

*Programming Assignment Sheet*

|  |  |
| --- | --- |
| To: | Troy Tuckett |
| From: | Elbio Iseas |
| Class: | PRG/410 |
| Date: | 11/9/2015 |
| Re: | Individual Assignment for Week Two |

|  |
| --- |
| Design: |
| Flow chart in word document :Prg410 - The Guessing Game Part One Flowchart - Elbio Iseas .docx |
|
| Source Program(s) : |
| // GuessingGameOne.cpp : Defines the entry point for the console application.  // Author : Elbio Iseas 2015  // ==========================================================================  #include "stdafx.h"  #include <iostream>  #include <iomanip>  #include <conio.h>  #include <cstdlib>  #include <ctime>  #include <string>  #include <windows.h>  using namespace std;  // ============================================================================================================  // https://msdn.microsoft.com/en-us/library/windows/desktop/ms686025(v=vs.85).aspx  // result of the online research for the old C language function for positioning the cursor  // The purspose of this function is to position the cursor on screen  // so there is no need to use endl or leading blanks  // ============================================================================================================  void gotoxy(int x, int y) // function with the two int parameters x, and y for column, and row respectively  {  HANDLE hStdOut = GetStdHandle(STD\_OUTPUT\_HANDLE);  COORD coord; // it creates an object COORD named coord for the screen coordinates  coord.X = x; // it assigns the value of the parameter x (column) to the object coord.X  coord.Y = y; // it assigns the value of the parameter y (row) to the object coord.Y  SetConsoleCursorPosition(hStdOut, coord); // it sets the cursor position with the values of the object coord  }  // ====================================================================================  // this function displays a message or character in the x, y coordinates of the console  // x is the column argument to display the message  // y is the line argument to display the message  // message is the argument you want to display at the coordinates x, y  // ====================================================================================  void say (int x, int y, string message)  {  gotoxy(x,y);  cout<< message;  }  // =====================================================  // this function when called it clear the console screen  // it uses the system OS command CLS  // =====================================================  void clearscreen()  {  system("CLS");  }  // ===============================================================  // this function makes the border of a box with five parameters :  // top left screen column coordinate  // top left screen line coordinate  // bottom right screen column coordinate  // bottom right screen line coordinate  // the parameter symbol can have one of three possible values  // 0 for blank  // 1 for single line  // 2 for double line  // ===============================================================  void drawboxborder(int x1, int y1, int x2, int y2, int symbol)  {  string topleftcorner,  toprightcorner,  bottomleftcorner,  bottomrightcorner,  topsymbol,  bottomsymbol,  leftsymbol,  rightsymbol;  if (symbol == 0)  {  topleftcorner = " ";  toprightcorner = " ";  bottomleftcorner = " ";  bottomrightcorner = " ";  topsymbol = " ";  bottomsymbol = " ";  leftsymbol = " ";  rightsymbol = " ";  }  else if (symbol == 1 || symbol == 2)  {  topleftcorner = ( symbol == 1 ) ? "Ú": "É";  toprightcorner = ( symbol == 1 ) ? "¿": "»";  bottomleftcorner = ( symbol == 1 ) ? "À": "È";  bottomrightcorner = ( symbol == 1 ) ? "Ù": "¼";  topsymbol = ( symbol == 1 ) ? "Ä" : "Í";  bottomsymbol = ( symbol == 1 ) ? "Ä" : "Í";  leftsymbol = ( symbol == 1 ) ? "³" : "º";  rightsymbol = ( symbol == 1 ) ? "³" : "º";  }  if (symbol >= 0 && symbol <= 2)  {  int col, line;  say(x1,y1,topleftcorner);  say(x2,y1,toprightcorner);  for (col = x1+1; col <= x2-1; col++)  say(col,y1,topsymbol);  for (line = y1+1; line <= y2-1; line++)  {  say(x1,line,leftsymbol);  say(x2,line,rightsymbol);  }  say(x1,y2,bottomleftcorner);  say(x2,y2,bottomrightcorner);  for (col = x1+1; col <= x2-1; col++)  say(col,y2,bottomsymbol);  }  }  // =======================================================================  // this function paints a screen box with the symbol passed as an argument  // x1 is the left column value of the top left corner  // y1 is the top line value for the top left corner  // x2 is the right column value of the bottom right corner  // y2 is the bottom line value of the bottom right corner  // symbol is the character you pass as argument to fill the box  // =======================================================================  void paintbox(int x1, int y1, int x2, int y2, string symbol)  {  int col, line;  for (line = y1; line <= y2; line++)  {  for (col = x1; col <= x2; col++)  say(col,line,symbol);  }  }  // ============================================================================================================  // This function paints the main screen for this program  // ============================================================================================================  void displayScreenOne()  {  clearscreen(); // it clears the screen  drawboxborder(10,1,68,3,1); // it displays a box with the coordinates provided with single line border  say(12, 2, "T H E G U E S S I N G G A M E - P A R T O N E"); // it displays the title of the program  drawboxborder(10,5,68,19,1); // it displays a box with single line for the coordinates provided  say(20, 9,"Your Computer's Secret Number is : "); // it display the message at the specified position  say(20,11,"Your Best Guess Number is : "); // it display the message at the specified position  say(20,15,"Type your Guess Number (1 - 99) : "); // it display the message at the specified position  drawboxborder(0,21,79,23,1); // it displays a box with single line for the coordinates provided  }  // =============================================================================================================  // This function generates a random number in the range minValue to maxValue arguments  // and returns an integer value in that range  // =============================================================================================================  int randomize (int minValue, int maxValue)  {  int randnum; // random number  unsigned int seed; // used in association with srand() and rand() to randomize numbers  seed = time(0); // needed to generate random numbers  srand(seed); // needed to generate random numbers  randnum = rand() % maxValue + minValue; // generates a random number in the range minValue to maxValue  return randnum; // it returns the random number  }  // =============================================================================================================  // this functions accepts input from the user in the range specified by the constants minnum and maxnum  // and it returns a valid integer value in the right range.  // =============================================================================================================  int userinput (int minnum, int maxnum)  {  int guess = 0; // it declares and initializes an int var with 0;  bool ready = false; // it creates and assigns the value of false to the var  while ( !ready ) // it stays doing the loop while it is not ready  {  say(56,15," "); // it clears the screen at the position established  gotoxy(56,15); //it places the cursor in the location established  cin >> guess; // it accepts an integer value and it assigns it to the variable guess  if (guess >= minnum && guess <= maxnum) // it asks if the number entered is in the range  {  ready = true; // assigns the value of true to the boolean variable ready  }  else  {  say(8,22,"Type your best guess ( 1 - 99 ). Press any key to continue ... "); // it displays an error message  getch(); // it pauses the program and it waits for the user to press any key  say(8,22," "); // it erases the message  }  }  return guess; // it returns the number entered by the user  }  // =============================================================================================================  // This function doesn't return any value, and it accepts two arguments:  // the random number the computer created, and the guessed number the user entered.  // =============================================================================================================  void verifynum (int randnum, int usernum)  {  gotoxy(56,9); // it positions the cursor at the coordinates location  cout << randnum; // it displays the computers number  gotoxy(56,11); // it positions the cursor at the coordinates location  cout << usernum; // it displays the user's guessed number    if (usernum == randnum) // it checks if the number guessed is equal to the computer's generated number  {  say(2,22,"Good job!!! Your guess was correct. Press any key to guess another number ... "); // it displays this message  }  else if (usernum > randnum) // it checks if the number guessed is greater than the computer's random number  {  say(2,22,"Sorry, your guessed was too high. Press any key to guess another number ... "); // it displays this message  }  else if (usernum < randnum) // it checks if the number guessed is smaller than the computer's random number  {  say(2,22,"Sorry, your guessed was too low. Press any key to guess another number ... "); // it displays this message  }  getch(); // it waits for the user to press a key  say(2,22," "); // it erases the last message6  say(56, 9," "); // it erases the old value from screen  say(56,11," "); // it erases the old value from screen  say(56,15," "); // it erases the old value from screen  }  int \_tmain(int argc, \_TCHAR\* argv[])  {  HANDLE hConsole = GetStdHandle(STD\_OUTPUT\_HANDLE);  SMALL\_RECT windowSize = { 0,0,80,24 }; // it creates an object with the screen size  SetConsoleWindowInfo(hConsole, TRUE, &windowSize); // it sets the console with the windowSize values  displayScreenOne(); // it displays the first screen  const int maxnum = 99; // it defines a constant with the value of 99 (highest number to guess)  const int minnum = 1; // it defines a constant with the value of 1 (lowest number to guess)  int counter = 0; // it defines an int variable to control the times of a guessing number  int maxguesses = 10; // it defines an int variable the amount of guesses  int randnumber = 0; // it defines an int variable to store the random number obtained by the computer  int userguess = 0; // if defines an int variable to store the guessed number typed by the user    while (counter < maxguesses) // it will stay on the while loop while counter is smaller than maxguesses  {  randnumber = randomize(minnum,maxnum); // it calls the randomize functions with the arguments  // minnum and maxnum to create a random number and it returns  // the random number generated by the computer and it stores it  // in the int variable randnumber  userguess = userinput(minnum, maxnum); // it calls the userinput function with the arguments randnumber  // and maxnum to accept the user's input and it stores its value  // in the int variable userguess  verifynum(randnumber, userguess); // it calls the function verifynum for comparing the random number  // and the user's typed number and displaying both numbers with a  // message if it was a match, lower or higher guessed number  counter++; // it increments the counter by one  }  say(14,22,"I hope you we can play again soon. Goodbye. Click. "); // it displays the Goodbye message  getch(); // it pauses the execution of the program until a key is pressed  return 0;  } |
| Output Results: |
| Screen shots in .png files |
| Testing: |
| Description of testing |
| Tested By Elbio Iseas |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test # | Random num | User's guessed num | Number in range (1-99) | Too low | Too high | Match |
| 1 | 39 | 18 | PASS | PASS |  |  |
| 2 | 27 | 37 | PASS |  | PASS |  |
| 3 | 16 | 99 | PASS |  | PASS |  |
| 4 | 78 | 1 | PASS | PASS |  |  |
| 5 | 35 | 22 | PASS | PASS |  |  |
| 6 | 87 | 48 | PASS | PASS |  |  |
| 7 | 34 | 71 | PASS |  | PASS |  |
| 8 | 58 | 12 | PASS | PASS |  |  |
| 9 | 97 | 95 | PASS | PASS |  |  |
| 10 | 53 | 65 | PASS |  | PASS |  |