**PROGRAMMING FUNDAMENTALS (CSC-103)**

**SEMESTER PROJECT**



|  |  |
| --- | --- |
| **SUBMITTING BY:** | Ayyat Fatima FA21-BSE018  Muaaz Bin Mukhtar FA21-BSE-045  Aayan Javed FA21-BSE-001 |
| **CLASS & SECTION:** | BSE-2A |
| **SUBMITTED TO:** | DR MUZAFFAR IQBAL |
| **DATE SUBMITTED:** | 13-06-2022 |

**Department of Computer Science**

Gaming Arcade

Welcome to the Arcade of ur

Childhood Games

Games included are:

1. Single Player Tic Tac Toe
2. Multi Player Tic Tac Toe
3. Hangman
4. Hand Cricket
5. Rock Paper Scissors
6. What Next?
7. Guess the Riddle
8. Mind Reader
9. Single Player Tic Tac Toe:

Description:

Single player Tic Tac Toe is one of the most remarkable childhood games, in which 3 rows and 3 columns are available and we will be playing against the computer.

Whoever gets 3 in a row wins.

Working:

Single player tic tac toe is a simple game in which 3 rows and 3 columns are available and we have to trick the computer for making a row of 3 same objects.

Working of Code:

Firstly, we have declared a 1D Array for storing 'X's and 'O's , with respect to user choice and computer.

Initially I have entered numbers 1-9 in this array and displayed the tic tac toe grid and marked the positions with these numbers.

Using if conditional statement, we will compare user’s choice or computer’s choice with these numbers and respectively 'X' or 'O' will be market to that position and displayed in the grid on that specific position.

If that position is already occupied user will be informed to enter again.

This whole game is running in a while(true) loop and it will exit when a player/computer wins or it’s a draw.

Using if statements and logical operators we have compared all the possible solutions and if any condition is true, the loop exits using break statement.

User/computer turn is detected by variable "turn". If it is even then computer's turn else its user's turn.

For computer's choice, I have made an intelligence method which returns the position number according to game condition.

It is a defensive method which enters the third position for winning game if user has made 2 positions and there is a possibility of winning on that third position.

Switch statement is used for comparing positions and declaring the choice.

Techniques used:

1. Loop Control Statements

2. Conditional Control Statements

3. Methods

4. Arrays

Classes used:

1. Main Class

2. Scanner Class

3. Math Class

1. Multiplayer Tic Tac Toe:

Description:

Multi player tic tac toe is a simple game of two players in which 3 rows and 3 columns are available and we have to trick the other player in allowing us to make a 3 in a row.

Working:

Firstly, we have declared a 1D Array for storing 'X's and 'O's ,with respect to user choice and computer.

Initially, I have entered numbers 1-9 in this array and displayed the tic tac toe grid and marked the positions with these numbers.

Using if condition we will compare user’s choices with these numbers and respectively 'X' or 'O' will be added to that position and displayed in the grid on the position of that number.

If that position is already occupied user will be informed to enter again.

This whole game is running in while(true) loop and it will exit when a player wins or it’s a draw.

Using if statements and logical operators we have compared all the possible solutions and if any condition is true, the loop exits using break statement.

User turn is detected by variable "turn". If it is even then Player 2’s turn, else its Player 1’s turn.

Switch statement is used for comparing positions and declaring the choice

Techniques used:

1. Loop Control Statements

2. Conditional Control Statements

3. Methods

4. Arrays

Classes used:

1. Main Class

2. Scanner Class.

1. Hangman:

Description:

Hangman is a game in which player has to guess a word letter by letter. The concept is that we have 6 chances and we have to save the person from hanging till death by guessing word. Each wrong guess brings him closer to death.

Working:

Firstly, I have made an object of file class to access the file with words. Then made an object of scanner class to read all the words from file. Then added all these words in an array list one by one.

Then we have used Math.random to randomly access any word from array list and by using loop we show dashes on screen equal to number of words. This random word string is converted into char array.

While loop is used for guessing word, it will end if user guesses the word or loses the game i-e, chances are over.

