taylor röðina sem er tegrið (heildið) af x.**villuveldi****Notkun:** `s := \villuveldi x`**Fyrir:** x er Taylor röð.**Eftir:** s inniheldur veldisvísinn úr leifarlið Taylor raðarinnar. M.ö.o. ef $x = a_0z^0 + \dots + a_nz^n + O(z^{n+1})$ þá inniheldur s fjöldatöluna n+1.**Innflutningar**

Úr einingunni TAYLOR er vísað í eftirfarandi stef sem vinna með stuðla Taylor raðanna.

*

Notkun: `s := x * y`**Fyrir:** x og y eru stuðlar.**Eftir:** s inniheldur stuðulinn sem er margfeldi x og y.

+

Notkun: $s := x + y$

Fyrir: x og y eru stuðlar.

Eftir: s inniheldur stuðulinn sem er summa x og y .

-

Notkun: $s := x - y$

Fyrir: x og y eru stuðlar.

Eftir: s inniheldur stuðulinn sem er mismunur x og y .

/

Notkun: $s := x / y$

Fyrir: x og y eru stuðlar.

Eftir: s inniheldur stuðulinn sem er kvóti x og y .

=

Notkun: $s := x = y$

Fyrir: x og y eru stuðlar.

Eftir: s inniheldur sanngildi samanburðarins, satt eða ósatt.

fjöltilstuðull

Notkun: $s := \backslash\text{fjöltilstuðull } n$

Fyrir: n er fjöldatala.

Eftir: s inniheldur stuðulinn sem samsvarar n .

skrifastuðul

Notkun: $\text{skrifastuðul}(;x)$

Fyrir: x er stuðull.

Eftir: stuðullinn hefur verið skrifaður á læsilegan hátt á aðalúttak.

Kafli 42

Einingin VIDBSKJA

Í þessari einingu eru ýmis stef sem tengja má einingunni SKJAR til að fá fleiri kosti en í henni eru.

Útflutningar

depiláfram

Notkun: depiláfram(;)

Fyrir: Skjáblaðsíða sú sem færa á depilinn á, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið veldur því að depillinn (e. *cursor*) færist einn dálk áfram en helst samt í sömu línu. Þegar enda línunnar er náð færist depillinn ekki lengra.

depilbakka

Notkun: depilbakka(;)

Fyrir: Skjáblaðsíða sú sem færa á depilinn á, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið veldur því að depillinn færist einn dálk aftur en helst samt í sömu línu. Þegar fremst í línuna er komið færist depillinn ekki lengra.

depilniður

Notkun: depilniður(;)

Fyrir: Skjáblaðsíða sú sem færa á depilinn á, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið veldur því að depillinn færist eina línu niður en helst samt í sama dálk. Þegar neðst á skjáinn er komið færist depillinn ekki neðar.

depilnýjalínu

Notkun: depilnýjalínu(;)

Fyrir: Skjáblaðsíða sú sem færa á depilinn á, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið veldur því að depillinn færist eina línu niður og í fremsta dálk þeirrar línu. Þegar neðst á skjáinn er komið færist depillinn ekki neðar.

depilupp

Notkun: depilupp(;)

Fyrir: Skjáblaðsíða sú sem færa á depilinn á, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið veldur því að depillinn færast eina línu upp en helst samt í sama dálk. Þegar efst á skjáinn er komið færast depillinn ekki ofar.

endursetjadepil

Notkun: endursetjadepil(;)

Fyrir: Skjáblaðsíða sú sem færa á depilinn á, verður að vera virk þegar kallað er á stefið.

Eftir: Depillinn er settur á þann stað sem hann var á þegar síðast var kallað á stefið geymadepilstöðu. Eftir að kallað hefur verið á þetta stef þá "gleymir"stefið í hvaða stöðu depillinn var síðast. Forritið geymir stöðurnar á stafla (fyrstur inn, síðast út), þannig að þegar kallað er aftur á stefið, án þess að kalla á stefið geymadepilstöðu, þá "man"stefið stöðu depilsins sem var geymd fyrir síðustu stöðu. Þegar engar stöður eru eftir þá færir stefið depilinn ekki til.

eyðalínu

Notkun: eyðalínu(;lína)

Fyrir: lína er fjöldatala. Skjáblaðsíða sú sem eyða á línunni á, verður að vera virk þegar kallað er á stefið.

