BIL 105E – Introduction to Scientific and Engineering Computing (C)

Spring 2015-2016

Homework 2

CRN:21834

Yunus Güngör

No:150150701

Date:10.04.2016

Introduction

This project's main aim is to create a trustable random generation function. Project includes and analyzes two different random number generators and creates histograms of the created numbers.

Development Environment

This program has only 1 source code file written in C:

150150701.c

This program tested and compiled in following system:

gcc 4.8.5 20150623 on Red Hat 4.8.5-4 (ITU SSH Server)

gcc compiler has been used to compile the program by the command:

```
gcc 150150701.c -o hw2
```

Important Variables

whichRNG: Holds info about which algorithm should be used. 1 is mid-rng algorithm and 2 is lsd-rng algorithm.

range1…range5: Has the variables necessary to create the histogram.

number: this variable acts as seed, and produced random variable

Program Flow

Pseudo code of the program:

```
int function main
{
do
{
print "Which PRNG Algorithm?\n  1.MID-PNRG\n    2.LSD-PNRG\n";
read whichRNG;
}
while whichRNG is not 1 or not 2;
initialize range1,range2,range3,range4,range5 to 0;
take_samples(whichRNG,range1,range2,range3,range4,range5);
draw_histogram(range);
```

```
}

void function mid_rng
{
     initialize square to square of the number;
     if square has 9 digits;
          skip first two digits of square and set number to the
following five digits of square;
     else
          skip first three digits of square and set number to the
following five digits of square;
     endif;
}

void function lsd_rng
{
     initialize product to 73 times number;
     set number to first five digits of product;
}

void function take_samples
{
     if whichRNG is 1//mid-rng
     {
          do
          {
               print Enter the seed:;
               read number;
          }
          while number does not have 5 digits
          print "Enter the Number of Samples:";
```

```
            read num_samples;
            for(i=0;i<num_samples;i++)
            {
                  mid_rng(number);
                  map(number mod 500, range1, range2, range3, range4,
range5);
            }
      }
      else //lsd-rng
      {
            do
            {
                  print "Enter the seed:";
                  read number;
            }
            while number does not have 5 digits AND number is even AND
number ends with 5
            print "Enter the Number of Samples:";
            read num_samples;
            for(i=0;i<num_samples;i++)
            {
                  lsd_rng(number);
                  map(number mod 500, range1, range2, range3, range4,
range5);
            }
      }
}

void funciton draw_histogram
{
      sum=range1+range2+range3+range4+range5;
```

```
range1=range1/sum*100;
print "0...100:";
for(i=0;i<*range1;i++)
      print "*";
print "(%range1)\n";

range2=range2/sum*100;
print "101..200:";
for(i=0;i<*range2;i++)
      print "*";
print "(%range2)\n";

range3=range3/sum*100;
print "201...300:";
for(i=0;i<*range3;i++)
      print "*";
print "(%range3)\n";

range4=range4/sum*100;
printf("301...400:");
for(i=0;i<*range4;i++)
      print "*";
print "(%range4)\n";

range5=range5/sum*100;
printf("401...500:");
for(i=0;i<*range5;i++)
      print "*";
print "(%range5)\n";
return;
```

```
}

/**        maps number such as
           range1=1..100,
           range2=101..200,
           range3=201..300,
           range4=301..400,
           and range5=401..500
*/

void function map
{
     if number smaller than 100
          add 1 to range1;
     else if number smaller than 200
          add 1 to range2;
     else if number smaller than 300
          add 1 to range3;
     else if number smaller than 400
          add 1 to range4;
     else if number smaller than 500
          add 1 to range5;
     endif;
}
```

Conclusion

This project helped me to understand random number generation functions and their properties and also it taught me to test a function by histograms.

One obstacle I have faced was not being able to use the arrays, there is lots of unnecessary lines that can be replaced with a few lines by using arrays.