

## Verkefni Alfa fyrri hluti eintak keppenda

### Alfa dæmi 1

Búið til forrit sem spyr um tölu. Forritið skrifar í bókstöfum hvaða tala þetta er. Talan sem slegin er inn getur verið á bilinu frá 0 til 299

*Dæmi um virkni:*

Hver er talan? 21  
tuttugu og einn

*Annað dæmi um virkni:*

Hver er talan? 211  
tvö hundruð og ellefu (hér má ekki koma tvö hundruð tíu og einn)

*Þriðja dæmið um virkni:*

Hver er talan? 127  
eitt hundruð tuttugu og sjö

### Alfa dæmi 2

Búið til forrit sem reiknar hvaða líkur eru á að þegar fimm teningum er kastað, þar sem hver teningur getur haft gildið 1-6, að upp komi 3 teningar með sama gildi og 2 teningar með sama gildi. (þ.e. hús).

Dæmi um hús er þá t.d.

1 4 4 1 4 og 3 3 2 2 3

Fimm eins tölur eru ekki hús.

Forritið framkvæmir þetta með því að fá fimm tilviljunarkenndar tölur á þessu bili mjög oft og telja hversu oft hús kemur upp og deila síðan með fjölda endurtekninga.

Ef til dæmis þetta er framkvæmt 10.000.000 sinnum og hús kemur upp í 121.234 tilvikanna þá eru líkurnar  $121234 / 10000000 = 0,0121234$

Líkurnar á að fá hús í einu kasti eru 0,003858 og ætti svar með þessari aðferð að nálgast mjög þá tölu.

### Alfa dæmi 3

Búið til forrit sem spyr um höfuðstól í upphafi tímabils og árlegan sparnað sem notandi sparar og leggur við upphæðina í byrjun hvers tímabils, en forritið reiknar hver heildarsparnaður er orðinn eftir ákveðinn árafjöldi (sparnaður er fyrst lagður við upphæðina eftir 1 ár). Vextir eru lagðir við höfuðstól árlega.

Dæmi: Ef höfuðstóll í upphafi er kr. 500.000 árlegur sparnaður er 50.000 og vextir eru 10% þá er:

Upphæð eftir 1 ár:  $500.000 + 10\% \text{ af } 500.000 = 550.000$

Upphæð eftir 2 ár:  $550.000 + 10\% \text{ af } 550.000 = 605.000 + 50.000 = 655.000$

Upphæð eftir 3 ár:  $655.000 + 10\% \text{ af } 655.000 = 720.500 + 50.000 = 770.500$

og svo framvegis

*Dæmi um virkni:*

Upphafsupphæð: 500000

Árleg viðbót: 50000

Vextir í prósentum: 10

Fjöldi ára: 3

Lokaupphæð er 770500

#### **Alfa dæmi 4**

Búið til forrit sem les textastreng. Forritið snyrtir textann til þannig að:

Fyrsti stafur á eftir . verður hástafur og það kemur bil á milli punktsins og hástafsins.

Aðrir bókstafir í setningunni verða lágstafir.

Alltaf aðeins eitt bil á milli orða.

*Dæmi um virkni:*

Texti: hann JÓN er SVAKaKALDUR KARL. hann er líka fallegur.

Hann jón er svakakaldur karl. Hann er líka fallegur.

#### **Alfa dæmi 5**

Búið til forrit sem les textaskrá. Forritið spyr um nafnið á skránni og ef skráin er ekki til segir forritið að skráin sé ekki til og biður notandann að slá inn annað nafn á skrá eða \$ táknið til að hætta í forritinu.

Ef skráin er til á forritið að telja hversu mörg orð eru í skránni. Tölur teljast sem orð, en en önnur tákni er bókstafir og tölur teljast ekki orð.

Til dæmis telst textinn

*jón er númer 10, ( var áður númer 11 )*

vera 8 orð þ.e. 10 og 11 eru orð en , ( og ) eru ekki sér orð.

Gerum ráð fyrir að skráin “ord.txt” innihaldi textann:

Fjöll í austri fagurblá

freista dalabarnsins.

Ungur fylgir æskuþrá

upp til jökulhjarnsins.

( þekki ekki höfund, líklega fyrir 1980 )

*Dæmi um virkni: (hér er setningin í skránni “ord.txt”)*

Skrá sem inniheldur texta: ordin.txt

Skráin ordin.txt er ekki til, sláðu inn nafn á skrá sem er til.

