CS2021 ASSIGNMENT 12 (Due Date: Dec 31, 2021)

Question 1

(a) C language is a compiled language or an interpreted language?

(b) In the heading of a C program, it always includes a number of header files like below.

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

What is the purpose for adding such header files?

(c) What is the purpose for the declaration of the variables and their data types in the beginning of the program?

(d) State three C compilers which are commonly installed in a computer.

(e) Owing to let a compiler to convert the source code of ”printf” to machine code, which header file has to be included?

(f) If we would like a C compiler to ignore a command, what could we add in the program source code file?

Answer:

(a) Compiled language.

(b) Some functions, like *printf()* and *scanf()*, their subroutines are included in the header file ”stdio.h”. The header files are used for a compiler to get the codes for the functions.

(c) First, it lets the compiler to check if there is data type mismatch in a command. Second, it lets an operating system to allocate the working space for running the program successfully.

(d) For Windows operating systems, two exemplar compilers are DevC and Borland Turbo C compilers. C IDE is a compiler for MacOS. GCC is a compiler for either Linux or Unix operating system.

(e) stdio.h

(f) Add remark (or comment) in the way like the following.

```
/*
    for(i=0; i<5; i++)
        printf("*");
*/
```

The for-loop will be ignored during compilation.

Question 2

(a) What is the difference between a pseudocode program and a C program source code?

(b) A C program which can be compiled by a C compiler X. It might not be compiled by another C compiler Y. Explain the reason(s) why?

Answer:

(a) A pseudo-code is a summary of the procedure to be executed in the program. It could be a list of steps written in English or a list of steps written in programming language. A C program source code is the detail listing of the commands which have to be conformed to the syntax of the programming language.

(b) Different compilers might have slightly different design for the set of header files. For instance, the subroutine a function call, say xxx() might have been included in the abc.h of the X compiler. But, the subroutine of xxx() might have not been defined in the abc.h in the Y compiler. Thus, the source code which can be compiled by the X compiler might not be able to be compiled by the Y compiler.

Question 3

This question is about the program command in Section 6.5 in the C Programming lecture note.

(a) In the program, the header files "stdlib.h", "time.h" and "math.h" are included. Explain why they are included.

(b) In the program, two commands are added above "main()".

```
#define PI 3.14159265358979323846
float randu(void);
```

Explain why they are added and why they are added above "main()".

(c) In the function "randu()", "(float)" is put in a command.

```
randnum = (float) rand()/RAND_MAX;
```

Explain why it is added in here. If "(float)" is removed, which number will be returned by the function "randu()".

(d) What is the purpose of the following command?

```
srand(time(NULL));
```

What will be the consequence if this command is removed?

Answer:

(a) In the program, the subroutines of the function calls *srand()* and *rand()* are defined in the stdlin.h header file. The subroutine of the function call *time()* is defined in the time.h header file. The subroutines of *sqrt()*, *log()* and *cos()* are defined in the math.h header file.

(b) In the main program, there is command calling the function *randu()*. Owing to check the datatypes of the variables in the command, the compiler needs to know the datatype of the data to be returned from the function *randu()*. Thus, its declaration has to be added above 'main()'. For PI, it declares (defines) a global constant for use in any part of the program.

Question 4

Suppose the program in Section 3.2 is modified as the following.

```
#include<stdio.h>
#include<stdlib.h>

main(int argc, char *argv[])
{
int length, i, j;

length = atoi(argv[1]);
for(i = length; i >0 ; i--)
    {
    for(j = 0; j < i+1; j++)
        printf("*");
    printf("\n");
    }
}
```

The filename of the source code is now called "triangle.cpp" and the program has been compiled without error. Suppose that the following command has been typed on the command prompt.

```
C:\>triangle 4
```

(a) What is the value of "argc"?

(b) What are the contents in "argv[0]" and "argv[1]" respectively?

(c) What will you see on the screen if the following command is typed on the command prompt?

```
C:\>triangle 4
```

Please show your answer like the following.

```
C:\>triangle 4

WHAT YOU SEE BEFORE THE NEXT C:\>

C:\>
```

Answer:

(a) 2.

(b) argv[0] = "triangle", argv[1] = "4". All of them are characters.

(c)

```
C:\>triangle 4
*
**
***
****
***
**
*

C:\>
```

A line space is above the prompt.