Then we have made another method to display the person that we have to save from hanging.

Another char array is declared of same length as word array and dashes are entered on each location for display and comparing. When user enters something firstly it is ensured that single character is entered and no digit is entered. Then whole array of word is checked on all positions one by one if user's guess matches any word.

If no character is matched life is detected. The characters which are matched with user's guess are shown on screen and entered in the array. Now it is checked if there are any dashes left in word i-e. any character left to guess then loop continues else it breaks and user wins.

Then if lives are 0 the loop breaks and user loses.

Techniques used:

1. Loop Control Statements

2. Conditional Control Statements

3. Methods

4. Exception Handling

5. File Handling

6. Arrays

Classes used:

1. Main Class

2. Scanner Class

3. ArrayList Class

4. File Class

5. IOException Class

6. Random Class

1. Hand Cricket:

Description:

Hand cricket is a game that we used to play in our childhood. In this game first the toss is decided where a player chooses heads or tails then chooses a number computer also chooses a number both are added if it is even, it is tails and if it is odd, it is heads. If user wins, he decides to bat or ball and if loses then computer chooses whether to bat or ball .... they both have choice from 1 to 6 if the numbers match batting one is out if not choice of score is added to score of the player (either user or computer ). 2 innings are played

Scores are compared higher score wins !

Working:

This game is based on complex looping techniques. First of all, a set of rules is printed for user information then toss essentials are extracted from user for example what he chooses heads or tails and a number from 1 to 5!

Then computer choice from 1 to 5 is generated !

Then all the variables needed in the game are declared. Then on the basis of number 1 for heads 2 for tails selective toss loops are entered for example if chosen heads than if it is even than user loses if it is even user wins . Also, if user wins, check is iterated +1, if loses +3. This iteration is further used to separate the user winning loop and user losing loop.

If user wins, the user chooses if he wants to bat or bowl with the help of numbers 1 for batting 2 for balling, then user choice from 1 to 6 is entered and computer generates a number too between 1 to 6. If both score matches 2nd inning loop start where other has to start. For example, if user was batting, score matches, the loops break and computer starts batting. In first loop, user score was iterated with the choice before he was out and loop was broken. When broken computer’s score is iterated and if it reaches the target loops break and computer has won the game else user wins . It happens vice versa when user balls first. The one who wins is declared.

Techniques used:

1. Methods
2. Conditional Statements
3. Loops
4. Nested loops

Classes used:

1. Random class
2. Scanner class
3. Random class

5. Rock Paper Scissors:

Description:

Basically, Rock Paper Scissors is a game of luck and chance. In this game, you have to choose 3 options.

1. Rock
2. Paper
3. Scissors

**Rock crushes scissors.**

**Paper covers rock.**

**Scissors cuts paper.**

In this way, each element crushes another in a circular manner!

Working:

The game is run in a while loop which runs 5 times before the condition is met. User score, computer score and counter are defined before while loop because they are to be displayed outside the loop. In while loop, user has to choose 1 out of 3 elements and a computer choice is generated with the help of random class. Rest of the game works in if conditional statement. Each element is assigned a number 1, 2 or 3 and the desired loop is executed based upon the user’s choice. Inside each if condition, there are further 3 conditions which are for computer’s choice and comparison of user’s choice with the computer’s choice. According to the rules of the game, if user wins user score is iterated, if both chose the same element, it is a draw and if computer wins computer score is iterated. This process is repeated 5 times.

When counter reaches the limit of the condition, while loop exits and user’s score and computer’s score are displayed then both are compared to decide who has won the game or is it a draw?

Techniques used:

1. Methods
2. Conditions
3. Loops
4. nested loops

Classes used:

1. Random class
2. Scanner class
3. Main class

6. What Next?

Description:

What Next is a text based adventure game in which each choice leads to a certain consequence. A game that follows an original story line. A game that merely tests your decision making skills. If you choose wisely, you may win but if you choose incorrect, you may lose the game.

