EEET2601 Engineering Computing 1

Lab – Control Statements

Write a program for each of the following exercises.

1. Get a final result of a course as a float between 0 and 100 inclusively, then print the final grade for that course based on the below table. If the user enters a final result smaller 0 or greater than 100, print an error message and keep on asking the user to re-enter again until the final result is eventually between 0 and 100.

Final result	**Final grade**
result < 50	NN
50 <= result < 60	PA
60 <= result < 70	CR
70 <= result < 80	DI
result >= 80	HD

2. Get a sequence of positive integers from the user and compute their sum. The sequence ends when the user enters a negative number. The sum should not include the negative number at the end of the sequence. How many positive numbers are there in the sequence?

   Here is a sample run of the program.

```
Enter a positive integer: 8
Enter a positive integer: 5
Enter a positive integer: 22
Enter a positive integer: -3
Sum is 35
There are 3 positive integers
```

3. Display the characters in the ASCII Table from '!' to '~' nicely as below. There are 15 characters per line. The characters are separated by one space.

```
! " # $ % & ' ( ) * + , - . /
0 1 2 3 4 5 6 7 8 9 : ; < = >
? @ A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z [ \
] ^ _ ` a b c d e f g h i j k
l m n o p q r s t u v w x y z
{ | } ~
```

4. Print a multiplication table nicely as below. *Hint: use nested loop.*

```
  x |   1   2   3   4   5   6   7   8   9  10
 ---------------------------------------------
  1 |   1   2   3   4   5   6   7   8   9  10
  2 |   2   4   6   8  10  12  14  16  18  20
  3 |   3   6   9  12  15  18  21  24  27  30
  4 |   4   8  12  16  20  24  28  32  36  40
  5 |   5  10  15  20  25  30  35  40  45  50
  6 |   6  12  18  24  30  36  42  48  54  60
  7 |   7  14  21  28  35  42  49  56  63  70
  8 |   8  16  24  32  40  48  56  64  72  80
  9 |   9  18  27  36  45  54  63  72  81  90
 10 |  10  20  30  40  50  60  70  80  90 100
```

5. Prompt the user to enter an integer between 1 and 15 and displays a pyramid like the below example. Assume that the user always enters an integer between 1 and 15.

```
Enter the number of lines: 7 ↵ Enter
7 6 5 4 3 2 1 2 3 4 5 6 7
6 5 4 3 2 1 2 3 4 5 6
5 4 3 2 1 2 3 4 5
4 3 2 1 2 3 4
3 2 1 2 3
2 1 2
1
```

6. Prompt the user to enter an integer between 1 and 15 inclusively then display a pyramid accordingly as shown in the below sample. If the user enters an integer outside of 1 to 15, the program repeatedly asks the user to re-enter until the input falls in the range. Here is a sample run:

```
Enter an integer between 1 and 15: 0
Enter an integer between 1 and 15: 16
Enter an integer between 1 and 15: -5
Enter an integer between 1 and 15: 7
                    1
                  2 1 2
                3 2 1 2 3
              4 3 2 1 2 3 4
            5 4 3 2 1 2 3 4 5
          6 5 4 3 2 1 2 3 4 5 6
        7 6 5 4 3 2 1 2 3 4 5 6 7
```

7. Develop a Rock-Paper-Scissors game to play against the computer. Assume that 0 is used to represent Rock, 1 to represent Paper, and 2 to represent Scissors.

   First, the computer selects a random integer among 0, 1, and 2. This value is not known by the user. The user then enters a choice as an integer. The program then determines if the user wins, loses, or draws against the computer. When the user enters -1, the program exits. But if the user enters any value other than 0, 1, 2, and -1 the program prints an error messages then continues to play another round.

   Note that the rule is Rock wins Scissors, Scissors wins Paper, and Paper wins Rock.

   Here is a sample run of the game.

```
Welcome to the ROCK-PAPER-SCISSORS game
ROCK: 0, PAPER: 1, SCISSORS: 2, EXIT: -1
Enter an integer for your choice: 3
Invalid choice. You must enter one of the above integers.

Welcome to the ROCK-PAPER-SCISSORS game
ROCK: 0, PAPER: 1, SCISSORS: 2, EXIT: -1
Enter an integer for your choice: 0
You picked ROCK
Computer picked ROCK
It's a DRAW

Welcome to the ROCK-PAPER-SCISSORS game
ROCK: 0, PAPER: 1, SCISSORS: 2, EXIT: -1
Enter an integer for your choice: 1
You picked PAPPER
Computer picked ROCK
You WON

Welcome to the ROCK-PAPER-SCISSORS game
ROCK: 0, PAPER: 1, SCISSORS: 2, EXIT: -1
Enter an integer for your choice: 2
You picked SCISSOR
Computer picked ROCK
You LOST

Welcome to the ROCK-PAPER-SCISSORS game
ROCK: 0, PAPER: 1, SCISSORS: 2, EXIT: -1
Enter an integer for your choice: -2
Invalid choice. You must enter one of the above integers.

Welcome to the ROCK-PAPER-SCISSORS game
ROCK: 0, PAPER: 1, SCISSORS: 2, EXIT: -1
Enter an integer for your choice: -1
Program exits. Goodbye!!
```

   Hint: user the function rand() in the library <stdlib.h> to generate a random integer.