ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Lecturer: Prof. B. Gärtner

Informatik für Mathematiker und Physiker HS13

Exercise Sheet 6

Submission deadline: 15:15 - Tuesday 29th October, 2013

Course URL: http://www.ti.inf.ethz.ch/ew/courses/Info1_13/

Assignment 1 – Skript-Aufgabe 96 (4 points)

Write a program `frequencies.cpp` that reads a text from standard input and outputs the frequencies of the letters in the text, where we do not distinguish between lower and upper case letters. For this exercise, you may assume that the type `char` implements ASCII encoding. This means that all characters have integer values in $\{0, 1, \ldots, 127\}$. Moreover, in ASCII, the values of the 26 upper case literals `'A'` up to `'Z'` are consecutive numbers in $\{65, \ldots, 90\}$; for the lower case literals `'a'` up to `'z'`, the value range is $\{97, \ldots, 122\}$.

Running this on the lyrics of *Yesterday* (The Beatles) for example should yield the following output.

```
Frequencies:
a:     45 of 520
b:     5 of 520
c:     5 of 520
d:     28 of 520
e:     65 of 520
f:     4 of 520
g:     13 of 520
h:     27 of 520
i:     27 of 520
j:     0 of 520
k:     3 of 520
l:     20 of 520
m:     10 of 520
n:     30 of 520
o:     43 of 520
p:     4 of 520
q:     0 of 520
r:     19 of 520
s:     36 of 520
t:     31 of 520
u:     9 of 520
v:     6 of 520
w:     19 of 520
x:     0 of 520
y:     34 of 520
z:     0 of 520
Other: 37 of 520
```

You find the file `yesterday.txt` on the course webpage. Download this file and store it in the same directory as your program is located. Then provide it as input to your program by starting your program as follows: `./frequencies < yesterday.txt`.

Assignment 2 – Skript-Aufgabe 93/94 (8 points)

a) Write a program `read_array.cpp` that reads a sequence of n integers from standard input into an array (realized by a vector). The number n is the first input, and then the program expects you to input another n values. After reading the n values, the program should output them in the same order. For example, on input `5 4 3 6 1 2` the program should output `4 3 6 1 2`.

b) Enhance your program from part a) so that the resulting program `sort_array.cpp` sorts the array elements into ascending order before outputting them. For example, on input `5 4 3 6 1 1` the program should output `1 1 3 4 6`.

   Your sorting algorithm does not have to be particularly efficient, the main thing here is that it works correctly. Test your program on some larger inputs (for instance sort the files `sort1.txt` and `sort2.txt` from the course webpage).

c) Let's try to analyze your algorithm form part b). Especially, we are interested in the number of comparisons (<, <=) it has to compute. For this, try to determine (for fixed n) the *worst possible input* to your algorithm, i.e. a sequence of n numbers, where among all possible sequences of n numbers the numbers of comparisons is largest. Count the number of comparisons your algorithm needs for this sequence.

Assignment 3 – Skript-Aufgabe 91 (4 points)

Let us call a natural number k-*composite* if and only if it is divisible by exactly k different prime numbers. For example, prime powers are 1-composite, and $6 = 2 \cdot 3$ as well as $20 = 2 \cdot 2 \cdot 5$ are 2-composite. Write a program `k_composite.cpp` that reads numbers $n \geq 0$ and $k \geq 0$ from the input and then outputs all k-composite numbers in $\{2, \ldots, n-1\}$. How many 7-composite numbers are there for $n = 1,000,000$?

Challenge - Skript-Aufgabe 90

The two logicians Mr. Sum and Mr. Product are friends who one day have a phone conversation. Before, Mr. Sum only knows the sum $s = a + b$ of two unknown integer numbers $1 < a < 100$ and $1 < b < 100$. Mr. Product, on the other hand, only knows the product $p = a \cdot b$ of a and b. The conversation goes as follows:

Mr. Product:	"I don't know the numbers a and b."
Mr. Sum:	"I knew that you don't know them."
Mr. Product:	"Ah... but now I know them."
Mr. Sum:	"Then I know them too, now."

What is the set of numbers $\{a, b\}$? You have to assume, of course, that all statements made during this conversation are true. Write a program that computes $\{a, b\}$, or submit a convincing written solution!