Working:

The coding begins with initializing the variables used within the game. The game starts within a while(true) loop.

First statement of the story is displayed along with certain choices using if-else conditional statements and from here the while story will be based upon our choices. When the desired option is chosen, the next part of the story is displayed and asks us to choose what happens next. Here two different options lead to two different outcomes and this continues throughout the game. The next three options must be chosen wisely in order for the player to survive for further game.

A certain point in the game reaches where computer generates a choice for us using Random class. If option 1 is chosen, the player survives and wins the game instantly. In contrast, if option 2 is chosen, the game continues displaying the rest of the story.

All of the working happens within a while(true) loop which breaks when the story line changes and remains to continue for a different story line.

A very interesting game for captivating the user using multiple choices.

Techniques used:

1. Conditional statements
2. Loops
3. Nested loops
4. Break statement

Classes used:

1. Main class
2. Random class
3. Scanner class

7. Mind Reader:

Description:

Mind reader is a personally favorite childhood game which my brother used to play with me. In this game, the computer asks the user to think of a word and it will guess it. A very interesting game which guesses every word the user thinks of despite its spelling or language.

Working:

The coding begins with initializing the variables and arrays used within the game. Two arrays are initialized with two different arrangement of the alphabets. When the game begins, the computer asks the user how many alphabets does the word contain. Then a matrix with a certain arrangement of the alphabets is displayed and asks one by one in which row does the first letter exists, then the second, the third and so on, till the last letter.

Then the second arrangement is displayed and again asks for the row numbers. When all of the rows are given to the computer as input, it displays the word and it all feels like magic. But the logic behind is that it compares the rows of both the arrangement and finds the common alphabet between them which gives us all the letters of the word that the user has thought of.

A very crafty game to catch people’s attention.

Techniques used:

1. Arrays
2. Loops
3. Nested loops

Classes used:

1. Scanner class
2. Main class

8. Guess the Riddle:

Description:

Guess the riddle is fun game for your brain in which you will have to guess single word answer of the riddle. If the user does not know the answer, he can take a hint but will lose a live. Computer will tell the user that the first letter of the word exists in the certain row and go search for it in the word cross puzzle.

A fun activity for your brain.

Working:

The coding begins with initializing the arrays for storing the coordinates of all the hints and an array for storing all the answers.

Then we are using File Handling to firstly create a file, providing it a proper path and then writing our riddles in it. After all the riddles are written within the file, we will close it. All of the creating and writing will be done in try-catch for handling the exceptions.

Then we will use 2 dimensional arrays to create the grid of the word cross puzzle and use nested for loops and conditional statements to display it.

Now, we will read the file that we have created. Initialize all the variables that will be needed further execution. We are using a while loop which will run for each line of the file one by one. Within the loop, we are declaring another scanner object which will be used for appending the answer given by user in an array. User will be given 6 lives. Then one by one each riddle will be displayed and user will be asked if he wants a hint or not. If he asks for a hint, a live will be deducted and the row will the told, if not then only the computer will ask for the answer.

The program will check if the entered answer matches with the answers of the riddles that we have stored Earlier and increment counter variable by one. If user runs out of lives, the loop breaks and a message is displayed of the number of correct answers. All of the reading of the file will again be done in try-catch to handle the exceptions.

Techniques used:

1. Loops

2. Conditional Statements

3. Exception Handling

4. File Handling

5. Arrays

Classes used:

1. Main Class

2. Scanner Class

3. File Class

4. IOException Class

Contributions:

1. Ayyat Fatima ( FA21-BSE-O18 ):

* What next?
* Mind Reader
* Guess the Riddle

1. Muaaz Bin Mukhtar ( FA21-BSE-045):

* Single Player Tic Tac Toe
* Multi Player Tic Tac Toe
* Hangman

1. Aayan Javed ( FA21-BSE-001 )

* Hand Cricket
* Rock Paper Scissors

SOURCE CODE:

package com.mycompany.semesterproject;

import java.io.\*;

import java.util.ArrayList;

import java.util.Random;

import java.util.Scanner;

public class Semesterproject

{

public static void main(String[] args) throws IOException

{

Scanner input=new Scanner(System.in);

