2-Digit/3-Digit Lottery Game

Hissamuddin Shaikh

CSCI 201

2nd June 2018

St. Cloud State University

**2-Digit/3-Digit Lottery Game**

**Problem Statement**:

Create a lottery game that randomly generates a 2-Digit number and a 3-Digit number, and then asks the user to predict these 2 numbers. If the person predicts correctly (or close enough), they are rewarded with a certain amount of money.

**Input/Output Description:**

Input: Their 2-Digit Prediction for the 2-Digit Lottery, as well as a “Yes” or “No” to whether they want to participate in the 3-Digit Lottery, and if they answer “Yes”, then their 3-Digit Prediction.

Output: The actual 2-Digit and 3-Digit lottery number, and how much money they have won, based on their prediction.

**Variables:**

* *rand ()*
  + First step in setting up the random number that will be the winning lottery number for the 2-Digit and the 3-Digit Lotterys.
* *lottery\_actual*
  + Stores the randomly generated lottery number, for both the 2-Digit and 3-Digit lottery.
  + Type: int
* *remaining\_lottery\_actual*
  + Stores information related to breaking down the randomly generated 3-digit lottery number, so that individual digits can be identified and compared later in the program.
  + Type: int
* *lottery\_1st\_digit\_actual*
  + Stores the 1st digit of the randomly generated lottery number for both, the 2-Digit Lottery and the 3-Digit Lottery.
  + Type: int
* *lottery\_2nd\_digit\_actual*
  + Stores the 2nd digit of the randomly generated lottery number for both, the 2-Digit Lottery and the 3-Digit Lottery.
  + Type: int
* *lottery\_3rd\_digit\_actual*
  + Stores the 3rd digit of the randomly generated lottery number for both, the 2-Digit Lottery and the 3-Digit Lottery.
  + Type: int
* *lottery\_prediction*
  + Stores the predicted lottery number inserted by the user, for both the 2-Digit and 3-Digit lottery.
  + Type: int
* *remaining\_lottery\_prediction*
  + Stores information related to breaking down the users predicted 3-digit lottery number, so that individual digits can be identified and compared later in the program.
  + Type: int
* *lottery\_1st\_digit\_prediction*
  + Stores the 1st digit of the user inserted lottery number for both, the 2-Digit Lottery and the 3-Digit Lottery.
  + Type: int
* *lottery\_2nd\_digit\_prediction*
  + Stores the 2nd digit of the user inserted lottery number for both, the 2-Digit Lottery and the 3-Digit Lottery.
  + Type: int
* *lottery\_3rd\_digit\_prediction*
  + Stores the 3rd digit of the user inserted lottery number for both, the 2-Digit Lottery and the 3-Digit Lottery.
  + Type: int
* *response\_3\_digit*
  + Stores the “Yes” or “No” response to the user wanting to participate in the 3-Digit Lottery.
  + Type: string

**Program Design:**

1. Prompt the user to insert their 2-Digit Lottery prediction.
2. Generate a random 2-Digit Lottery number.
3. Check if the random number is the same, as the one inserted by the user. If so, then display a message rewarding them with $10,000.
4. If not, check if the number they predicted contains the same digits as the randomly generated one. If so, display a message rewarding them with $3000.
5. If not, check if the number they predicted contains at least one same digit as the randomly generated one. If so, display a message rewarding them with $1000.
6. Ask the user if they wish to participate in the 3-Digit Lottery. If they answer “Yes”, repeat the process above, except with a randomly generated 3-Digit number, instead of a 2-Digit number. If “No”, then terminate the program.

