BIL105E – Homework 2
Due Date : 08.04.2012 20:00
g++ compilable CPP codes and reports should be uploaded to Ninova.

In this homework you are asked to write a program that generates and displays "**lottery (lotto) coupons**" automatically with different functions for given user parameters. The program you will write generates lotto coupons randomly in different types according to the option selections of the user. So your program starts with asking the user what type of lotto coupon he/she wants and then program waits for different parameters according to the selection of user. After input parameters is entered by the user, program executes functions which are implemented for a specific task (type of the lotto coupon will be generated).
Lotto coupons can have different count of columns and each column must contain six (6) numbers will have been generated randomly between 1 & 49.

In this homework it's aimed to you improve your abilities on loops and functions. Your program will have six (6) base functions excluded main function.

- Four (4) of these functions are about generating lotto coupons and these are detailed below.

Function 1 & Details

```
void generateCoupon(int columnCount);
```

Function generates columns as the value of "***columnCount***", with six (6) numbers on each column and display them on screen.

```
Enter column count : 2
-------------------------------------------------------------
22     26     19     34     21     27
-------------------------------------------------------------
7      7      26     48     27     14
-------------------------------------------------------------
```

Function 2 & Details

```
void generateCouponWithConstant(int columnCount, int constantNumber, int
constantColumnCount);
```

Function generates columns as the value of "***columnCount***". And a constant number represented as "***constantNumber***" placed in a number of columns represented as "***constantColumnCount***".

```
Enter column count           : 3
Enter const. number          : 21
Enter const. column count    : 2
-------------------------------------------------------------
21     14     2      45     34     13
-------------------------------------------------------------
21     35     19     48     32     1
-------------------------------------------------------------
27     3      40     26     7      41
-------------------------------------------------------------
```

Function 3 & Details

```
void generateCouponWithRange(int columnCount, int lowerLimit, int
upperLimit, int numberCount);
```

Function generates columns as the value of "***columnCount***". And in the generated columns, numbers as "***numberCount***" is choosen from range defined as [***lowerLimit***, ***upperLimit***].

```
Enter column count           : 2
Enter lower limit of range   : 20
Enter upper limit of range   : 30
Enter number count in range  : 4
-------------------------------------------------------------
23     29     22     26     36     35
-------------------------------------------------------------
23     22     22     23     1      14
-------------------------------------------------------------
```

Function 4 & Details

```
void generateCouponWithDistance(int columnCount, int distance);
```

Function generates columns as the value of "***columnCount***". And distances between numbers in each column are as at least "***distance***".

```
Enter column count : 1
Enter min. distance between numbers : 6
-------------------------------------------------------------
1      8      15     24     33     40
-------------------------------------------------------------

   7      7       9     9      7
```

Note that because of numbers will be generates between 1-49 maximum distance between numbers **can have value of 9**.

- Two (2) of functions you will write are about generating random number and these are detailed below.

Function 5 & Details

```
int myRand();
```

Function generates random numbers for an algorithm described following lines uses time(0) seed. You can use "***srand(time(0))***" and "***rand()***" functions placed in "***ctime***" and "***cstdlib***" libraries respectively.

E.g. it is started with seed 54321.
54321^2 = 2950771041 → getRange (2950771041, **3**, **7**) // returns 50771
50771^2 = 2577694441 → getRange (2577694441, **3**, **7**) // returns 77694
…………
…………

Note that these steps will repeat as a predefined number of iteration which is fixed with ***"#define***" keyword and at the end of this process a random number will have been generated by your function. Also you can manipulate generated random seeds without manipulating the skeleton of the given algorithm in your function in order to prevent integer overflows.

Function 6 & Details

```
int getRange(int number, int startIndex, int endIndex);
```

Function returns a sub number which starts with "***startIndex***" th. digit and ends "***endIndex***" th. digit in the given "***number***".

