Student Name and Surname: Sercan Şahan
Student Number: 150140005
Lesson Code: BIL105E
Lesson Name: Introduction to Scientific and Engineering Computing (C)
Homework 2
Submission Date: 10.04.2016

Aim of this homework is programming a program that consists of a main function, two Pseudo Random Number Generators (Mid-Prng and Lsd-Prng) and a printing function (printhistogram).

Development environment of this homework consists of Dev C++ IDE and TDM-GCC 5.1.0 Compiler on Windows 10 Home 64-Bit Operating System, and Geany IDE and GCC Compiler on Ubuntu 15.10 64-Bit Operating System. This homework includes one .c file named 151040005_h2.c. This .c file consist of 4 functions; main, midprng, lsdprng and printhistogram. Flowcharts of this program are at the end of this report and in zip file as ".vsdx" and ".jpg" files.

Firstly, in main function, the program asks user for function number, which decides Mid-Pnrg or Lsd-Prng to use. After that, depending on function, program asks user for inputs, seed number(seed) and sample number(numsample) respectively. In both functions, program checks whether seed is a five-digit number. If user did not enter a five-digit number, program asks user to enter a five-digit number until he/she enters a valid number. For function 2, which is Lsd-Prng, program checks whether seed number is even or ends with 5. If this condition is true, which means number is even or ends with 5, program asks for input again, until user enters a valid number. If user enters a valid number, program asks for numsample, which used to decide how many times functions run. Since there is no condition for this number, program uses the number user entered.

In Function 1, which is Mid-Prng, as long as numsample is bigger than zero (which is a while loop), program takes square of seed number, which is assigned to square variable. After that, program takes middle five digit of square, if square has even numbered digits, the digits removed from right-end side is one more than the digits removed from left-end side. The final five-digit number is assigned to newnumber2 and seed variables. newnumber2 variable is used to obtain range from 1 to 500. There are five ranges which are 1-100(range1), 101-200(range2), 201-300(range3), 301-400(range4) and 401-500(range5). After deciding which range

newnumber2 belongs to, number of range that newnumber2 belongs to is incremented by 1. After that if new seed number has less digits than 5, number of digits are incremented by multiplying it by powers of 10. After that numsample is decremented by one and while loop restarts with new seed number and continues until numsample becomes zero. After while loop is finished, program sums up ranges of numbers to get total of ranges, which is assigned to total_range variable. To calculate percentages of ranges, range variables (range1, range2, range3, range4 and range5) are multiplied with 100 and divided by total_range variable. After that name of range and percentage of range is printed on screen with histogram of percentage. To print histogram, printhistogram function is used.

In Function 2, which is Lsd-Prng, as long as numsample is bigger than zero (which is a while loop), program multiplies seed number with 73 and assigns it to product variable. To find range of seed, five digits at right-end side is taken and assigned to number variable. After that number variable is used to obtain range from 1 to 500. There are five ranges which are 1-100(range1), 101-200(range2), 201-300(range3), 301-400(range4) and 401-500(range5). After deciding which range number belongs to, number of range that number belongs to is incremented by 1. After that value of number variable is assigned to seed variable and numsample is decremented by 1. After that while loop restarts with new seed variable and continues until numsample reaches 0. After while loop is finished, program sums up ranges of numbers to get total of ranges, which is assigned to total_range variable. To calculate percentages of ranges, range variables (range1, range2, range3, range4 and range5) are multiplied with 100 and divided by total_range variable. After that name of range and percentage of range is printed on screen with histogram of percentage. To print histogram, printhistogram function is used.

In printhistogram function, an integer is taken from another function. As long as percentage variable is bigger than 0, program prints "*" character side-by-side and decrements percentage variable by 1. When value of percentage variable reaches zero, function goes to next line on screen and finishes function.

In conclusion, this project aims to teach defining functions and calling them in another function. The most difficult part of this project is removing digits from square integer since it can take different digit values. Therefore, I had to think all of these situation and had to write a long midprng function.

```
                              Start

              printf("Choose PRNG function:\n1- MID-
                     PRNG\n2- LSD-PRNG\n")

                          Input(function)

         While( function != 1 && function != 2)        False

                              True

             printf("You choosed an invalid function. Please
             enter a valid function number.\n\n Choose PRNG
                   function:\n1- MID-PRNG\n2- LSD-PRNG\n")

                          Input(function)

                   printf("\nEnter the 5 digit seed
                              number:\n")

                          Input(seed)

      True              if(function == 1)          else if (function == 2)

   while(seed > 99999 || seed < 10000)              while(seed > 99999 || seed < 10000)

        True          False                    False              True

  printf("\nPlease enter a 5    printf("\nEnter                   printf("\nPlease enter a 5
     digit number:\n")          the number of                       digit number:\n")
                                samples:\n")

        Input(seed)          Input(numsample)                       Input(seed)

                        printf("\n%lld random
                        numbers were generated
                        by using MID-PRNG\n",
                              numsample)

                                     while(seed % 5 == 0 || seed % 2 == 0)        False

                        midprng(seed, numsample)

                                          True                     printf("\nEnter
                                                                   the number of
                                                                   samples:\n")
                                    printf("\nPlease
                                    enter another 5
                                    digit number:\n")            Input(numsample)

                                                              printf("\n%lld random
                                      Input(seed)              numbers were generated
                                                              by using LSD-PRNG\n",
                                                                    numsample)

                                                              lsdprng(seed, numsample)

                              End
```

```
midprng(seed, numsample)
```

```
while(numsample > 0)  ──→  False
```