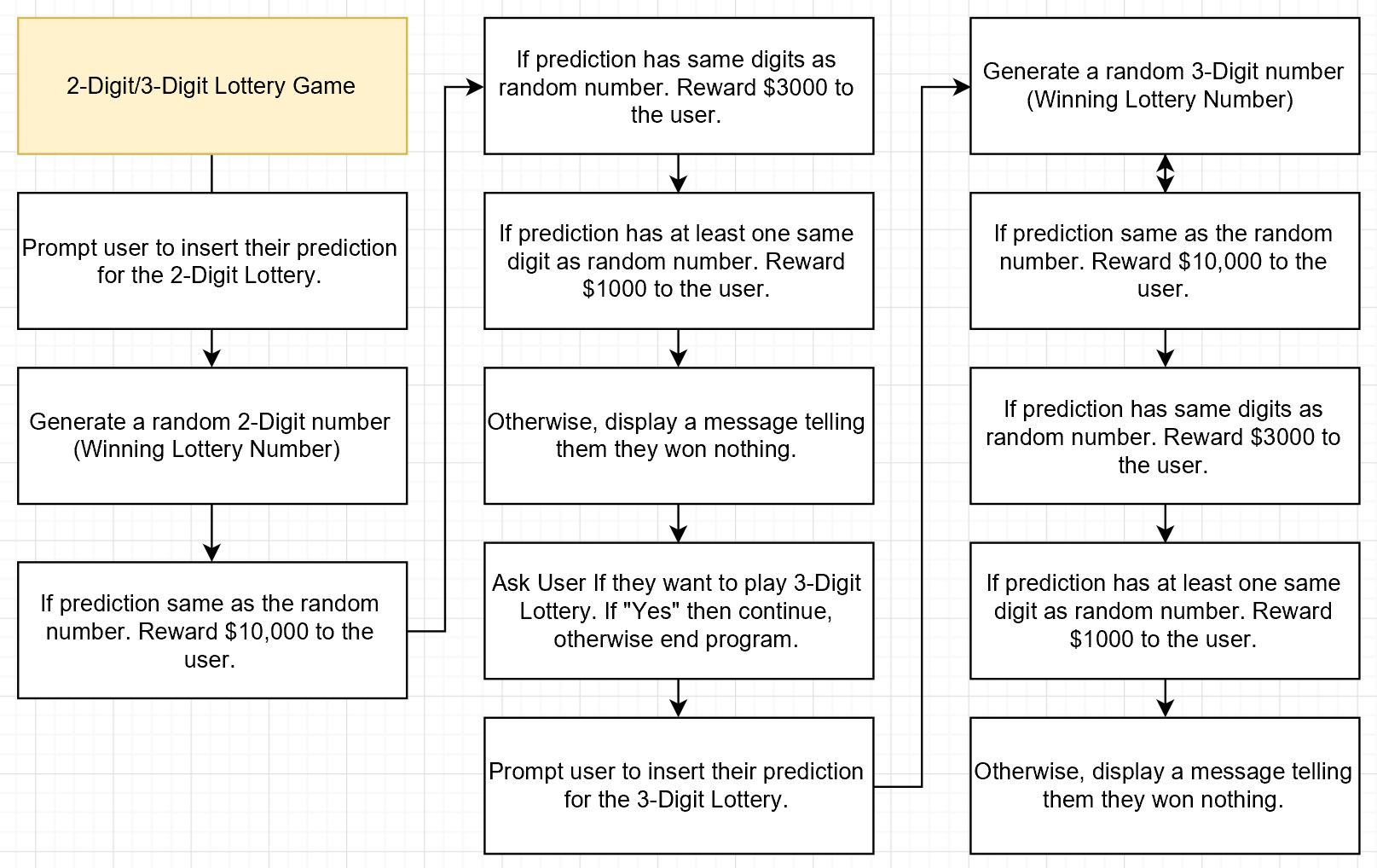
**Flowchart:**

Figure. A

Note: “If prediction same as the random number. Reward $10,000 to the user.”, “If prediction has same digits as random number. Reward $3000 to the user.”, and “If prediction has at least one same digit as random