

SLOT MACHINE

CSC-5 40375



February 4, 2014

ABEL SALAZAR

Asalazr25@student.rcc.edu

INDEX

PREFACE page2

INFO page2

VARIABLES page3

FUNCTIONS page3

FLOWCHART page6

Copyright - February 4, 2014

**Preface**

The first slot machine was made in 1895 by a mechanic by the name of Charles Fey and it was called the Liberty Bell. They originally operated with a lever and was nicknamed the One Armed Bandit because it would have you spending all of your money. Now a days the lever has changed to a button and screen but you are left equally broke. Newer model slot machines take into account the amount of spins and even the time of day they are being used to create a complex algorithm.

**INFO**

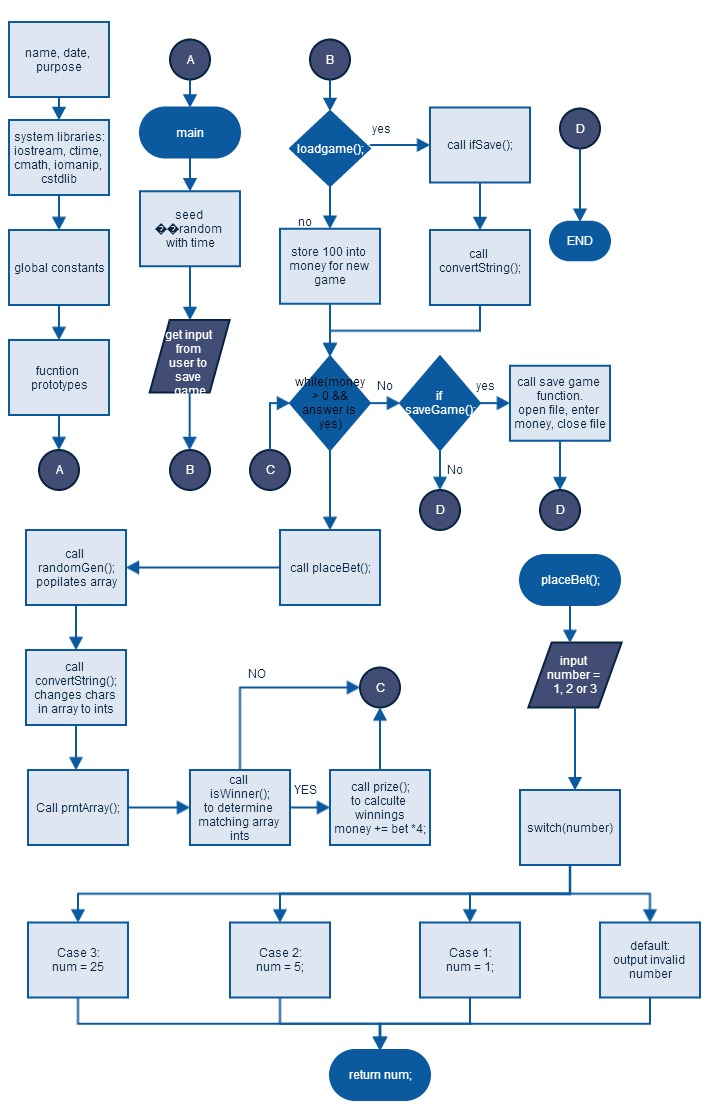
This program is my rendition on the slot machine game made over a century ago. Although it has been simplified to only include horizontal and vertical matches versus having the diagonals included. The game allows you to save your game for a later time. It saves the value of your saved game on a txt file “saved\_game.txt”. The file is included in the program file and has a default value of 100 dollars to start off. There are 3 predetermined values for bets you can make to take some of the user error out of the equation. Wins return the amount bet times 4. The program will terminate when the players balance reaches 0. Currently the game is optimized for Netbeans but cstdlib has been added for some older machines. Bloodshed will need a separate return at the end of main or it will not run.

Variables

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Function** | **Description** | **Location** |
| Int | Whole numbers | 4 bytes | Every function |
| Unsigned short | Whole numbers | 2 bytes | Used with random number generator  int main(); |
| Long | Whole numbers | 8 bytes | Global variable “money” |
| Char | Character | 1 byte | int convertString(string);  int convertNum(char);  void cvn\_char (int [], int, int); |
| String | Character Array | Depends on the size of the array | int convertString(string);  string loadGame(); |
| Bool | True / false |  | bool ifSave(); |
| Float | Numbers with decimal | 4 bytes | N/A |
| Double | Gets you an F | Self-explanatory  8bytes | N/A |
| Array | Stores ints and chars | Stores data in for my slot grid | main |

Functions

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Function** | **Description** | **Location** |
| void randGen (int [], int); | Generates random numbers for an array | Uses do while loop to fill 9 sections of array  a[ counter ]. Uses modulus 6 + 1 to ensure value between 1 and 6. | Line 228 |
| void prntArray(int [], int, int); | Outputs array that has been populated | Used to output the values of arrays used in program. Uses modulus 3 to end each line for formative of grid | Line 218 |
| void cvn\_char (int [], int, int); | Outputs letters for corresponding numbers | Uses for loop to sort through array and output in the form created by prntArray() | Line 193 |
| void prntSpc(); | Prints empty space | Uses for loop to output 25 end lines.  Used to clear the screen after inputs | Line 186 |
| bool is\_Winner(int [], int); | Tests columns horizontally and vertically for matching numbers | Tests populated array with corresponding patterns. Uses else if statements to figure out if each pattern is a winner. Winner returns true | Line 163 |
| bool ifSave(); | Determines if you want to load a saved game | If user decides to use a saved game, if returns a true value | Line 237 |
| string loadGame(); | Loads saved game value | Uses fstream library to load numbers store in “saved\_game.txt” Uses getline to read in the string stored in the file | Line 255 |
| int convertString (string); | Converts string to character array while adding up values from the file | Uses string.size to determine size of the string. The only files being stored are numbers. The string is plugged into a character array with a for-loop. To save time the function calls another function to convert the numbers while they go through the for-loop | Line 271 |
| int convertNum (char); | Used to change character to an int | Uses an int datatype in function. When a number between 0 and 9 occur it returns the integer | Line 296 |
| int thouConvert(int, int); | Converts single digits to their actual value | Uses 10 and n – 1 as the power. Uses size to determine n and multiplies the new total with the int passed through as a parameter. Works with large integers | Line 343 |
| void saveGame(int); | Saves your money | Opens “saved\_file.txt” and overwrites the old number with the new number. Closes the file | Line 352 |
| int placeBet(); | Reads in the bet you want to place | Outputs bet options and gets input as an integer. Uses switch case to return the value of the target bet | Line 141 |
| int prize(int); | Modifies bet | Uses modifier to multiply bet because you only win 1 every 5 times. So it is set to return 4 times the amount of money bet. | Line 136 |

FLOWCHART