Skrá sem inniheldur texta: ord.txt

Skráin inniheldur 18 orð.

## Alfa dæmi 6

Búið til forrit sem spyr um fjölda klukkustunda, mínútna og sekútna. Búa á til forrit sem setur tíma á staðlað form. Tími á stöðluðu formi er t.d. 17:09:19

Það er fyrst kemur klukkustund sem er á bilinu 0 til 23, síðan :, síðan mínúta sem er á bilinu 0-59, síðan : og síðast sekúnta sem er á bilinu 0-59

Til að setja tölu á staðlað form má gera eftirfarandi:

Ef klukkustund er hærri tala en 23 þá er farið yfir á næsta dag, ef mínúta er stærri en 60 þá er mínútum fækkað um 60 en klukkustund í staðinn hækkuð um 1, þar til mínútur eru færri en 60 og síðan sambærileg leiðrétting ef sekúntur fara yfir 60.

*Dæmi um virkni*

klukkustund: 24

mínúta: 31

sekúnta: 61

Tími á stöðluðu formi: 0:32:01 (má líka vera 0:32:1)

Við getum gert ráð fyrir að ekki séu slegnar inn tölur minni en 0 fyrir klukkustund, mínútur og sekúntur.

## Alfa dæmi 7

Búið til forrit sem leitar að textastreng í öðrum textastreng. **Forritið á að nota endurkvæmt fall (recursive function) til að leita að strengnum.** Gera má ráð fyrir að eingöngu séu slegnir inn lágstafir og engin biltákn.

Dæmi um virkni:

Strengur sem leita á í: hannerhippi

Strengur sem leita á að: hip

Textinn hip er í strengnum hannerhippi

## Alfa dæmi 8

Búið til forrit sem les inn textastreng sem les inn hnit hornpunkta í marghyrningi. Hvert hnit er tvær heiltölur sem eru aðgreindar með kommu. Utan um hnitin eru svigar.

Dæmi um slíkan textastreng er:

$(22,3)(11,-4)(79,158)(-31,79)$

Dæmi um textastrengi sem ekki fylgja þessum reglum:

$(22,3)(11,-4)(79,158)(-31,79)$

$(22,3,6)(11,-4)(79,158)(-31,79)$

$(22,3)(11,-4)(79,158)(-31,79)$

$(22,3)(11,-4)(79,158)(-31,79)$

Búið til forrit sem spyr um textastreng sem á að fylgja þessum reglum en forritið svarar hvort textinn fylgir þessum reglum eða ekki.

*Dæmi um virkni:*

Sláið inn textastreng sem inniheldur hnit hornpunkta:

$(22,3)(11,-4)(79,158)(-31,79)$

þessi textastrengur fylgir reglum.

*Annað dæmi um virkni:*

Sláið inn textastreng sem inniheldur hnit hornpunkta:

$(22,3)(11,-4)(79,158)(-31,79)$

þessi textastrengur fylgir ekki reglum.

## **Alfa dæmi 9**

Póstnúmer í Bandaríkjunum er fimm stafa tala, t.d. 95014. Til að auðvelda flokkun pósts nota stærri fyrirtæki strikamerki til að tákna póstnúmerin, en það auðveldar vélræna flokkun póstsins. Til öryggis er notuð svo kölluð vartala til að auka öryggi flokkunarinnar en vartalan er byggð upp á eftirfarandi hátt:

Reiknuð er þversumma póstnúmersins, en þversumman eru allar tölur í póstnúmerinu lagðar saman. Þannig er t.d. þversumma póstnúmersins 95014 = 9 + 5 + 0 + 1 + 4 = 19.

Vartalan er síðan sú tala sem vantar á þversummuna til að hún hún verði margfeldi af 10 (þ.e. fari í næsta tug). Í okkar tilfelli verður vartalan því 1 þar sem  $19+1 = 20$  (eða það vantar 1 til að 19 verði margfeldi af 10)

Búið til forrit sem spyr um póstnúmer en forritið birtir síðan hver vartala viðkomandi póstnúmers er.