number. Reward $1000 to the user.” are further explained below in Figure. B, Figure. C, and Figure. D.

A screenshot of a cell phone

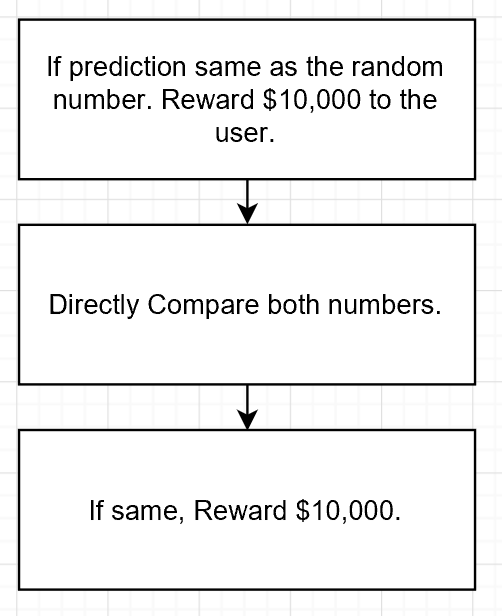
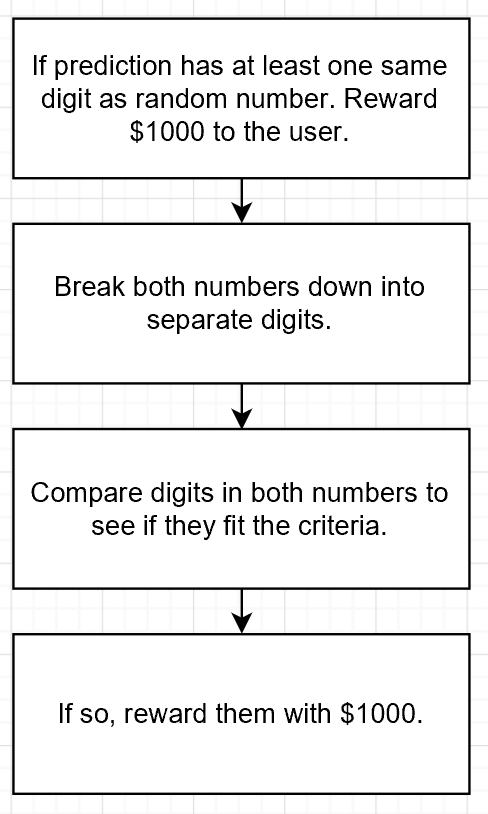
Description generated with very high confidenceFigure. B