Eftir: Sér um að eyða línunni lína á skjá. Línurnar fyrir neðan hana færast síðan upp um eina línu (neðsta línan á skjánum verður auð).

geymadepilstöðu

Notkun: geymadepilstöðu(;)

Fyrir: Skjáblaðsíða sú sem geyma á stöðu depilsins fyrir, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið geymir stöðu depilsins á skjánum, á stafla (fyrstur inn, fyrstur út), hægt er að geyma fleiri en eina stöðu í einu. Hægt er að staðsetja depilinn síðan aftur á sama stað með því að kalla á stefið endursetjadepil. Þegar ekki hefur verið kallað á geymadepilstöðu eða þegar staffinn er tómur þá hefur stefið endursetjadepil engin áhrif á stöðu depilsins.

innsetjalínu

Notkun: innsetjalínu(;lína)

Fyrir: lína er fjöldatala. Skjáblaðsíða sú sem setja á línuna á, verður að vera virk þegar kallað er á stefið.

Eftir: Stefið setur inn eina auða línu í línunni lína á skjánum, allar línur þar fyrir neðan færast niður um eina línu, þ.a. neðsta línan á skjánum hverfur.

Innflutningar

Auk falinna kjarnastefja er vísað í eftirfarandi stef sem þurfa að vinna eins og þeim er lýst í lýsingunni fyrir eininguna skjar.

depilstaða
depilstaðsetja
skjástaða
skrunaniður
skrúnaupp

Kaflí 43

Einingin VIDFÖNG

Einingin vidföng inniheldur tvær aðgerðir til að fá upplýsingar um þá skipanalínu sem gefin er þegar viðkomandi forrit er keyrt.

Þegar forrit er sett af stað í DOS, er það gert með skipanalínu þar sem fram kemur nafn forritsins, ásamt þeim vidföngum sem forritið þarfnast. DOS sér um að koma vidföngunum fyrir á ákveðnum stað í minni tölvunnar, þar sem forritið getur nálgast þau. Það svæði sem DOS velur til geymslu þeirra, er samt sem áður alls ekki óhult, þar sem DOS notar það undir margt annað.

Til þess að notandi sé öruggur um að ekki sé skrifað yfir vidföngin, er best að láta kall á aðra þá aðgerð sem útluttar eru úr þessari einingu, vera eitt af því fyrsta sem forrit gerir. Báðar aðgerðirnar taka afrit af vidföngunum yfir á öruggan stað og geyma þau þar, þar til keyrslu forritsins lýkur.

Úr einingunni er aðeins vísað í falin kjarnastef.

viðfang

Notkun: strengur := viðfang(;i)

Fyrir: i er fjöldatala.

Eftir: Ef i er minni en eða jafnt viðfangafjöldi(;) (sjá hér að ofan) og stærri en 0, þá fær strengur gildi strengs sem inniheldur viðfang númer i í skipanalínunni sem gefin var þegar viðkomandi forrit var sett af stað. Ef i er stærri en viðfangafjöldi(;) eða i er 0, fær strengur gildið ósatt.

Dæmi:

```
C:\>prufa vidföng
```

Hér gæfi vidföng(;1) strenginn "vidföng", vidföng(;2) gildið ósatt.

```
C\>prufa vidföng1 vidföng2
```

Hér gæfi vidföng(;1) strenginn "vidföng1", vidföng(;2) strenginn "vidföng2"en vidföng(;3) gildið ósatt.

viðfangafjöldi

Notkun: fjöldi := viðfangafjöldi(;)

Fyrir: Ekkert.

Eftir: fjöldi fær gildi fjöldatölu sem segir til um fjölda viðfanga í skipanalínunni sem gefin var þegar forritið var sett af stað. Ef engin viðföng voru gefin, fær fjöldi gildið 0.

Sem dæmi um skipanalínu væri hægt að keyra forritið prufa með skipuninni:

```
C:\>prufa vidföng
```

Ef forritið prufa framkvæmdi skipunina fjöldi := viðfangafjöldi(;), fengi fjöldi gildið 1.

```
C:\>prufa vidföng1 vidföng2
```

Hér fengi fjöldi gildið 2.

```
C:\>prufa
```

Hér fengi fjöldi gildið 0.

Kafli 44

Einingar fyrir teiknun

Einingarnar TATAHGA, TATACGA, TATACGA3, TATAEGA, TATAMCGA og TATAVGA eru allar svipaðar í notkun. Þær innihalda stef til að teikna á skjá. Teikna má á þrennan hátt. Nota má tátuteiknun (e. *turtle graphics*) svipað og í LOGO forritunarmálinu, teikna má samkvæmt fleytitöluhnitum sem eru tiltölulega óháð gerð skjásins sem notaður er, eða teikna má samkvæmt fjöldatöluhnitum sem eru háð gerð skjásins. Allar þessar gerðir teiknunar má nota samtímis ef óskað er.