True

```
square=seed*seed
```

if(square >= 1000000000)	else if(square < 1000000000 && square >= 100000000)	else if(square < 100000000 && square >= 10000000)	else if(square < 10000000 && square >= 1000000)	else if(square < 1000000 && square >= 100000)	else if(square < 100000)

newnumber1 = square % 100000000;
 	
newnumber2 = newnumber1 / 1000;
 	
seed = newnumber2;
	newnumber1 = square % 1000000;
	
	newnumber2 = newnumber1 / 100;
	
	seed = newnumber2;
		newnumber1 = square % 10000000;
		
		newnumber2 = newnumber1 / 100;
		
		seed = newnumber2;
			newnumber1 = square % 1000000;
			
			newnumber2 = newnumber1 / 10;
			
			seed = newnumber2;
				newnumber1 = square;
				
				newnumber2 = newnumber1 / 10;
				
				seed = newnumber2;
					newnumber1 = square;
					
					newnumber2 = newnumber1;
					
					seed = newnumber2;

```
range = newnumber2 % 500 + 1;
```

if(range <= 100)	else if(range <= 200)	else if(range <= 300)	else if(range <= 400)	else if(range <= 500)
range1++	range2++	range3++	range4++	range5++

```
numsample--
```

```
printf("Range:\n")
```

```
total_range = range1 + range2 + range3 + range4 + range5
```

percentage1 = range1*100 / total_range	percentage2 = range2*100 / total_range	percentage3 = range3*100 / total_range	percentage4 = range4*100 / total_range	percentage5 = range5*100 / total_range
printf("1-100 : %%%d ", percentage1)	printf("101-200: %%%d ", percentage2)	printf("201-300: %%%d ", percentage3)	printf("301-400: %%%d ", percentage4)	printf("401-500: %%%d ", percentage5)
printhistogram(percentage1)	printhistogram(percentage2)	printhistogram(percentage3)	printhistogram(percentage4)	printhistogram(percentage5)

```
End of midprng(seed, numsample)
```

```
lsdprng(seed, numsample)
```

```
while(numsample > 0)
```

False

True

```
product = seed * 73;

number = product % 100000;

range = number % 500 + 1;
```

```
if(range <= 100)      else if(range <= 200)      else if(range <= 300)      else if(range <= 400)      else if(range <= 500)
```

True   True   True   True   True

```
range1++          range2++          range3++          range4++          range5++
```

```
seed = number;

numsample--;
```

```
printf("Range:\n")
```

```
total_range = range1 + range2 + range3 + range4 + range5
```

```
percentage1 =          percentage2 =          percentage3 =          percentage4 =          percentage5 =
range1*100 /           range2*100 /           range3*100 /           range4*100 /           range5*100 /
total_range            total_range            total_range            total_range            total_range
```

```
printf("1-100 : %%%d    printf("101-200:       printf("201-300:       printf("301-400:       printf("401-500:
", percentage1)         %%%d ", percentage2)   %%%d ", percentage3)   %%%d ", percentage4)   %%%d ", percentage5)
```

```
printhistogram(percentage1)  printhistogram(percentage2)  printhistogram(percentage3)  printhistogram(percentage4)  printhistogram(percentage5)
```

```
End of lsdprng(seed, numsample)
```

```
          ┌─────────────────────────────┐
          │  printhistogram(percentage) │
          └─────────────────────────────┘
                        │
                        ▼
                    ◇ while(percentage > 0) ◇ ──────────────►  False
                        │                                        │
                        ▼                                        │
                     True                                        │
                        │                                        │
                        ▼                                        │
                   printf("*")                                   │
                        │                                        │
                        ▼                                        │
                  percentage--                                   │
                        │                                        │
                        ◄────────────────────────────────────────
                        ▼
                   printf("\n")
                        │
                        ▼
          ┌─────────────────────────────────┐
          │ End of printhistogram(percentage) │
          └─────────────────────────────────┘
```