System.out.println("WELCOME TO THE ARCADE OF OUR CHILDHOOD GAMES");

System.out.println("1.Single Player Tic Tac Toe");

System.out.println("2.Double Player Tic Tac Toe");

System.out.println("3.Hangman");

System.out.println("4.Mind Reader");

System.out.println("5.What Next?");

System.out.println("6.Guess The Riddle");

System.out.println("7.Rock Paper Scissor");

System.out.println("8.Hand Cricket");

System.out.println("Which game you want to play:");

int gamechoice=input.nextInt();

switch(gamechoice)

{

case 1:

ticTacToeSingle();

break;

case 2:

ticTacToe();

break;

case 3:

hangman();

break;

case 4:

mindReader();

break;

case 5:

whatNext();

break;

case 6:

wordCross();

break;

case 7:

rps();

break;

case 8:

handCricket();

break;

}

}

public static void ticTacToe()

{

Scanner input=new Scanner(System.in);

char[] position={'1','2','3','4','5','6','7','8','9'};

System.out.println("WELCOME TO TIC TAC TOE");

System.out.println("Enter player 1 name:");

String player1=input.next();

System.out.println("Enter player 2 name:");

String player2=input.next();

System.out.println(player1+"=X\n"+player2+"=O");

int turn=1;

while(true)

{

System.out.println( position[0]+" | "+position[1]+" | "+position[2] );

System.out.println("-----------");

System.out.println( position[3]+" | "+position[4]+" | "+position[5] );

System.out.println("-----------");

System.out.println( position[6]+" | "+position[7]+" | "+position[8] );

if(turn%2==0)

{

System.out.println(player2+"'s turn");

}

else

{

System.out.println(player1+"'s turn");

}

System.out.println("Enter the available position:");

int choice=input.nextInt();

switch(choice)

{

case 1:

case 2:

case 3:

case 4:

case 5:

case 6:

case 7:

case 8:

case 9:

if(position[choice-1]=='O'||position[choice-1]=='X')

{

System.out.println("POSITION OCCUPIED!!");

break;

}

if(turn%2==0)

{

position[choice-1]='O';

}

else

{

position[choice-1]='X';

}

turn++;

break;

default:

System.out.println("Wrong choice!!");

//continue;

}

if((position[0]=='O' && position[1]=='O' && position[2]=='O')||

(position[3]=='O' && position[4]=='O' && position[5]=='O')||

(position[6]=='O' && position[7]=='O' && position[8]=='O')||

(position[0]=='O' && position[3]=='O' && position[6]=='O')||

(position[1]=='O' && position[4]=='O' && position[7]=='O')||

(position[2]=='O' && position[5]=='O' && position[8]=='O')||

(position[0]=='O' && position[4]=='O' && position[8]=='O')||

(position[2]=='O' && position[4]=='O' && position[6]=='O'))

{

System.out.println(player2 +" WINS!!");

break;

}

else if((position[0]=='X' && position[1]=='X' && position[2]=='X')||

(position[3]=='X' && position[4]=='X' && position[5]=='X')||

(position[6]=='X' && position[7]=='X' && position[8]=='X')||

(position[0]=='X' && position[3]=='X' && position[6]=='X')||

(position[1]=='X' && position[4]=='X' && position[7]=='X')||

(position[2]=='X' && position[5]=='X' && position[8]=='X')||

(position[0]=='X' && position[4]=='X' && position[8]=='X')||

(position[2]=='X' && position[4]=='X' && position[6]=='X'))

{

System.out.println(player1+" WINS!!");

break;

}

else if(position[0]=='1'||position[1]=='2'||position[2]=='3'||

position[3]=='4'||position[4]=='5'||position[5]=='6'||

position[6]=='7'||position[7]=='8'||position[8]=='9')

{

}

else

{

System.out.println("DRAW!!");

break;

}

}

}

public static void ticTacToeSingle()