Þegar notuð eru fleytitöluhnit eða tátuteiknun vísa hnitin (0.0,0.0) í punktinn neðst til vinstri á skjánum. Punkturinn (1.0,1.0) er efst á skjánum þannig að hann er andstæður (0.0,0.0) á ferningi.

Þegar notuð eru fjöldatöluhnit er vísað beint í þá punkta á skjánum sem teikna má í. Hnitin eru mismunandi eftir gerð skjásins, en punkturinn neðst til vinstri er ávallt (0,0).

Unnt er að teikna stafi og strengi á skjáinn, og eru stafir þá teiknaðir í 8x8 punkta kassa. Ef notaðir eru íslenskir stafir í einhverju forriti er nauðsynlegt að keyra forritið GRAFTBIC sem fylgir stýrikerfinu áður en fyrirnefnda forritið er keyrt. Annars verða íslensku stafirnir ekki teiknaðir rétt.

Þegar teiknieiningarnar eru notaðar er nauðsynlegt að setja viðkomandi skjá í teikniham. Í hverri teiknieiningu er stef sem heitir hefja. Ávallt skal kalla á þetta stef áður en byrjað er að teikna á skjáinn. Síðan má teikna að vild, og að lokum kalla á stefið ljúka til að setja skjáinn aftur í ritham.

Úr þessum einingum er aðeins vísað í falin kjarnastef.

Útflutningar

áfram

Notkun: áfram(;1)

Fyrir: Skjárinn er í teikniham, tátan er stödd á einhverjum stað (x,y) með stefnu (dx,dy).

Eftir: Tátan er stödd á (x+dx,y+dy). Ef tátan var niðri þá hefur verið dregin lína frá (x,y) til (x+dx,y+dy).

baugur

Notkun: baugur(;r)

Fyrir: r er fleytitala stærri en núll.

Skjárinn er í teikniham, tátan er stödd á einhverjum stað (x,y), sem er a.m.k. í fjarlægð r frá jaðri skjásins.

Eftir: Teiknaður hefur verið baugur um (x,y) með geisla r.

fpunktur

Notkun: fpunktur(x,y)

Fyrir: x og y eru fjöldatölur þ.a. punkturinn (x,y) er á skjánum.

Eftir: Teiknaður hefur verið punktur á (x,y).

flína

Notkun: flína(x1,y1,x2,y2)

Fyrir: x1,y1, x2 og y2 eru fjöldatölur þ.a. punktarnir (x1,y1) og (x2,y2) eru á skjánum.

Eftir: Teiknuð hefur verið lína frá (x1,y1) til (x2,y2).

hefja

Notkun: hefja(x)

Fyrir: Ekkert.

Eftir: Skjárinn er auður í teikniham, tátan er staðsett á (0.5,0.5) með stefnu (1.0,0.0), númer litarins er 1 og tátan er niðri.

hægri

Notkun: hægri(x)

Fyrir: Ekkert.

Eftir: Tátunni hefur verið snúið um 90 gráður til hægri.

litur

Notkun: s := litur(x)

Fyrir: x er fjöldatala sem er litarnúmer.

Eftir: Liturinn er nú x og s inniheldur gamla litarnúmerið.

lína

Notkun: s := lína(x1,y1,x2,y2)

Fyrir: x1, y1, x2 og y2 eru fleytitölur.

Eftir: Dregin hefur verið lína frá (x1,y1) til (x2,y2).

ljúka

Notkun: s := ljúka(x)

Fyrir: Skjárinn er í teikniham.

Eftir: Skjárinn er auður í ritham.

niðri

Notkun: `s := niðri(;x)`

Fyrir: `x` er eitthvert gildi.

Eftir: Ef `x` er satt þá er tátan niðri, annars uppi. `s` inniheldur satt ef tátan var áður niðri, annars ósatt.

punktur

Notkun: `s := punktur(;x,y)`

Fyrir: `x` og `y` eru fleytitölur.

Eftir: Teiknaður hefur verið punktur í (x,y) .

snúa

Notkun: `snúa(;h)`

Fyrir: `h` er fleytitala.

