Tutorial 4 (Part B)

Part B – Repetition Control Structure

1. Identify the syntax/logic error(s) in each of the code snippets below and suggest the correction(s).

(a)
```
//this code should print out the numbers from 1 to 9
int x = 1;
while ( x > 10 );
{  cout <<setw(3)<< x;
   x ++;
}
```

(b)
```
//this code should print out the numbers from 1 to 100
int j = 1;
while (j <= 100)
cout << j << endl;
j++;
```

(c)
```
//This code is supposed to display "5! is 120."
int N = 5;
int factorial = 1;
while ( N >= 1 ) {
   factorial = factorial * N;
   N--;
   cout << "5! is " << factorial << ".\n";
}
```

2. Assume that Mr. Frodo is developing a game. Once the game ends, the user will be prompted to check whether they want to play again. Mr. Frodo wanted to ensure that regardless the user has entered an upper or lower-case "y" as the *response*, the game shall be repeated. Therefore he wrote the while loop as follows:

```
while (response == "y" && response == "Y")
{
      game(); //to execute the game module
}
```

The program can be compiled, but it doesn't seem to run properly. Explain what the error is.

3. For each of the nested structure below, suggest the appropriate type of **selection** statement and **loop** statement for each of the following program. The first question is done for you as example.

(a) You would like to print the numbers from 1 to 31 if the user enters "January", "March" or "May"; 1 to 28 if "February"; and 1 to 30 if it is "April" or "June".
Example of answer: Use *if-else* statement to select the month, use *for* loop to display the numbers.

(b) You would like to continuously generate 30 random numbers, and all the numbers must be unique.

(c) The user can order food through a system. At first, the user can select an option from the menu: 1 for Appetizer, 2 for Main Course, 3 for Dessert and 4 for Payment. Each time the user has selected food from one option, the menu will be prompted again for his other selection until he chooses option 4.

4. Given the code below, answer the following questions.

```
1    #include <iostream>
2    using namespace std;
3    int main()
4    {     int x=5, y=100;
5          do {
6                x = x + 10;}
7          while (x < y);
8          cout << x << " " << y << endl;
9
10         return 0;
11   }
```

(a) How many times is the code in Line 6 executed?
(b) What is the output of the program above?
(c) Rewrite the above program using a *for* loop.
(d) Explain the difference between a *while* loop and a *do-while* loop.

5. What does the following code print out?

(a)
```
for (int i=10; i > 5; i--) {
      if ( i %2 == 0)
            cout << 2*i << "a" << endl;
      else
            cout << i << "b" << endl;
}
```

(b)
```
int y = 0;
do {
      if (y == 6)
            y -= 10;
      if (y < 8)
            y += 2;
      cout << y << " ";
} while (y>0);
```

(c) What is the output of (b) if
   i. int y = 9 on Line 1
   ii. we changed the *do-while* loop into a *while* loop?

6. Write the output of the following segments of code.

```
for (int i=1; i<4; i++) {
  for (int j=1; j<=i; j++)
     cout << j << " ";
  cout << "\n";
}
```

7. A *for* statement as shown below is written in a program.

```
for( ; ; )
{
   ...
}
```

(a) Describe the implications behind the null expressions in the *for* statement above.

(b) Suggest how this statement can be exited.

(c) Is this a good structured programming style? Explain your answer.