Figure. C

Figure. D

**Algorithm Development (Detailed Pseudocode):**

* Prompt the user to insert a 2-Digit number that they think is going to be the winning lottery number through the statement, “Please insert a 2 digit number, that you think will be the lucky lottery number:”. Store their inserted number in *lottery\_prediction*.
* Check if the number they have inserted is indeed 2 digits, through a while loop. If not, ask them to insert it again. Continue this until a 2-digit number is inserted.
* Generate a 2-digit random number, and store this in *lottery\_actual*.
* Check if the number generated is indeed 2 digits, through a while loop. If not, keep repeating the generation process until a 2-digit number is produced.
* Break the random number down through the following:
  + *lottery\_1st\_digit\_actual* = *lottery\_actual*/10
  + *lottery\_2nd\_digit\_actual* = *lottery\_actual*%10
* Break the user inserted number down through the following:
  + *lottery\_1st\_digit\_prediction* = *lottery\_prediction*/10
  + *lottery\_2nd\_digit\_prediction* = *lottery\_prediction*%10
* Determine rewarding amount through the following:
  + If *lottery\_actual* = *lottery\_prediction,* reward $10,000.
  + Else If *lottery\_1st\_digit\_actual* = *lottery\_2nd\_digit\_prediction* AND *lottery\_2nd\_digit\_actual* = *lottery\_1st\_digit\_prediction,* reward$3000*.*
  + Else if *lottery\_1st\_digit\_actual* = *lottery\_2nd\_digit\_prediction* OR *lottery\_2nd\_digit\_actual* = *lottery\_1st\_digit\_prediction* OR *lottery\_1st\_digit\_actual* = *lottery\_1st\_digit\_prediction* OR *lottery\_2nd\_digit\_actual* = *lottery\_2nd\_digit\_prediction,* reward$1000*.*
  + Else display, “Sorry, your prediction didn’t win you anything :(”.
* Prompt user to insert “Yes” or “No”, regarding their desire to play the 3-Digit Lottery, through the statement, “Would you like to play another lottery, with a 3 digit number? (Respond Yes or No):”. Store their response in *response\_3\_digit*.
* If the response is “Yes”, continue onwards. Otherwise, display “Alright, Goodbye then.”, and terminate the program.
* Prompt the user to insert a 3-Digit number that they think is going to be the winning lottery number through the statement, “Please insert a 3 digit number, that you think will be the lucky lottery number:”. Store their inserted number in *lottery\_prediction* (Overwrite).
* Check if the number they have inserted is indeed 3 digits, through a while loop. If not, ask them to insert it again. Continue this until a 3-digit number is inserted.
* Generate a 3-digit random number, and store this in *lottery\_actual* (Overwrite).
* Check if the number generated is indeed 3 digits, through a while loop. If not, keep repeating the generation process until a 3-digit number is produced.
* Break the random number down through the following:
  + *lottery\_1st\_digit\_actual* = *lottery\_actual*/100(Overwrite)
  + *remaining\_lottery\_actual* = *lottery\_actual*%100
  + *lottery\_2nd\_digit\_actual* = *remaining\_lottery\_actual*/10(Overwrite)
  + *lottery\_3rd\_digit\_actual* = *remaining\_lottery\_actual*%10
* Break the user inserted number down through the following:
  + *lottery\_1st\_digit\_prediction* = *lottery\_prediction*/100(Overwrite)
  + *remaining\_lottery\_prediction* = *lottery\_prediction*%100
  + *lottery\_2nd\_digit\_prediction* = *remaining\_lottery\_prediction*/10(Overwrite)
  + *lottery\_3rd\_digit\_prediction* = *remaining\_lottery\_prediction*%10
* Determine rewarding amount through the following:
  + If *lottery\_actual* = *lottery\_prediction,* reward $10,000.
  + Else If ( *lottery\_1st\_digit\_actual* = *lottery\_1st\_digit\_prediction* OR *lottery\_1st\_digit\_actual* = *lottery\_2nd\_digit\_prediction* OR *lottery\_1st\_digit\_actual* = *lottery\_3rd\_digit\_prediction)* AND (*lottery\_2nd\_digit\_actual* = *lottery\_1st\_digit\_prediction* OR *lottery\_2nd\_digit\_actual* = *lottery\_2nd\_digit\_prediction* OR *lottery\_2nd\_digit\_actual* = *lottery\_3rd\_digit\_prediction)* AND *(lottery\_3rd\_digit\_actual* = *lottery\_1st\_digit\_prediction OR lottery\_3rd\_digit\_actual* = *lottery\_2nd\_digit\_prediction OR lottery\_3rd\_digit\_actual* = *lottery\_3rd\_digit\_prediction)*, reward $3000.
  + Else if ( *lottery\_1st\_digit\_actual* = *lottery\_1st\_digit\_prediction* OR *lottery\_1st\_digit\_actual* = *lottery\_2nd\_digit\_prediction* OR *lottery\_1st\_digit\_actual* = *lottery\_3rd\_digit\_prediction)* OR (*lottery\_2nd\_digit\_actual* = *lottery\_1st\_digit\_prediction* OR *lottery\_2nd\_digit\_actual* = *lottery\_2nd\_digit\_prediction* OR *lottery\_2nd\_digit\_actual* = *lottery\_3rd\_digit\_prediction)* OR *(lottery\_3rd\_digit\_actual* = *lottery\_1st\_digit\_prediction OR lottery\_3rd\_digit\_actual* = *lottery\_2nd\_digit\_prediction OR lottery\_3rd\_digit\_actual* = *lottery\_3rd\_digit\_prediction)*, reward $1000.
  + Else display, “Sorry, your prediction didn’t win you anything :(”.

