**CPSC1012 Core Portfolio 4 – Methods**

**Weight: 5% of your final mark**

**Introduction**

In this assignment you will design and write a modularized menu-driven program that allows the user to select one of two games (Craps and Pig) to play or to quit the program. The description of the two games are described next.

**Game of Craps**

In the game of craps, a pass line bet proceeds as follows: Two six-sided dice are rolled; the first roll of the dice in a craps round is called the "come out roll." A come out roll of 7 or 11 automatically wins, and a come out roll of 2, 3, or 12 automatically loses. If 4, 5, 6, 8, 9, or 10 is rolled on the come out roll, that number becomes "the point." The player keeps rolling the dice until either 7 or the point is rolled. If the point is rolled first, then the player wins the bet. If a 7 is rolled first, then the player loses.

Write code to play a variation of the game as follows:

* Ask the user for a wager amount
* Roll two six-side dice.
* Check the sum of the two dice.
  + If the sum is 2, 3, or 12 (called craps), you lose.
  + If the sum is 7 or 11 (called naturals), you win.
  + If the sum is another value (i.e., 4, 5, 6, 7, 8, 9, or 10), a point is established. Continue to roll the dice until either a 7 or the same point value is rolled. If 7 is rolled, you lose. Otherwise you win.
* Ask the user if they want to play another round
* If the user does not want to play another round, display the total amount of money they won or lost.

Here is a sample run:

-----------------

| Game of Craps |

-----------------

Enter amount to bet: 20

You rolled 5 + 6 = 11

You win $20.00

Do you want to play again (y/n): y

Enter amount to bet: 5

You rolled 1 + 2 = 3

You lost $5.00

Do you want to play again (y/n): y

Enter amount to bet: 5

You rolled 4 + 4 = 8

Point is 8

You rolled 2 + 6 = 8

You win $5.00

Do you want to play again (y/n): y

Enter amount to bet: 5

You rolled 2 + 3 = 5

Point is 5

You rolled 5 + 2 = 7

You lose $5.00

Do you want to play again (y/n): y

Enter amount to bet: 5

You rolled 3 + 3 = 6

Point is 6

You rolled 5 + 3 = 8

You rolled 5 + 5 = 10

You rolled 1 + 2 = 3

You rolled 5 + 4 = 9

You rolled 4 + 3 = 7

You lose $5.00

Do you want to play again (y/n): n

Your net winning is $10.00

**Game of Pig**

The game of Pig is a simple two-player dice game in which the first player to reach 100 or more points wins. Players take turns. On each turn, a player rolls a six-sided die:

* If the play rolls a 1, then the player gets no new points and it becomes the other player’s turn.
* If the player rolls 2 through 6, then they can either:
  + ROLL AGAIN or
  + HOLD. At this point, the sum of all rolls is added to the player’s score, and it becomes the other player’s turn.

Write code to play the game of Pig, where one player is a human and the other is the computer. When it is the human’s turn, the program shows the score of both players and the previous roll. Allow the human to input **‘r’** to roll again or **‘h’** to hold.

The computer program should play according to the following rules:

* Keep rolling when it is the computer’s turn until it has accumulated 10 or more points, then hold. If the computer wins or rolls a 1, then the turn ends immediately.

Allow the human to roll first.

Here are some sample runs:

---------------

| Game of Pig |

---------------

Enter the point total to play for: 20

It’s your turn.

You rolled a 6

Enter r to roll or h to hold (r/h): r

You rolled a 3

Enter r to roll or h to hold (r/h): r

You rolled a 4

Enter r to roll or h to hold (r/h): r

You rolled a 3

Enter r to roll or h to hold (r/h): r

You rolled a 5

Your turn point total is 21

You WIN

---------------

| Game of Pig |

---------------

Enter the point total to play for: 20

It’s your turn.

You rolled a 6

Enter r to roll or h to hold (r/h): r

You rolled a 6

Enter r to roll or h to hold (r/h): r

You rolled a 6

Enter r to roll or h to hold (r/h): r

You rolled a 1

Your turn score is 0

Your total points: 0

Computer total points: 0

It’s the computer’s turn.

Computer rolled a 6

Computer rolled a 5

Computer HOLD

Computer turn score is 11

Your total points: 0

Computer total points: 11

It’s your turn.

You rolled a 6

Enter r to roll or h to hold (r/h): r

You rolled a 6

Enter r to roll or h to hold (r/h): h

You HOLD

Your turn score is 12

Your total points: 12

Computer total points: 11

It’s the computer’s turn.

Computer rolled a 6

Computer rolled a 2

Computer rolled a 1

Computer turn total is 0

Your total points: 12

Computer total points: 11

It’s your turn.

You rolled a 2

Enter r to roll or h to hold (r/h): r

You rolled a 4

Enter r to roll or h to hold (r/h): r

You rolled a 3

Your turn score is 9

You WIN

Your total points: 21

Computer total points: 11

**Program Menu**

Create a program menu to allow the user to select which game to run or exit the program. Menu choice input validation is required. Here is a sample run:

|------------------|

| CPSC1012 Casino |

|------------------|

| 1. Play Craps |

| 2. Play Pig |

| 0. Exit Program |

|------------------|

Enter your menu number choice > x

x is not a valid menu choice. Try again.