Eftir: Tátunni hefur verið snúið um `h` gráður til vinstri.

staður

Notkun: `staður(x,y;)`

Fyrir: Ekkert.

Eftir: `x` og `y` innihalda fleytitöluhnit tátunnar.

stefna

Notkun: `stefna(dx,dy;)`

Fyrir: Ekkert.

Eftir: `dx` og `dy` innihalda fleytitölur þ.a. (dx,dy) er stefna tátunnar.

strengur

Notkun: `strengur(;s)`

Fyrir: `s` er textastrengur.

Eftir: Textinn í `s` hefur verið teiknaður frá (x,y) til hægri.

vinstri

Notkun: `vinstri(;)`

Fyrir: Ekkert.

Eftir: Tátunni hefur verið snúið um 90 gráður til vinstri.

44.1 Einingin TATAHGA

Einingin TATAHGA inniheldur stef þau sem lýst er að ofan. Skjárinn er einlitur þ.a. einu löglegu litirnir eru 0 og 1. Skjárinn hefur 720 punkta í hverri láréttri línu og hefur 348 láréttar línur. Um lögleg fjöldatöluhnit (x,y) gildir því $0 \leq x < 720$ og $0 \leq y < 348$.

Til að keyra forrit sem nota þessa einingu þarf Hercules teiknisþjald í tölvuna.

44.2 Einingin TATACGA

Einingin TATACGA inniheldur stef þau sem lýst er að ofan. Skjárinn er einlitur þ.a. einu löglegu litirnir eru 0 og 1. Skjárinn hefur 640 punkta í hverri láréttri línu og hefur 200 láréttar línur. Um lögleg fjöldatöluhnit (x,y) gildir því $0 \leq x < 640$ og $0 \leq y < 200$.

Til að keyra forrit sem nota þessa einingu þarf "Color Graphics"spjald í tölvuna.

44.3 Einingin TATACGA3

Einingin TATACGA3 inniheldur stef þau sem lýst er að ofan. Skjárinn er fjórliturlit þ.a. einu löglegu litirnir eru 0, 1, 2 og 3. Skjárinn hefur 320 punkta í hverri láréttri línu og hefur 200 láréttar línur. Um lögleg fjöldatöluhnit (x,y) gildir því $0 \leq x < 320$ og $0 \leq y < 200$.

Til að keyra forrit sem nota þessa einingu þarf "Color Graphics"spjald í tölvuna.

44.4 Einingin TATAEGA

Einingin TATAEGA inniheldur stef þau sem lýst er að ofan. Skjárinn hefur 16 liti þ.a. löglegir litir eru frá 0 til 15. Skjárinn hefur 640 punkta í hverri láréttri línu og hefur 400 láréttar línur. Um lögleg fjöldatöluhnit (x,y) gildir því $0 \leq x < 640$ og $0 \leq y < 400$.

Til að keyra forrit sem nota þessa einingu þarf EGA spjald í tölvuna.

44.5 Einingin TATAMCGA

Einingin TATAMCGA inniheldur stef þau sem lýst er að ofan. Skjárinn hefur 16 liti þ.a. löglegir litir eru frá 0 til 15. Skjárinn hefur 320 punkta í hverri láréttri línu og hefur 200 láréttar línur. Um lögleg fjöldatöluhnit (x,y) gildir því $0 \leq x < 320$ og $0 \leq y < 200$.

Til að keyra forrit sem nota þessa einingu þarf tölvan að gefa kost á MCGA teiknun eins og PS/2 vélarnar gera.

44.6 Einingin TATAVGA

Einingin TATAVGA inniheldur stef þau sem lýst er að ofan. Skjárinn hefur 16 liti þ.a. löglegir litir eru frá 0 til 15. Skjárinn hefur 640 punkta í hverri láréttri línu og hefur 480 láréttar línur. Um lögleg fjöldatöluhnit (x,y) gildir því $0 \leq x < 640$ og $0 \leq y < 480$.

Til að keyra forrit sem nota þessa einingu þarf tölvan að gefa kost á VGA teiknun eins og stærri afbrigði PS/2 vélanna gera.