**Program Listing:**

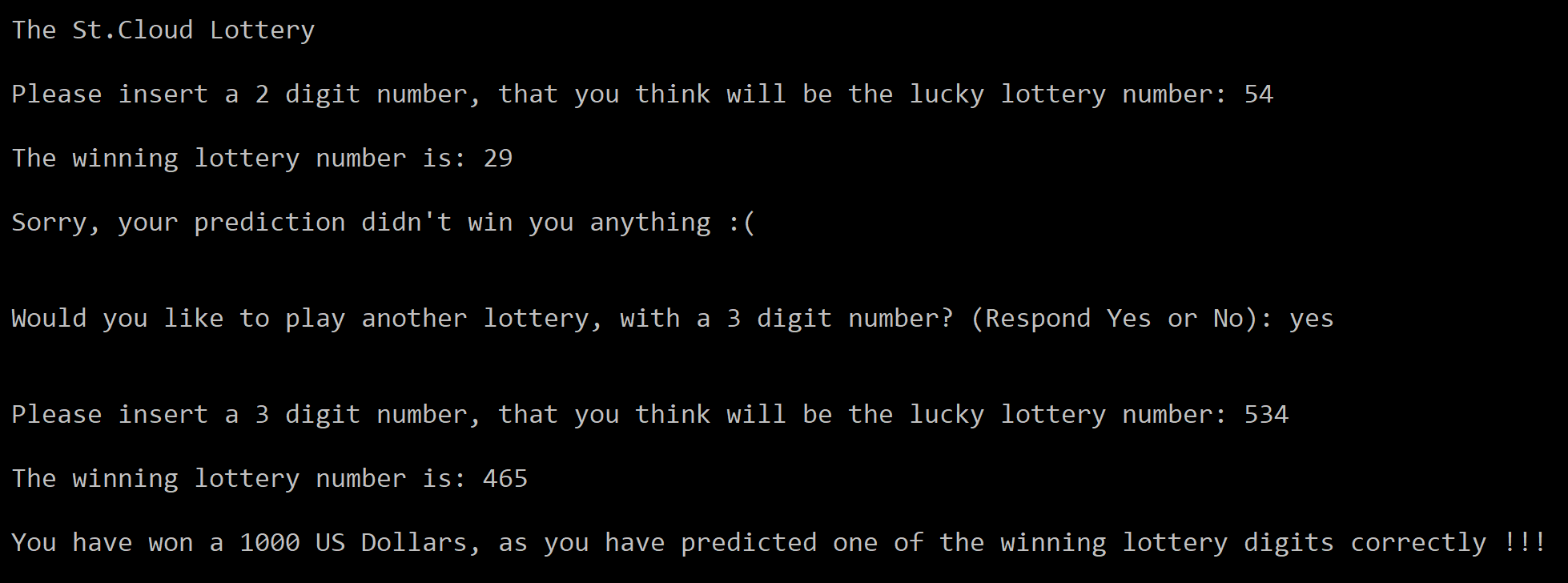
1. /\*2-Digit/3-Digit Lottery Game by Hissamuddin Shaikh\*/
2. #include <iostream>
3. #include <cmath>
4. #include <cstdlib>
5. #include <ctime>
6. #include <string>
7. using namespace std;
8. int main()
9. {
10. /\*Variable Setup\*/
11. int rand();
12. int lottery\_actual;
13. int remaining\_lottery\_actual;
14. int lottery\_1st\_digit\_actual;
15. int lottery\_2nd\_digit\_actual;
16. int lottery\_3rd\_digit\_actual;
17. int lottery\_prediction;
18. int remaining\_lottery\_prediction;
19. int lottery\_1st\_digit\_prediction;
20. int lottery\_2nd\_digit\_prediction;
21. int lottery\_3rd\_digit\_prediction;
22. string response\_3\_digit;
23. /\*Prompt the user to enter their 2 digit prediction, for the winning lottery number\*/
24. cout << endl << " The St.Cloud Lottery"<< endl;
25. cout << endl << " Please insert a 2 digit number, that you think will be the lucky lottery number: ";
26. cin >> lottery\_prediction;
27. /\*Error control for user's predicted lottery number\*/
28. while (lottery\_prediction>100)
29. {cout << endl << " Please insert a '2 digit number', that you think will be the lucky lottery number: ";
30. cin >> lottery\_prediction;}
31. /\*Randomly generate the 2 digit winning lottery number\*/
32. srand(time(NULL));
33. lottery\_actual = rand()%100;
34. /\*Error control for the generation of the winning lottery number\*/
35. while (lottery\_actual<10)
36. lottery\_actual = rand()%100;
37. cout << endl << " The winning lottery number is: " <<lottery\_actual <<endl;
38. /\*Setup for 1st digit and 2nd digit extraction of the actual lottery number\*/
39. lottery\_1st\_digit\_actual = lottery\_actual/10;
40. lottery\_2nd\_digit\_actual = lottery\_actual%10;
41. /\*Setup for 1st digit and 2nd digit extraction of the predicted lottery number\*/
42. lottery\_1st\_digit\_prediction = lottery\_prediction/10;
43. lottery\_2nd\_digit\_prediction = lottery\_prediction%10;
44. /\*Test if the user's lottery number is the same as the winning lottery number, or if the digits are the
45. or if one of the digits is the same\*/
46. if (lottery\_actual==lottery\_prediction)
47. cout << endl <<" You have won 10,000 US Dollars, as you have predicted the winning lottery number correctly !!!" <<endl <<endl;
48. else if (lottery\_1st\_digit\_actual==lottery\_2nd\_digit\_prediction && lottery\_2nd\_digit\_actual==lottery\_1st\_digit\_prediction)
49. cout << endl <<" You have won 3000 US Dollars, as you have predicted the winning lottery digits correctly !!!" <<endl << endl;
50. else if (lottery\_1st\_digit\_actual==lottery\_2nd\_digit\_prediction || lottery\_2nd\_digit\_actual==lottery\_1st\_digit\_prediction ||
51. lottery\_1st\_digit\_actual==lottery\_1st\_digit\_prediction || lottery\_2nd\_digit\_actual == lottery\_2nd\_digit\_prediction)
52. cout << endl <<" You have won a 1000 US Dollars, as you have predicted one of the winning lottery digits correctly !!!"<<endl << endl;
53. else
54. cout << endl << " Sorry, your prediction didn't win you anything :(" << endl <<endl;
55. /\*Set up for a 3 digit lottery game\*/
56. cout << endl << " Would you like to play another lottery, with a 3 digit number? (Respond Yes or No): ";
57. cin >> response\_3\_digit; cout<< endl;
58. if (response\_3\_digit == "Yes" || response\_3\_digit == "YES" || response\_3\_digit == "yes" || response\_3\_digit == "yEs" ||
59. response\_3\_digit == "yeS" || response\_3\_digit == "YEs" || response\_3\_digit == "yES")