Enter your menu number choice > 1

-----------------

| Game of Craps |

-----------------

…

|------------------|

| CPSC1012 Casino |

|------------------|

| 1. Play Craps |

| 2. Play Pig |

| 0. Exit Program |

|------------------|

Enter your menu number choice > 2

---------------

| Game of Pig |

---------------

…

|------------------|

| CPSC1012 Casino |

|------------------|

| 1. Play Craps |

| 2. Play Pig |

| 0. Exit Program |

|------------------|

Enter your menu number choice > 0

Good-bye and thanks for coming to the CPSC1012 casino.

Design your program to use methods where each method cannot have more than 50 statements. Your program must handle invalid input values and not crash.

**Marking Guide**

|  |  |  |
| --- | --- | --- |
| **Description** | **Marks Possible** | **Marks Earned** |
| Correctness   * Program Menu (4 marks)   + Able to run a selected game or quit   + Loops back after running a game   + Exits Program   + Invalid menu choice validation * Game of Craps (4 marks)   + Come-out-roll win or lose logic   + Point roll win or lose logic   + Can play around round   + Tracks total amount won/lost * Game of Pig (5 marks)   + Human Player Turn logic (2)   + Computer Player Turn logic (2)   + Game over logic (1) | 13 |  |
| Structure   * Appropriate use of methods (2) * Appropriate use of method parameters/arguments (2) * No static variables (1) * Invalid input value exception handling (1) | 6 |  |
| Style and Readability   * Horizontal white space * Vertical white space * Meaningful identifiers | 3 |  |
| Documentation   * Opening documentation * Source code comments | 3 |  |
| **Total:** | **25** |  |

**Coding Requirements**

The following coding standards must be followed when developing your program:

* Your C# Console App project must be named as **CorePortfolio04-*YourFullName*** (eg: CorePortfolio04-CodeGuru)
* Opening documentation at the beginning of the source file describing the **purpose**, **input**, **process**, **output, author, last modified date** of the program.
* Write only one statement per line.
* Write only one declaration per line.
* Use camelCase for local variable names.
* Use PascalCase for method names and constant variable names.
* If continuation lines are not indented automatically, indent them one tab stop (four spaces).
* Do NOT use the goto statement.
* There can only be one exit point for a loop, do not use the break statement inside a loop

Submission **Requirements**

* Submit a compressed (zip) copy of your Visual Studio 2019 project folder to Moodle on or before the due date.

using System;

namespace CorePortfolio04\_MayThuyNguyen

{

class Program

{

static void DisplayMenu()

{

Console.Clear();

Console.WriteLine("| ------------------|");

Console.WriteLine("| CPSC1012 Casino |");

Console.WriteLine("| ------------------|");

Console.WriteLine("| 1.Play Craps |");

Console.WriteLine("| 2.Play Pig |");

Console.WriteLine("| 0.Exit Program |");

Console.WriteLine("| ------------------|");

}

static int GetChoice()

{

Console.Write("Please enter your menu number choice: ");

int choice = int.Parse(Console.ReadLine());

switch (choice)

{

case 1:

CrapsGame();

break;

case 2:

PigGame();

default:

Console.WriteLine($"{choice} is not a valid menu choice. Try again.");

return GetChoice();

}

}

static int CrapsGame()

{

string playagain = "y", status = "continue";

int point = 0;

Console.Write("Enter amount to bet: ");

double amount = double.Parse(Console.ReadLine());

Random rand = new Random();

int die1 = rand.Next(1, 6);

int die2 = rand.Next(1, 6);

int total = (die1 + die2);

Console.Write($"You roll {die1} + {die2} = {total}");

switch (total)

{

case 2:

case 3:

case 12:

Console.Write($"You lost {amount:c}");

status = "lost";

break;

case 7:

case 11:

Console.Write($"You won {amount:c}");

status = "won";

break;

default:

Console.Write("Let's roll again");

point = total;

status = "continue";

break;

}

while (status == "continue")

{

die1 = rand.Next(1, 6);

die2 = rand.Next(1, 6);

total = (die1 + die2);

if (total == 7)

{ Console.Write($"You lost {amount:c}"); }

else if (total == point)

{ Console.Write($"You won {amount:c}"); }

else { status = "continue"; }

}

while (playagain != "n" && playagain != "y")

{

Console.WriteLine("Do you wanna play again? (y/n): ");

playagain = Console.ReadLine().ToLower();

if (playagain == "y")

{ CrapsGame(); }

else if (playagain == "n")

{ Console.WriteLine("Your net winning is: "); }

else

{

Console.WriteLine("Invalid option. Please try again.");

}

}

}

static int pigRoll()

{

Random random = new Random();

int die1 = 0;

die1 = random.Next(1, 6);

Console.WriteLine($"You rolled {die1}");

return die1;

}

static double humanTurn()

{

double pointTotal = 0;

string gameSelect = null;

var pigDiceRoll = 0;

Console.WriteLine("It's your turn");

do

{

pigDiceRoll = pigRoll();

if (pigDiceRoll != 1)

{

Console.WriteLine("r to roll or h to hold (r/h)");

gameSelect = Console.ReadLine();

pointTotal = pointTotal + pigDiceRoll;

}

else if (pigDiceRoll == 1)

{

pointTotal = 0;

}

} while (gameSelect != "r" || pigDiceRoll != 1);

Console.WriteLine($"Your turn point total is {pointTotal}");

return pointTotal;

}

static void Main(string[] args)

{

Console.WriteLine("Hello World!");

}

}

}