*Dæmi um virkni:*

Póstnúmer: 90101

Vartalan er 9

### Alfa dæmi 10

Búa á til pöntunarforrit fyrir flugsæti í flugvél. Forritið spyr um hversu margar sætaraðir eru í flugvélinni og hversum mörg sæti eru í hverri röð - þó mest 30 raðir og 6 sæti í hverri röð. Notandi getur pantað þau sæti sem hann vill en ef reynt er að panta sæti sem búið er að panta neitar forritið því og einnig bregst forritið við með villumeldingu ef reynt er að panta sæti sem ekki er í vélinni (t.d. sæti 110A eða 2G). Forritið sýnir með X ef búið er að panta ákveðið sæti.

Forritið birtir mynd af sætaskipulagi vélarinnar. (hér miðað við 10 raðir og 4 sæti)

Röð	Sæti
1	A B C D
2	A B C D
3	A B C D
4	A B C D
5	A B C D
6	A B C D
7	A B C D
8	A B C D
9	A B C D
10	A B C D

Forritið á að gera notanda mögulegt að panta ákveðin sæti með því að velja röð og sætisnúmer og merkja sæti sem pöntuð eru með X. Ekki er mögulegt að tvípana sama sætið. Þegar búið er að panta sæti 1A og 2B lítur myndin svona út:

Röð	Sæti
1	X B C D
2	A X C D
3	A B C D
4	A B C D
5	A B C D
6	A B C D
7	A B C D
8	A B C D
9	A B C D
10	A B C D

### Alfa dæmi 11

Þetta dæmi fjallar um rútufyrirtæki. Á álagstímum leitast fyrirtækið eftir að nýta rúturnar sem best þ.e. að hafa sem fæst auð sæti. Þannig vill fyrirtækið frekar nota 20 sæta rútu og 30 sæta rútu til að flytja 50 farþega frekar en eina 60 sæta rútu.

Rútufyrirtækið á rútur af 3 gerðum þ.e. 20 manna rútur, 30 manna rútur og 60 manna rútur. Forritið á að spyrja um fjölda rúta sem fyrirtækið á af hverri gerð og fjölda farþega

sem flytja á, en forritið birtir síðan hvernig rútur er best að nota en til að hámarka nýtingu bílanna er alltaf reynt að hafa sem fæst auð sæti.

*Dæmi um keyrlsu forrits:*

Farþegafjöldi: 121

Fjöldi 60 manna rúta: 3

Fjöldi 30 manna rúta: 2

Fjöldi 20 manna rúta: 3

1 stk 60 manna rúta

1 stk 30 manna rúta

2 stk 20 manna rúta

Farþegafjöldi: 89

Fjöldi 60 manna rúta: 3

Fjöldi 30 manna rúta: 2

Fjöldi 20 manna rúta: 3

1 stk 60 manna rúta

1 stk 30 manna rúta

Farþegafjöldi: 49

Fjöldi 60 manna rúta: 3

Fjöldi 30 manna rúta: 2

Fjöldi 20 manna rúta: 3

1 stk 30 manna rúta

1 stk 20 manna rúta

Farþegafjöldi: 229

Fjöldi 60 manna rúta: 3

Fjöldi 30 manna rúta: 2

Fjöldi 20 manna rúta: 3

3 stk 60 manna rúta

1 stk 30 manna rúta

1 stk 20 manna rúta

Farþegafjöldi: 249

Fjöldi 60 manna rúta: 3

Fjöldi 30 manna rúta: 2

Fjöldi 20 manna rúta: 3

3 stk 60 manna rúta

1 stk 30 manna rúta

2 stk 20 manna rúta

Farþegafjöldi: 189

Fjöldi 60 manna rúta: 2

Fjöldi 30 manna rúta: 2

Fjöldi 20 manna rúta: 1

2 stk 60 manna rúta

2 stk 30 manna rúta

1 stk 20 manna rúta

### Alfadaemi 12 - Mylla (tic - tac - toe)

Skrifið forrit sem leikur leikinn Myllu (tic - tac - toe). Leikurinn er leikinn á 3x3 borði. Tveir leikmenn leika leikinn. Fyrri leikmaður (leikmaður a) teiknar hring í einhvern af hinum 9 reitum á borðinu. Leikmaður b setir þá kross í annan reit. Reitirnir fá hnitin 1A, 1B, ..., 3B, 3C.

Leikmennirnir velja reit með því að tilgreina hnitin á reitnum, en forritið sem þið gerið teiknar borðið með því að nota o fyrir hring x fyrir kross og . fyrir reit sem ekki er búið að velja.