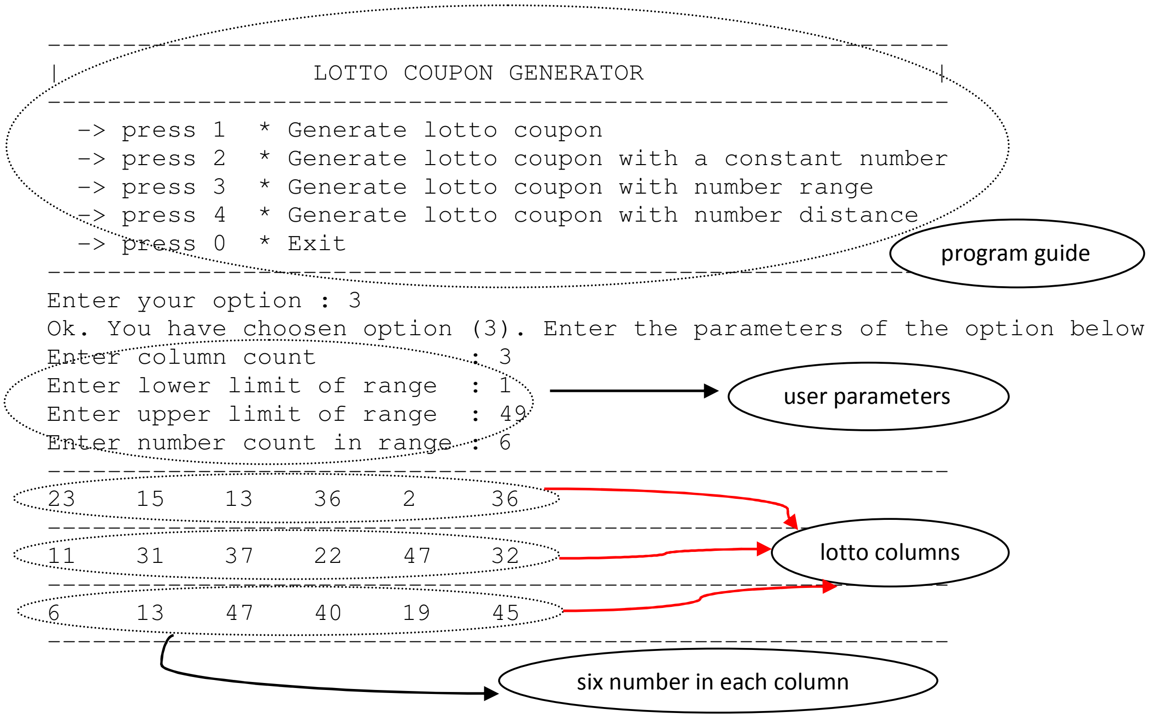
Note that start index of function begins with one (1) and do not forget to control the state of end index is exceeding total digits of the given number.

Important points & cues you should consider while coding your homework.

- Random numbers in lotto columns do not have to be ordered.
- Same numbers generated via your random function can be placed in columns. Also programs considering and implementing the situation of "all numbers are different from each other in a column" **will be graded +15 bonus points**.

- Please check input parameters given by user and put control statements inside of your code not to get unexpected states. Also display meaningful messages for unexpected states.
- Please check random seeds not to overflow range of data types.
- Please insert comment lines among your codes.
- Please pay attention to give meaningful names for variables you used.

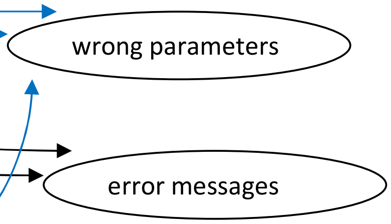
Example sessions -1



```
------------------------------------------------------------
|                 LOTTO COUPON GENERATOR                   |
------------------------------------------------------------
  -> press 1  * Generate lotto coupon
  -> press 2  * Generate lotto coupon with a constant number
  -> press 3  * Generate lotto coupon with number range
  -> press 4  * Generate lotto coupon with number distance
  -> press 0  * Exit
------------------------------------------------------------
Enter your option : 3
Ok. You have choosen option (3). Enter the parameters of the option below!
Enter column count           : 3
Enter lower limit of range   : 1
Enter upper limit of range   : 49
Enter number count in range  : 6
------------------------------------------------------------
23     15     13     36     2      36
------------------------------------------------------------
11     31     37     22     47     32
------------------------------------------------------------
6      13     47     40     19     45
------------------------------------------------------------
```

Example session 2 (Unexpected state – undefined option)

```
------------------------------------------------------------
|                  LOTTO COUPON GENERATOR                  |
------------------------------------------------------------
  -> press 1  * Generate lotto coupon
  -> press 2  * Generate lotto coupon with a constant number
  -> press 3  * Generate lotto coupon with number range
  -> press 4  * Generate lotto coupon with number distance
  -> press 0  * Exit
------------------------------------------------------------
Enter your option : d
Undefined option !
Enter your option : e
Undefined option !
Enter your option :
```



Example session 3 (Unexpected state – wrong parameters)

```
Enter your option : 3
Ok. You have choosen option (3). Enter the parameters of the option below!
Enter column count          : 1
Enter lower limit of range  : 1
Enter upper limit of range  : 10
Enter number count in range : 7
Count of numbers which will be choosen from range, cannot be larger than 6
(six)!
```

Skeleton code that you might useful to follow

```cpp
#include <iostream>
#include <ctime>
#include <cstdlib>

using namespace std;
int main() {
	char option;
	bool exitLotto = false;

	srand(time(0));   // random seed with time(0)
	while (!exitLotto) {    // Loop if 'exit' option not selected
		cout << "Enter your option : "; cin >> option;

		switch (option) {
		case '0': {
			exitLotto = true;
			break;
		}
		case '1': {
			/* Declare necessary variables and get parameters from
			*  user and call function.
			break;
		}
		case '2': {
			/* Declare necessary variables and get parameters from
			*  user and call function.
			break;
		}
```

```
            case '3': {
                  /* Declare necessary variables and get parameters from
                  *  user and call function.
                  break;
            }
            case '4': {
                  /* Declare necessary variables and get parameters from
                  *  user and call function.
                  break;
            }
            default:
            }
      }

      return 0;
}
```

Note : Please be aware that any attempt on academic dishonesty and cheating will be subject to offical penalties.