Assignment 2 - Software I, Spring 2003 (0368-2157-9,0368-2157-12)

`http://www.cs.tau.ac.il/~efif/courses/software1`

Due: Mars 25, 2003

Before starting to answer the questions, please read very carefully the "Submission Guidelines"[1]. Make sure your programs detect invalid input data, and print out appropriate error messages. Do not add "friendly" messages to your programs, as they are tested automatically by other programs.

Ex 2.1 dec2eng

Write a program that converts the decimal representation of a non-negative integer to the base-27 representation. The digits used in the base-27 representation are the space ' ' and the English letters 'a' through 'z'. The space represents 0, 'a' represents 1, 'b' represents 2,..., and 'z' represents 26.

$$l_n, l_{n-1}, ..., l_0 = \sum_{i=0}^{n} (l_i \times 27^i)$$

Each place value to the left is equal to 27 times the place value to the right, which implies that each place value to the right is equal to the place value to the left divided by 27.

The program should scan in a non-negative integer in decimal format, and print out the number in base-27 format (in English letters). You may assume that the input number is smaller than 2^{32}. Examples:

$$\begin{aligned}
\texttt{a} &= 1 = 1 \\
\texttt{aaaaaa} &= 1 \times 27^5 + 1 \times 27^4 + 1 \times 27^3 + 1 \times 27^2 + 1 \times 27^1 + 1 \times 27^0 = 14900788 \\
\texttt{abcdef} &= 1 \times 27^5 + 2 \times 27^4 + 3 \times 27^3 + 4 \times 27^2 + 5 \times 27^1 + 6 \times 27^0 = 15473895 \\
\texttt{hello} &= 4359030
\end{aligned}$$

Ex 2.2 maxbin2eng

Write a program that reads the number of digits in a binary format, and prints out the largest number that can be represented in this format, but it prints it out in base-27 format (in English letters). Example:

```
maxbin2eng
8
il
```

[1] `http://www.cs.tau.ac.il/~efif/courses/software1`

Ex 2.3 dice

Modify the dice-rolling program so that it computes:

1. the floating-point average μ of all the rolls of the pair of dice
$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}$$
2. the standard deviation s, which can be expressed as
$$s = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Add the line

```
#include <math.h>
```

at the top of the source file, and link with the math library (add `-lm` to the command line) to use the function `sqrt()`.

The program should read the random-generator seed and print out the average followed by the standard deviation. Use `double` type to represent any real number in the program, and print out the result with 6 digits of precision for the fraction part.

Ex 2.4 crystal

You have b identical crystal balls. There is a high rise with a f floors. One of the floors could be critical. When a crystal ball is dropped from the critical floor or from a higher floor the ball breaks. You need to find the critical floor with the minimum number of experiments in the worst case. Each drop is counted as one experiment regardless of whether the ball breaks or not.

If you have only one ball, you must start your experiments from the first floor. In the worst case you will end up performing an experiment every floor just to find out that there is no critical floor. Given one ball the minimum number of experiments required in the worst case is f.

This is a riddle, that you are going to solve with a program, but first let's count the maximum number of floors that can be covered, given an allowed number of experiments e.

Let $f(e, b)$ denote the maximum number of floors we can cover with e experiments and b balls.

If $b = 0$ or if $e = 0$, we cannot do much. Thus,

$$f(e, 0) = 0$$
$$f(0, b) = 0$$

Suppose we are allowed e experiments with b balls. Let e' be the maximum number of floors we can cover with $e - 1$ experiments and $b - 1$ balls. Then, the best we can do is perform the first experiment at the floor $e' + 1$. If the ball breaks, we are down to $e - 1$ experiments and $b - 1$ balls, and this is exactly what we need to cover all the floors bellow e'. Otherwise, we can proceed and cover $f(e - 1, b)$ additional floors. This leads to the following recursing formula:

$$f(e, b) = f(e - 1, b) + f(e - 1, b - 1) + 1$$

Write a program that reads the number of floors f and the number of b balls, and prints out the minimum number of experiments required. You may do it the hard way. That is, write the function $f(e, b)$, and look for the first $\hat{e} = 1, 2, \ldots, \infty$ that satisfies $f \leq f(\hat{e}, b)$.

Ex 2.5 palindrome

Write a program that reads an input line, determines whether it is a palindrome or not, and prints out "true" or "false" accordingly.

A palindrome is a string that is the same when read backward or forward.

Disregard spaces, non alphabetic and numeric characters, and ignore case distinctions. For example, the following are palindromes:

```
Was it a rat I saw?
Dennis and Edna sinned.
step on no pets
A Fool, A Tool, A Pool; LOOPALOOTALOOFA!
Madam, I'm Adam.\\
A man, a plan, a cat, a ham, a yak, a yam, a hat, a canal-Panama!
```

The program should disregard characters beyond the first 80, and terminate upon EOF as well as the end of the line.

Good Luck!

More Information on the Submission

Files Name

The files for the exercises should be located under `~/software1/assign2`, and their names should match the name of the exercise. For example, the `C` source file and corresponding executable for exercise 2.1, namely dec2eng, should be `~/software1/assign2/dec2eng.c` and `~/software1/assign2/dec2eng` respectively. Note that names are case sensitive (i.e. `ex1.C` is different than `ex1.c`).

Giving Permission to the Files

Before submitting the solution set, please give permission to the files by executing the following command:

```
chmod 705 ~ ~/software1 ~/software1/assign2 ~/software1/assign2/*
```