Hafi t.d. leikmaður a valið reytinn 2B og leikmaður b valið reytinn 1A er borðið teiknað

x . .

. o .

. . .

Leikmaður sem fær þrjú tákn í röð lárétt, lóðrétt eða í hornalínu vinnur leikinn, en forritið á að greina og tilkynna hvor leikmaðurinn vinnur leikinn.

### Alfa dæmi 13

Búa á til forrit sem líkir eftir því að einstaklingur A er að leggja inn pening á bankabók en einstaklingur B er að taka út pening af sömu bók. Nota á tvo þræði í þessu forriti. Einn þráður á að hafa leyfi til að uppfæra stöðu á bankabókinni hverju sinni. Vinnslan fer þannig fram að viðkomandi þráður fær leyfi til að uppfæra stöðu, skrifar á skjá að uppfærsla sé í gangi, bíður í random 0-5 sekúntur þar til hreyfing framkvæmd og skilar síðan leyfi til að uppfæra stöðuna. Báðir þræðirnir framkvæma þessa vinnslu í um það bil eina mínútu.

### Alfa dæmi 14 - Svigasamsetning

Búa á til forrit sem tekur við strengjum sem samanstanda af svigunum () og []. Forritið þarf að staðfesta hvort svigarnir í strengunum séu réttir.

Strengirnir eru sagðir réttir ef eftirfarandi skilyrði eru uppfyllt.

1. Ef A og B eru réttir strengir, þá er AB einnig réttur.

2. Ef A er réttur strengur, þá er (A) og [A] einnig rétt.

Þetta má einnig segja með því að ef við höfum opnum innri sviga í ytri sviga þá verðum við að loka innri sviganum áður en við lokum ytri sviganum. Einig getum við haft sviga sem opnast og lokast við hliðina á hverjum öðrum.

*Inntak:*

Intakið á eingöngu að taka við strengjum sem innihalda svigana () og [].

*Úttak:*

"Rett" ef strengurinn er réttur, en "Rangt" ef hann er það ekki.

*Dæmi um Inntak:*

Dæmi 1:

([])

Dæmi 2:

(([]))

Dæmi 3:

([[]])

*Dæmi um Úttak:*

Dæmi 1:

Rett

Dæmi 2:

Rangt

Dæmi 3:

Rett

### **Alfa dæmi 15 - Minesweeper**

Búa á til forrit sem hermir eftir virkni tölvuleiksins Minesweeper. Minesweeper er leikur þar sem finna þarf jarðsprengjur sem eru huldnar á  $M \times N$  fleti. Til þess að notandinn geti fundið sprengjurnar á fletinum eru reitirnir merktir með tölum sem segja til um fjölda jarðsprengja sem umliggja reitinn.

Dæmi um flöt af stærðinni  $4 \times 4$  og sem inniheldur tvær jarðsprengjur myndi vera:

```
* . . .
. . . .
. * . .
. . . .
```

Hér fyrir ofan eru sprengjurnar táknáðar með '\*' en auðir reitir með '.'.

Þegar búið er að merkja reitina lítur flöturinn út á eftirfarandi hátt:

```
*100
2210
1*10
1110
```

Takið eftir að hver jarðsprengja er umkringd allt að 8 reitum sem þarf að passa upp á tölurnar á.

*Inntak:*

Hér þurfa að koma tvær tölur sem segja til um hve stór flöturinn er.

Þar á eftir þarf að slá inn mynd af fletinum þar sem auðir fletir eru slegnir inn sem punktar og jarðsprengjur með stjörnum.

### Úttak:

Úttakið úr forritinu er mynd af fletinum þar sem búið er að merkja alla auða reiti með fjölda jarðsprengja sem umkringja þá.

### Dæmi um Inntak:

Dæmi 1:

```
4 4
* . . .
. . . .
. * . .
. . . .
```

Dæmi 2:

```
3 5
** . . .
. . . .
. * . . .
```

### Dæmi um Úttak:

Dæmi 1:

```
*100
2210
1*10
1110
```

Dæmi 2:

```
**100
33200
1*100
```

## Alfa dæmi 16 - Braggreinir

Verkefnið er að búa til braggreini 0.1 en braggreinirinn á að greina mögulegan bragarhátt vísna. Eins og áður segir er þetta útgáfa 0.1 þannig að greinirinn fæst ekki við að greina stuðla og höfuðstafi heldur aðeins endarím vísunnar. Forritið greinir hvort endarím passar við ferskytlu, stafhendu eða samhendu. Öll þessi form hafa fjórar línur.