Hluti IV

Atriðisorðaskrá

Atriðisorðaskrá

- * , 97, 129, 131, 145–147, 157, 167, 169
- ** , 97, 161
- *** , 97
- + , 97, 129, 131, 145–147, 167, 170
- ++ , 97, 147
- +++ , 97
- , 97, 129, 131, 145–147, 157, 167, 170
- , 98, 161
- , 98
- / , 98, 129, 131, 157, 167, 170
- // , 98, 161
- /// , 98
- .; , 98
- < , 99, 130, 132, 143, 148, 158
- « , 99, 162
- «< , 99
- <= , 99, 130, 148, 158
- <=<= , 99, 162
- <=<=<= , 99
- <> , 100, 130, 132, 158
- <><> , 100, 162
- <><><> , 100
- = , 100, 130, 132, 143, 145, 146, 148, 158, 170
- == , 100, 162
- === , 100
- > , 101, 130, 132, 158
- >= , 101, 130, 158
- >=>= , 101, 162
- >=>=>= , 101
- » , 101, 162
- »> , 101
- % , 96, 157
- %% , 96, 161
- & , 96
- ”E , 158
- ˘C , 102
- ˘C˘C , 102
- Ítrunarhlíðsetning , 37
- áfram , 177
- íflenskiriröð , 139
- ígrip10 , 108
- ígrípsstaður , 87
- íröð , 149
- íslsamanb , 139
- ítrun , 11, 31
- óendanleg tré , 55
- út , 22
- útbæti , 112
- útgáfunúmer , 87
- ýttástaf , 142
- þurrkaendalínu , 154
- þurrkaendaskjá , 154
- aðgerðarsegðir , 18
- aðgerðir:forangur , 19
- abs , 89
- AFRITA , 63
- afrita , 63, 138
- asin , 89
- atan2 , 89
- bætfylla , 102
- búatilsafn , 66
- búatilskrá , 68
- baugur , 177
- bráðabirgðarskrá , 71
- breytanafni , 68
- breytaskráraðgangi , 73
- breytur , 16
- brjóta , 102
- BROT , 129
- brot , 102, 130
- cos , 89, 168
- depiláfram , 171
- depilbakka , 171
- depilgerð , 151
- depilnýjalínu , 171
- depilniður , 171
- depilstaða , 151
- depilstaðsetja , 151
- depilupp , 172
- diffra , 168
- disklausK , 65

diskstærð, 65
 DOSDISK, 65
 DOSSKRIU, 75
 DOSSTIU, 81
 DOSYMISL, 85

 eða, 17
 ef, 18
 efnisskrársafnlisti, 73
 einingaraðgerðir, 31
 ekki, 17
 endurkvæmni, 11
 endursetjadepil, 172
 erfleyt, 102
 erfleytnúll, 103
 erfleytneikvæð, 102
 erhlunkur, 103
 erinntak, 83
 erpar, 103
 erskrátil, 67
 erstafuristreng, 119
 erstef, 103
 erstrengur, 103
 ertóm, 103
 ertala, 103
 exp, 90, 168
 eyða, 165
 eyðalínu, 172
 eyðasafni, 66
 eyðaskrá, 68

 færaskrárbendi, 71
 fasti, 168
 fdeiling, 104, 159
 fgcd, 159
 Fibonacci, 27
 fjöldataölur, 13
 fjöltilfleyt, 104
 fjöltilstuðull, 170
 flína, 178
 fletja, 165
 FLEYFÖLL, 89
 fleytitölur, 13
 fleytitala, 104
 fleytmínus, 104
 fleytslembi, 155
 fleyttilfjöl, 104
 flytja, 104
 fmargfeldi, 105
 fmismunur, 105
 forgangur aðgerða, 19

formerki, 105
 fpunktur, 178
 frátekin tákni, 44
 fsumma, 105
 fylki, 20, 135, 136
 fylkissækja1, 105, 135, 136
 fylkissetja1, 105, 135, 136
 fylla, 105
 fyllameðstaf, 151
 fyrsti, 168

 gcd, 132
 geymadepilstöðu, 172
 gildi, 168
 grunnstef, 61
 GRUNNUR, 61

 hábæti, 106
 hægri, 178
 hætta, 108
 halasetja, 106
 hali, 106
 haus, 106
 haussetja, 106
 hdeiling, 106, 162
 hefja, 178
 heiltölur, 13
 hexbæti, 137
 HEXEININ, 137
 hexorð, 137
 hexstafur, 137
 HFYLKI, 135
 hgcd, 163
 hhliðra, 106
 hliðsetning, 37
 hlunkar, 14
 hlunksækja, 107
 hlunksetja, 107
 hlunkstærð, 107
 hlunkur, 107
 hlutstrengur, 119
 hmargfeldi, 107
 hmismunur, 107
 hreinsaskjá, 152
 hsumma, 107