60. /\*Setup for 3 number lottery prediction\*/
61. {cout << endl << " Please insert a 3 digit number, that you think will be the lucky lottery number: ";
62. cin >> lottery\_prediction;
63. /\*Error control for 3 digit prediction\*/
64. while (lottery\_prediction>1000 || lottery\_prediction<100)
65. {cout << endl << " Please insert a '3 digit number', that you think will be the lucky lottery number: ";
66. cin >> lottery\_prediction;}
67. /\*Setup for the generation of the winning 3 digit lottery number\*/
68. srand(time(NULL));
69. lottery\_actual = rand()%1000;
70. /\*Error control for the 3 digit actual lottery number\*/
71. while (lottery\_actual<100 || lottery\_actual>1000)
72. lottery\_actual = rand()%100;
73. cout << endl << " The winning lottery number is: " <<lottery\_actual <<endl;
74. /\*Setup for 1st digit, 2nd Digit and 3rd Digit Extraction for actual lottery number\*/
75. lottery\_1st\_digit\_actual = lottery\_actual/100;
76. remaining\_lottery\_actual = lottery\_actual%100;
77. lottery\_2nd\_digit\_actual = remaining\_lottery\_actual/10;
78. lottery\_3rd\_digit\_actual = remaining\_lottery\_actual%10;
79. /\*Setup for 1st digit, 2nd digit and 3rd digit extraction for predicted lottery number\*/
80. lottery\_1st\_digit\_prediction = lottery\_prediction/100;
81. remaining\_lottery\_prediction = lottery\_prediction%100;
82. lottery\_2nd\_digit\_prediction = remaining\_lottery\_prediction/10;
83. lottery\_3rd\_digit\_prediction = remaining\_lottery\_prediction%10;
84. /\*Test if the user's lottery number is the same as the winning lottery number, or if the digits are the
85. or if one of the digits is the same\*/
86. if (lottery\_actual==lottery\_prediction)
87. cout << endl <<" You have won 10,000 US Dollars, as you have predicted the winning lottery number correctly !!!" <<endl;
88. else if ((lottery\_1st\_digit\_actual==lottery\_1st\_digit\_prediction || lottery\_1st\_digit\_actual==lottery\_2nd\_digit\_prediction || lottery\_1st\_digit\_actual==lottery\_3rd\_digit\_prediction)
89. &&(lottery\_2nd\_digit\_actual==lottery\_1st\_digit\_prediction || lottery\_2nd\_digit\_actual==lottery\_2nd\_digit\_prediction || lottery\_2nd\_digit\_actual==lottery\_3rd\_digit\_prediction)
90. &&(lottery\_3rd\_digit\_actual==lottery\_1st\_digit\_prediction|| lottery\_3rd\_digit\_actual==lottery\_2nd\_digit\_prediction ||lottery\_3rd\_digit\_actual==lottery\_3rd\_digit\_prediction))
91. cout << endl <<" You have won 3000 US Dollars, as you have predicted the winning lottery digits correctly !!!" <<endl;
92. else if ((lottery\_1st\_digit\_actual==lottery\_1st\_digit\_prediction || lottery\_1st\_digit\_actual==lottery\_2nd\_digit\_prediction || lottery\_1st\_digit\_actual==lottery\_3rd\_digit\_prediction)
93. ||(lottery\_2nd\_digit\_actual==lottery\_1st\_digit\_prediction || lottery\_2nd\_digit\_actual==lottery\_2nd\_digit\_prediction || lottery\_2nd\_digit\_actual==lottery\_3rd\_digit\_prediction)
94. ||(lottery\_3rd\_digit\_actual==lottery\_1st\_digit\_prediction|| lottery\_3rd\_digit\_actual==lottery\_2nd\_digit\_prediction ||lottery\_3rd\_digit\_actual==lottery\_3rd\_digit\_prediction))
95. cout << endl <<" You have won a 1000 US Dollars, as you have predicted one of the winning lottery digits correctly !!!" <<endl;
96. else
97. cout << endl << " Sorry, your prediction didn't win you anything again :(" << endl;}
98. else
99. cout << endl << " Alright, Goodbye then." << endl;
100. return 0;
101. }

