

Java

Obligatory Exercises Week 19

Anders “Bongo” Bjerg Pedersen

17. maj 2006

Exercise 1

Create a program that reads a string from the command line. The program should parse the string and count the number of occurrences of ASCII characters in it. Finally the program must output a list of characters with frequency larger than 0. The program must be robust i.e. not crash on any input.

Allerede inden vi begynder kodningen af programmet, er der et par særlige tegn, vi er nødt til at tage hensyn til. Bl.a. er terminalen ikke glad for karakterer som ! og & (lidt afhængig af styresystem). Dette løses ved at sætte argumentet (strengen der skal tælles op) i ”enkeltgåseøjne”. Dette fortæller terminalen, at den ikke skal tage hensyn til specielle tegn i argumentet efter programnavnet. Desuden antager vi, at vi kun skal se på én streng (altså ét argument). Programmet kan let udvides til flere, men f.eks. mellemrum vil kunne komme med vha. enkeltgåseøjne, da de ellers ville indikere begyndelsen af et nyt argument.

Selve koden kommer så her:

```
class aflu19 {
    public static void main(String[] args) {

        String s = args[0];           //Gemmer strengen fra terminalen i s
        int[] a = new int[267];       //Opretter array af størrelse 267

        for (int i=0;i<s.length();i++) //Lægger én til hver position i a,
            a[s.charAt(i)]++;         //hvor en karakter findes

        for (int i=0;i<266;i++)       //Udskriver strengens karakterer og antal
            if (a[i]!=0)
                System.out.println((char)i+" forekommer "+a[i]+" gang(e)");
    }
}
```

Det grundlæggende idé er altså at oprette et array *a* med én plads til hver ASCII-karakter. Programmet identificerer derefter hver karakter én for én

og lægger 1 til den pågældende plads i a . Til slut udskrives alle de pladser (karakterer) i a , der har antal større end 0.

Jeg har ikke kunnet finde nogen måder at køre programmet i sænk på...men for lige at vise, at det virker (bemærk gåseøjnene omkring strengen):

```
java aflu19 'AnBjePe 123_!"#%&/()=?+[\ ] , . - ; : * $ '
  forekommer 1 gang(e)
! forekommer 1 gang(e)
" forekommer 1 gang(e)
# forekommer 1 gang(e)
$ forekommer 1 gang(e)
% forekommer 1 gang(e)
& forekommer 1 gang(e)
( forekommer 1 gang(e)
) forekommer 1 gang(e)
* forekommer 1 gang(e)
+ forekommer 1 gang(e)
, forekommer 1 gang(e)
- forekommer 1 gang(e)
. forekommer 1 gang(e)
/ forekommer 1 gang(e)
1 forekommer 1 gang(e)
2 forekommer 1 gang(e)
3 forekommer 1 gang(e)
: forekommer 1 gang(e)
; forekommer 1 gang(e)
= forekommer 1 gang(e)
? forekommer 1 gang(e)
A forekommer 1 gang(e)
B forekommer 1 gang(e)
P forekommer 1 gang(e)
[ forekommer 1 gang(e)
\ forekommer 1 gang(e)
] forekommer 1 gang(e)
_ forekommer 1 gang(e)
e forekommer 2 gang(e)
j forekommer 1 gang(e)
n forekommer 1 gang(e)
{ forekommer 1 gang(e)
} forekommer 1 gang(e)
```