 INN, 91
 innútfjöldi, 108
 innbæti, 108
 innfjöldi, 108
 innflutningur, 32
 innsetja, 165

innsetjalínu, 172
ISLRÖÐ, 139

kíkjabæti, 108
kíkjaorð, 109
köll, 23
kaflí, 108
kjarnastef, 61
KJARNI, 61, 93

lágbæti, 109
lína, 178
laus_K, 109
leita, 165
lengd, 119
lesa, 91
lesalínu, 91, 113
LESALINU, 113
lesastaf, 92, 109
lesastafánbergmáls, 83
lesastafánstopps, 83
lesfastar, 44
listar, 23
litur, 178
ljúka, 178
ln, 90
lokaskrá, 68
lykilorð, 43
lykkjur, 22
lyklamengi, 166
lyklasniðmengi, 166

málfræði, 43
málrit, 48
MARGLID, 145
max, 90, 136
MENGI, 147
mergja, 149
min, 90, 136, 148
minnka, 109
msdos, 109

náistafánstopps, 82
nákvæmursafnlisti, 73
næstistafur, 92
núll, 132
núllstuðull, 146
nýlína, 121
nefnari, 131
niðri, 179

og, 17

opnaskrá, 67
orðfylla, 109

pör, 14
prentastaf, 81
prentastreng, 81
punktur, 179

QUICKSOR, 143

rót, 90
rökaðgerðir, 17
röksegðir, 17
raða, 143, 149
RADA, 149
radian, 90
rest, 148, 168
ritastaf, 152
ritastafi, 152
RSskilti, 110

sækjadag, 85
sækjadaghandfangs, 69
sækjaskrárdag, 69
sækjaskrártíma, 69
sækjatíma, 85
sækjatímahandfangs, 70
söfn, 74
safna, 110
safnlisti, 73
samanburður, 166
samsetja, 166
samsetning, 32
samskeyta, 169
setjaöryggiá, 86
setjablaðsíðu, 152
setjadag, 85
setjadaghandfangs, 70
setjadepil, 152
setjaeiginleika, 153
setjatíma, 86
setjatímahandfangs, 70
SFYLLKI, 136
sin, 90, 169
sjálfgafiðdrif, 74
sjálfgafiðsafn, 66
skalsafna, 110
skipta, 149
skiptaumsafn, 65
skiptistaða, 141
skjastaða, 154
SKJAR, 151

- skrálesaístreng, 77
 skrálesabæti, 75
 skrálesageymslu, 77
 skrálesaorð, 75
 skrálesastreng, 76
 skráraðgangur, 72
 skráskrifaúrstreng, 77
 skráskrifabæti, 75
 skráskrifageymslu, 78
 skráskrifaminni, 78
 skráskrifaorð, 76
 skráskrifastreng, 76
 skrifa, 121
 skrifaþöl, 121, 159
 skrifaheil, 163
 skrifaflínu, 115
 SKRIFALIN, 115
 skrifaþargliðu, 146
 skrifaþaf, 110
 skrifaþafánstopp, 83
 skrifaþastreng, 121
 skrifaþaðul, 146, 170
 skrifaþataylor, 169
 skrunaniður, 154
 skrunaupp, 154
 SLEMBI, 155
 slembi, 155
 snúa, 117, 179
 SNUA, 117
 stækka, 112
 staðastopp, 87
 staður, 179
 stafir, 13
 stef, 23
 stefna, 179
 stillastopp, 87
 STORFJOL, 157
 STORHEIL, 161
 strengflytja, 111
 strengfylla, 111
 STRENGIR, 119
 strengir, 13
 strengsækjabæti, 111
 strengsækjaorð, 111
 strengsetjabæti, 111
 strengsetjaorð, 111
 strengskeyta, 119
 strengstærð, 111
 strengur, 112, 179
 strengurístreng, 120
 tóm, 166
 tómagildið, 13
 tóm, 148
 tölur, 13
 TAFLA, 165
 takaöryggiaf, 86
 TAYLOR, 167
 tegra, 169
 teiknun, 177
 teljari, 131
 UT, 121
 val, 21
 varýttastaf, 141
 veitingar:einingaveitingar, 39
 veitingar:forritsveitingar, 39
 veldi, 112
 veljadrif, 74
 vhlíðra, 112
 viðfang, 175
 viðfangafjöldi, 175
 VIDBSKJA, 171
 VIDFÖNG, 175
 villuveldi, 169
 vinstri, 179
 vistfang, 112