IC 150 P Computation for engineers lab
Lab assignment sheet no: 9, Odd semester, 2016
Structures, file operations, and using `rand()`

Prepared by: Maben Rabi

Objective for this lab session

- In this lab session, you will learn to use structures.

- You will also learn to use the `<stdlib.h>` function `rand()`, and to plot simple diagrams using the `printf` command.

- Get some more practice with manipulating strings.

Use of `rand()`, and creating ASCII-art diagrams :

The `rand()` functions is used to generate a uniformly distributed random number between 0 and the integer `RAND_MAX`. Repeated calling of this function produces a pseudorandom sequence of 'independent' samples of the same random variable. For more information, see:
`www.cs.yale.edu/homes/aspnes/pinewiki/C(2f)Randomization.html`
For illustration of this, and also of using the `printf` command to create simple ASCII-art based diagrams, see: `https://en.wikipedia.org/wiki/ASCII_art` and the sample code on pages 3, 4.

Task 1:

In the last assignment sheet, you were given 5 protein sequences in the file ''`seqs.faa`''. You must write a program that will read each protein sequence and compute:

A List the top *k* amino acids in descending order of occurrence in each line. Draw an ASCII-art histogram for each line. User inputs the integer *k*.

B User inputs a list of sequences of amino acids. For each protein, your program must draw a histogram of occurrence frequencies.

You are asked to store the read symbols for amino acids, and their occurrence counts in an array of structures, line suggested below:

```
1   struct aminoAcid {
2       char            letterName;
3       unsigned int   numOfOccurrences;
4   };
5   struct aminoAcidsInProtein[26]; /* at most 26 upper case letters
        */
```

<div align="center">fewLinesForStructExample.c</div>

Your histogram must be informative, and user-friendly. For example, you histogram can look like:

```
The protein CCQ76313.1 has the following amino acids
as the top four most frequently present in it sequence:

Amino acid V ****************                    [40 %]
Amino acid I *******                            [20 %]
Amino acid A ******                             [18 %]
Amino acid M ***                                [ 9 %]
```
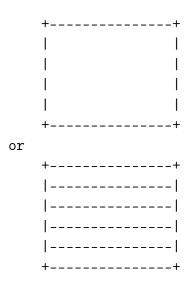
Task 2: Drawing axis-parallel rectangles

An axis parallel rectangle is a rectangle whose four corners take the special form of coordinates:

$$(x, y), (x + a, y), (x + a, y + b), (x, y + b).$$

A Write a program that will draw an ASCII diagram of a user specified axis-parallel rectangle. Depending on the user's wish, you must draw the rectangle in outline, or filled. For example:

```
        +---------------+
        |               |
        |               |
        |               |
        |               |
        +---------------+
    or
        +---------------+
        |---------------|
        |---------------|
        |---------------|
        |---------------|
        +---------------+
```

You must use a structure for storing a point, and a structure for storing an axis parallel rectangle. The structure for a point must have as its members the x, y coordinates of the point. The structure for storing an axis parallel rectangle must have as its members two diagonally opposite points.

B User inputs a list of axis-parallel rectangles. Your program must draw the intersection of all of these rectangular sets.

Note: You must adopt a legend and scale that is tuned to the coordinates of rectangles specified by the user. That means you the programmer must decide what coordinates to assign to the corners of the display, AFTER processing the user's inputs.

Task 3: Simulating, and tabulating histograms for an average

Consider a random variable X with mean μ. Let

$$X_1, X_2, X_3, \ldots$$

be a sequence of 'independent' random variables, each having the same probability density function as the random variable X. For some positive integer N, consider the sums:

$$S_{\text{average}} = \frac{1}{N} \sum_{k=1}^{N} X_i$$

We will experimentally find some patterns in the probabilistic behaviour of this sum. Take the random variable to the uniform random variable between 0 and 1. For this choice, we get $\mu = 0.5$. User inputs positive integer N. User also enters an odd, positive integer $2l + 1$. Your program must print on screen an ASCII-art histogram with the bins:

bin 1	S_{average} falls between 0 and $1/(2l+1)$
bin 2	S_{average} falls between $1/(2l+1)$ and $2/(2l+1)$
$\vdots$	$\vdots$
bin l+1	S_{average} falls between $l/(2l+1)$ and $(l+1)/(2l+1)$
$\vdots$	$\vdots$
bin 2l	S_{average} falls between $(2l-1)/(2l+1)$ and $(2l)/(2l+1)$
bin 2l+1	S_{average} falls between $(2l)/(2l+1)$ and 1

Note: You must create an array of structures. Each structure in it must contain as members the bin number and the corresponding frequency of $S_{average}$ falling within this bin.

To get each sample of the random variable S_{average}, you need N samples of the random variable X. The problem is asking you to compute several samples of random variable S_{average} and compute the histogram. Let the user decide the number of samples of S_{average}.

Sample code

```
1  #include<stdio.h>
2  #include<math.h>
3  #include<stdlib.h>
4
5  #define MAX_WIDTH  65 /* default terminal on ubuntu has a width of
       about 70 */
6
7  void main(){
8      unsigned int  count,  numOfSamples, numBelowHalf, percent;
9      float  dummy, fractionBelowHalf, scale;
10
11     printf("%d\n",RAND_MAX);
12     printf("This program generates independent samples of\n");
13     printf("a uniform random variable bewteen 0, 1.\n\n");
14     printf("Enter the number of independent samples:  ");
15     scanf("%u",&numOfSamples);
16
17     numBelowHalf = 0;
18     for (count=1 ; count <= numOfSamples ;  count++){
19       dummy = ((float) rand())/(float) RAND_MAX;  /* RAND_MAX is
             defnied in stdlib.h */
20         if (dummy <= 0.5) {
```

```
21        numBelowHalf = numBelowHalf + 1;
22        }
23      }
24    fractionBelowHalf =  (float) numBelowHalf / (float)
         numOfSamples;
25    percent = (unsigned int) ( (float) 100 * (float)
         fractionBelowHalf );
26
27    scale = (float) MAX_WIDTH ;
28    printf("scale=%f, fractionBelowHalf=%f
         \n",scale,fractionBelowHalf);
29
30    printf("\n");
31    for (count=1 ; count <= MAX_WIDTH ;  count = count + 1){
32      if ( (fractionBelowHalf * scale) > ((float) count ) ) {
33      printf("*");
34      }
35      else {
36      printf(" ");
37      }
38    }
39    printf("  %u \% \n", percent );
40 }
```

randomNumbersAndHistogram.c

If you would like to find out from inside your C program, what the terminal width is, then use something like:

```
1  /* authored by LainIwakura
2     http://stackoverflow.com/questions/6657048/
3     how-can-i-find-the-number-of-terminal-columns-from-a-c-c-program
4  */
5
6  #include <sys/ioctl.h>
7  #include <stdio.h>
8
9  int main()
10 {
11     struct winsize w;
12     ioctl(0, TIOCGWINSZ, &w);
13
14     printf("lines %d\n", w.ws_row);
15     printf("columns %d\n", w.ws_col);
16     return 0;
17 }
```

gettingTerminalWidth.c