**Note:**

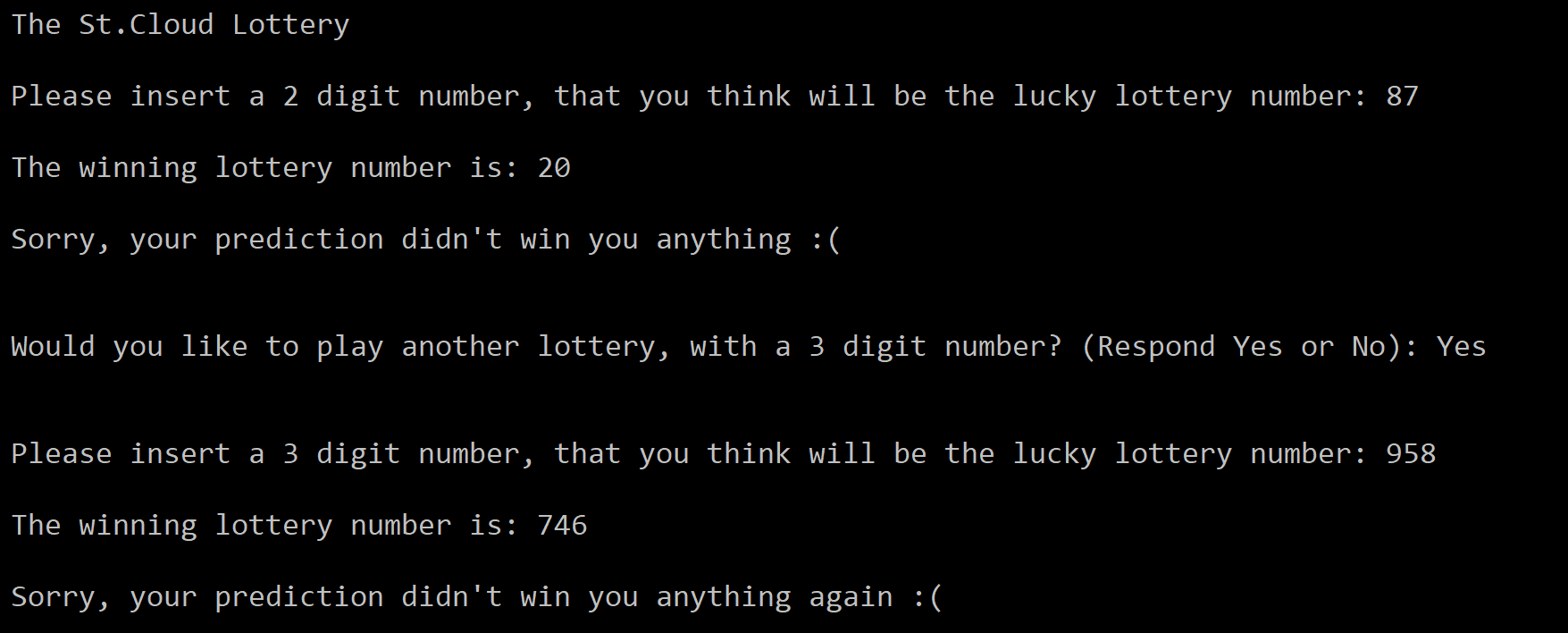
Screenshots of the original code are attached at the end of the document (in the notes section), incase this is unclear.

**Sample test run of the program:**

1st Sample:



2nd Sample:



3rd Sample:

A screenshot of a computer screen

Description generated with very high confidence

**Observations, error handling and general comments:**

The program runs correctly as the result in the 1st Sample, 2nd Sample and 3rd Sample correspond to manual computations.

**1st Sample (Manual Computation) =**

User Prediction for 2-Digit Lottery: 54

Random Generated Number for 2-Digit Lottery: 29

Winnings: None

Desire of User to Play 3-Digit Lottery: yes

User Prediction for 3-Digit Lottery: 534

Random Generated Number for 3-Digit Lottery: 465

Winnings: $1000, as 2 numbers match (if 3 match, only then you get $3000).

This is the same as the one computed by the computer.

**2nd Sample (Manual Computation) =**

User Prediction for 2-Digit Lottery: 87

Random Generated Number for 2-Digit Lottery: 20

Winnings: None

Desire of User to Play 3-Digit Lottery: Yes

User Prediction for 3-Digit Lottery: 958

Random Generated Number for 3-Digit Lottery: 746

Winnings: None

This is the same as the one computed by the computer.