Við skilgreinum að tvö orð rími ef að síðasti sérhljóðinn í orðinu og allir bókstafir fyrir aftan hann séu þeir sömu.

Forritið les vísu úr textaskrá og ef línufjöldi og endarím passa við ferskeytlu, stafhendu eða samhendu þá þá segir forritið það, en annrs segir forritið að vísan passi ekki við neinn af þessum bragháttum.

Ferskeytla hefur víxlrim (aBaB) þar sem 1og 3 lína ríma og 2 og 4 lína einnig. Dæmi um ferskeytlu er:

Fjöll í austri fagurblá  
freista dalabarnsins.  
Ungur fylgir æskuþrá  
upp til jökulhjarnsins.

Stafhenda hefur runurím (aabb) þar sem 1 og 2 lína ríma og 3 og 4 lína einnig. Dæmi um stafhendu er:

Viltu heyra væna mín,  
vísur, sem ég kvað til þín  
eina þögla þorranótt  
þegar allt var kyrrt og rótt.

Samhendur hefur samrím (aaaa) þar sem allar línur hafa endarím. Dæmi um samhendu er:

Naumast unír hyggjan heit  
hverja stund á sama reit,  
fer því oft í ljóðaleit  
langt, ef hún til fanga veit.

Fleiri dæmi um visur:

*Ferskeytlur:*

Sveimað heimahögum frá  
hef ég vors á degi,  
víða stríða þræddi þá  
þunga hraunavegi.

*Stafhendaur*  
runurím (aabb)

Svinnust hlynnir söngvum að  
sætan mæt, ég kveð um það,  
mest og bezt þitt lundarlag,  
leiðir greiðir mínum brag.

*Samhenda*  
samrím (aaaa)

Gefið hefur glaða lund  
góður óður marga stund,  
engan feng í minni mund  
met ég betri heillafund.

## Alfa dæmi 17

Tíðnigreining er ein leið sem notuð er til að ráða dulmál. Tíðnigreiningin byggir á að finna hvaða bókstafir í tungumálinu eru oftast notaðir. Búið til forrit sem les inn textaskrá en forritið birtir öll tákni sem koma fyrir í skránni, fyrst kemur sá bókstafur sem er algengastur, síðan sá sem er næst algengastur o.s.frv. Ef tveir stafir eru jafn algengir skiptir ekki máli í hvaða röð þeir koma. Ekki er gerður greinarmunur á há og lágstöfum. Forritið sleppir tölustöfum, og táknum sem eru ekki bókstafir.

*Dæmi um virkni:*

Ef textaskráin inniheldur eftirfarandi texta:

Örninn smíðar Ægisdýr,  
álftin járníð rekur,  
hvalir mjólka hross og kýr,  
hákarl grösín tekur.

r 12  
i 6  
n 6  
s 5  
a 4  
k 4  
l 4  
á 3  
g 3  
h 3  
ð 2  
e 2  
j 2  
u 2  
t 2  
ý 2  
ö 2  
m 2  
o 2  
d 1  
f 1  
í 1  
ó 1  
v 1  
æ 1

### Alfa dæmi 18

Nonni er sérvitringur sem við vinnum fyrir (en hann borgar vel svo við gerum hvað sem er fyrir hann)

Nonni vill að við búum til forrit sem les inn textaskrá. Forritið á að raða orðunum í skránni í íslenska stafrófsröð en hann vil að röð stafanna sé eftirfarandi.

áabcdéefghíjklmnoóprstúuvxýyzþæö

ef skráin t.d. inniheldur eftirfarandi texta:

*aáli áali boýsa boyam*

raðast þau:

áli  
aáli  
boýsa  
boyam

### Alfa dæmi 19

Ein dulkóðunaraðferð sem notuð hefur verið í gegn um tíðina er að skipta bókstöfum fyrir aðra bókstafi. Til að ákveða hvernig skipt er á táknum er notað ákveðið lykilorð. Aðferðin er eftirfarandi, en hér er miðað við að við ætlum að dulkóða tákn í enska stafrófinu. Við byrjum á að skrifa öll táknið í enska stafrófinu:

a b c d e f g h i j k l m n o p q r s t u v w x y z

síðan tökum við lykilorðið okkar, en við ætlum að nota lykilorðið JULIUS CAESAR. Við þörum saman fremstu táknið í stafrófinu við táknið í lykilorðinu, en ef sami bókstafur kemur tvisvar fyrir í lykilorðinu þá sleppum við honum í síðara sinnið sem hann kemur, þá erum við komin með:

a b c d e f g h i j k l m n o p q r s t u v w x y z  
J U L I S C A E R

þá verður a dulkóðað með j, b með u o.s.frv. Að lokum höldum við áfram með stafrófið á eftir lykilorðinu en notum eingöngu þá stafi sem á eftir að nota. Þá fáum við eftirfarandi:

a b c d e f g h i j k l m n o p q r s t u v w x y z  
J U L I S C A E R T V W X Y Z B D F G H K M N O P Q

þannig verður t.d. a dulkóðað með j, m er dulkóðað með x og w dulkóðað með n.

Verkefni okkar er að búa til forrit sem tekur við lykilorði og texta sem á að dulkóða en forritið birtir hvernig textinn er eftir dulkóðun.

Dæmi um dulkóðun skv. þessari aðferð:

Lykilorð: FORRITUNARKEPPNI

Vörpun tákna:

a b c d e f g h i j k l m n o p q r s t u v w x y z  
f o r i t u n a k e p q s v w x y z b c d g h j l m

eldgos -> tqinwb

tannkul -> cfvvpdq

Verkefni okkar er að búa til forrit sem tekur við lykilorði og texta sem á að dulkóða en forritið birtir hvernig textinn er eftir dulkóðun.

### Alfa dæmi 20 - Túlkur

Viss tölva er með 10 vistföng (e. register) og er með 1000 minnishólf í minni. Bæði minnishólfin og vistföngin geta aðeins haldið tölum af stærðinni 0 til 999. Skipanirnar sem þessi tölva getur tekið við eru þriggja stafa tölur og eru eftirfarandi:

- \* 100 þýðir að tölvan eigi að hætta vinnslu.
- \* 2dn þýðir að talan n verður sett í vistfang d. (Athugið að n er á milli 0 og 9)
- \* 3dn þýðir að leggja töluna n við töluna í vistfangi d.
- \* 4dn þýðir að margfalda töluna n með tölunni í vistfangi d.
- \* 5ds þýðir að afrita töluna úr vistfangi s í vistfang d.
- \* 6ds þýðir að leggja töluna úr vistfangi s við töluna í vistfangi d.
- \* 7ds þýðir að margfalda töluna í vistfangi s með tölunni í vistfangi d.
- \* 8da þýðir að afrita töluna úr minnishólfinu sem vistfang a vísar á yfir í vistfang d.
- \* 9sa þýðir að afrita töluna í vistfangi s í minnishólfið sem vistfang a vísar á. (Dæmi: Ef a inniheldur töluna 050, þá á að afrita töluna úr minnishólfi 50 yfir í vistfang a)
- \* Ods þýðir að tölvan eigi að færa sig yfir í það minnishólf sem vistfangið d bendir á, nema ef vistfang s innihaldi 000.

Í upphafi eiga öll vistföng og minnishólf að innihalda 000.

Minnishólfin eru síðan fyllt með skipununum sem eru lesnar inn í forritið.

Tölvan byrjar að keyra skipunina sem er í minnishólfi 0, síðan keyrir hún skipunina úr minnishólfi 1, o.s.frv, nema þegar skipunin er Ods.

Allar útkomur úr aðgerðunum hér fyrir ofan eiga að vera modulus 1000 til þess að tryggja að allar skipanir/tölurnar séu á forminu 0 til 999.

*Inntak:*

Hér eru skipanirnar/forritið sem tölvan á að keyra slegnar inn. Á hverri línu eiga að vera ein skipun fyrir hvert minnishólf. Þegar búið er að slá inn allar skipanirnar, þá slær notandinn inn "go".

Fyrsta línan/skipunin fer í minnishólf 1 og næsta lína/skipun fer í minnishólf 2, o.s.frv.

*Úttak:*

Forritið prentar að lokum út hve margar skipanir voru keyrðar af tölvunni.

*Dæmi um Inntak:*

299  
492  
495  
399  
492  
495  
399  
283  
279  
689  
078  
100  
000  
000  
000  
go

*Dæmi um Úttak:*

16