

Final assignment: Report of Lottery machine

Make a command line application that draws lottery numbers. 7 numbers and 3 bonus numbers. The numbers are between 1 and 39. At the end, the application will sort the numbers from biggest to smallest. First the regular numbers and then the bonus numbers. You can also include lottery tickets from few users and define prize money for different results. For example, 5 numbers correct, 5+1 correct, 6 correct, 6+1 correct and all seven correct. You can also add a feature that draws random lottery numbers for the user. After the actual lottery draw the application tells how many numbers were correct and how much money the user won. Text based user interface is defined by the student.

Program Code:

```
#include<iostream>
```

```
#include<vector>
```

```
using namespace std;
```

```
void getUserNumbers(int[], const int);
```

```
void findDuplicateNumbers(vector<int>&);
```

```
void getMatchingDigits(vector<int>,
```

```
    vector<int>&,
```

```
    const int[],
```

```
    int&,
```

```
    const int);
```

```
void displayInfo(const int[],
```

```
    const int[],
```

```
    const vector<int>,
```

```
    const int,
```

```
    const int);
```

```
int main()
{
    const int ARRAY_SIZE = 5;
    int total_matching_digits = 0;
    int lottery_numbers[ARRAY_SIZE] = { 7, 6, 5, 4, 3 };
    int user_numbers[ARRAY_SIZE];
    vector<int> duplicate_numbers;
    vector<int> matching_digits;

    getUserNumbers(user_numbers, ARRAY_SIZE);

    for (int i = 0; i < ARRAY_SIZE; i++)
        duplicate_numbers.push_back(user_numbers[i]);

    findDuplicateNumbers(duplicate_numbers);

    getMatchingDigits(duplicate_numbers,
        matching_digits,
        lottery_numbers,
        total_matching_digits,
        ARRAY_SIZE);

    displayInfo(user_numbers,
        lottery_numbers,
        matching_digits,
        total_matching_digits,
        ARRAY_SIZE);

    return 0;
}
```

```
} // End int main()
```

```
void getUserNumbers(int user_numbers[], const int ARRAY_SIZE)
```

```
{  
    cout << "Enter 5 digits: " << endl;  
    for (int i = 0; i < ARRAY_SIZE; i++)  
    {  
        cout << "Number #" << (i + 1) << ": ";  
        while (!(cin >> user_numbers[i]) ||  
                (user_numbers[i] < 0 || user_numbers[i] > 9))  
        {  
            cout << "Error: enter a number between 0 and 9: ";  
            cin.clear();  
            cin.ignore(numeric_limits<streamsize>::max(), '\n');  
        }  
    }  
}
```

```
void findDuplicateNumbers(vector<int>& duplicate_numbers)
```

```
{  
    int size_of_array = duplicate_numbers.size();  
  
    for (int i = 0; i < size_of_array; i++)  
    {  
        for (int j = (i + 1); j < size_of_array;)   
        {  
            if (duplicate_numbers[i] == duplicate_numbers[j])  
            {  
                cout << "Duplicate number found: " << duplicate_numbers[i] << endl;  
            }  
        }  
    }  
}
```

```

        for (int k = j; k < (size_of_array - 1); k++)
            duplicate_numbers[k] = duplicate_numbers[k + 1];

        duplicate_numbers.pop_back();
        size_of_array--;
    }
    else
        j++;
}
}
}

```

```

void getMatchingDigits(vector<int> duplicate_numbers,
    vector<int>& matching_digits,
    const int lottery_numbers[],
    int& total_matching_digits,
    const int ARRAY_SIZE)

{
    for (int i = 0; i < duplicate_numbers.size(); i++)
    {
        for (int j = 0; j < (ARRAY_SIZE); j++)
        {
            if (duplicate_numbers[i] == lottery_numbers[j])
            {

                total_matching_digits++;
                matching_digits.push_back(duplicate_numbers[i]);
            }
        }
    }
}

```

```

        }
    }
}

```

```

void displayInfo(const int user_numbers[],
    const int lottery_numbers[],
    const vector<int>matching_digits,
    const int total_matching_digits,
    const int ARRAY_SIZE)

{
    cout << "\nUser numbers:  { ";
    for (int i = 0; i < ARRAY_SIZE; ++i)
        cout << user_numbers[i] << " ";
    cout << "}" << endl;

    cout << "Lottery numbers: { ";
    for (int i = 0; i < ARRAY_SIZE; i++)
        cout << lottery_numbers[i] << " ";
    cout << "}" << endl;

    cout << "Matching digits: { ";
    for (int i = 0; i < matching_digits.size(); ++i)
        cout << matching_digits[i] << " ";
    cout << "}" << endl;

    cout << "\nTotal matching #s: "
        << total_matching_digits

```

```
<< endl;

if (total_matching_digits == ARRAY_SIZE)

    cout << "You are the grand prize winner!!" << endl;

}
```

Output:

```
Microsoft Visual Studio Debug Console
Enter 5 digits:
Number #1: 4 7 8 3 2
Number #2: Number #3: Number #4: Number #5:
User numbers: { 4 7 8 3 2 }
Lottery numbers: { 7 6 5 4 3 }
Matching digits: { 4 }

Total matching #s: 1

C:\Users\USER\source\repos\Lottery machine\Debug\Lottery machine.exe (process 8248) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```