**3rd Sample (Manual Computation) =**

User Prediction for 2-Digit Lottery: 69

Random Generated Number for 2-Digit Lottery: 34

Winnings: None

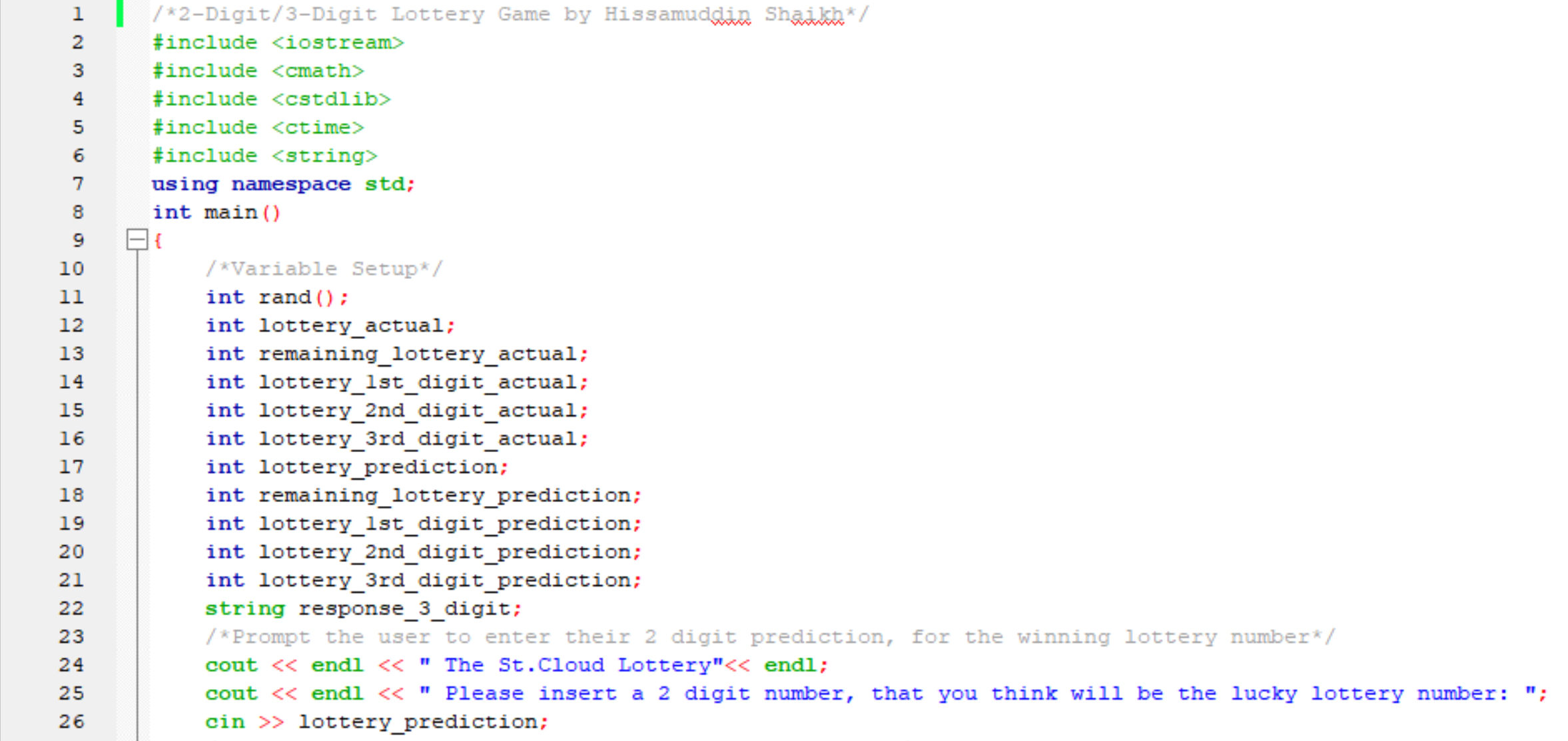
Desire of User to Play 3-Digit Lottery: no

This is the same as the one computed by the computer.

No errors should be encountered at all as various measures have been implemented into the program to prevent predictable issues. Loops have been setup to detect if a number other than 2 digits or 3 digits has been entered or generated and carry out the appropriate action to fix the issue. Moreover, to minimize errors related to the user’s desire to participate in the 3-Digit Lottery, various forms of “Yes” have been made to register as the desired command, to prevent misinterpretation on the programs behalf (anything else entered will be registered as a “No” and lead to the termination of the program).

**Conclusions**

The 2-Digit/3-Digit Lottery Game that has been constructed in this report should serve sufficiently and efficiently, with regards to rewarding the person for their prediction of a 2-Digit and 3-Digit number. There is minimal room for error, and confusion for the user.

**Notes:**