{

Scanner input=new Scanner(System.in);

char[] position={'1','2','3','4','5','6','7','8','9'};

System.out.println("Welcome to Tic Tac Toe");

System.out.println("You will be playing against computer, computer will only defend\nLet's see how you trick computer!!");

System.out.println("Enter your name:");

String player=input.next();

System.out.println(player+"=X\nComputer=O");

int turn=1;

while(true)

{

System.out.println( position[0]+" | "+position[1]+" | "+position[2] );

System.out.println("-----------");

System.out.println( position[3]+" | "+position[4]+" | "+position[5] );

System.out.println("-----------");

System.out.println( position[6]+" | "+position[7]+" | "+position[8] );

int choice;

if(turn%2==0)

{

System.out.println("Computer turn");

choice=ticTacToeIntelligence(position);

}

else

{

System.out.println(player+"'s turn");

System.out.println("Enter the available position:");

choice=input.nextInt();

}

switch(choice)

{

case 1:

case 2:

case 3:

case 4:

case 5:

case 6:

case 7:

case 8:

case 9:

if(position[choice-1]=='O'||position[choice-1]=='X')

{

System.out.println("POSITION OCCUPIED!!");

break;

}

if(turn%2==0)

{

position[choice-1]='O';

}

else

{

position[choice-1]='X';

}

turn++;

break;

default:

System.out.println("Wrong choice!!");

continue;

}

if((position[0]=='O' && position[1]=='O' && position[2]=='O')||

(position[3]=='O' && position[4]=='O' && position[5]=='O')||

(position[6]=='O' && position[7]=='O' && position[8]=='O')||

(position[0]=='O' && position[3]=='O' && position[6]=='O')||

(position[1]=='O' && position[4]=='O' && position[7]=='O')||

(position[2]=='O' && position[5]=='O' && position[8]=='O')||

(position[0]=='O' && position[4]=='O' && position[8]=='O')||

(position[2]=='O' && position[4]=='O' && position[6]=='O'))

{

System.out.println( position[0]+" | "+position[1]+" | "+position[2] );

System.out.println("-----------");

System.out.println( position[3]+" | "+position[4]+" | "+position[5] );

System.out.println("-----------");

System.out.println( position[6]+" | "+position[7]+" | "+position[8] );

System.out.println("COMPUTER WINS!!");

break;

}

else if((position[0]=='X' && position[1]=='X' && position[2]=='X')||

(position[3]=='X' && position[4]=='X' && position[5]=='X')||

(position[6]=='X' && position[7]=='X' && position[8]=='X')||

(position[0]=='X' && position[3]=='X' && position[6]=='X')||

(position[1]=='X' && position[4]=='X' && position[7]=='X')||

(position[2]=='X' && position[5]=='X' && position[8]=='X')||

(position[0]=='X' && position[4]=='X' && position[8]=='X')||

(position[2]=='X' && position[4]=='X' && position[6]=='X'))

{

System.out.println( position[0]+" | "+position[1]+" | "+position[2] );

System.out.println("-----------");

System.out.println( position[3]+" | "+position[4]+" | "+position[5] );

System.out.println("-----------");

System.out.println( position[6]+" | "+position[7]+" | "+position[8] );

System.out.println(player+" WINS!!");

break;

}

else if(position[0]=='1'||position[1]=='2'||position[2]=='3'||

position[3]=='4'||position[4]=='5'||position[5]=='6'||

position[6]=='7'||position[7]=='8'||position[8]=='9')

{

}

else

{

System.out.println("DRAW!!");

break;

}

}

}

public static int ticTacToeIntelligence(char[] x)

{

int c;

if((x[0]=='X')&&(x[1]=='X')&&(x[2]!='O')&&(x[2]!='X'))

c=3;

else if((x[0]=='X')&&(x[2]=='X')&&(x[1]!='O')&&(x[1]!='X'))

c=2;

else if((x[1]=='X')&&(x[2]=='X')&&(x[0]!='O')&&(x[0]!='X'))

c=1;

else if((x[3]=='X')&&(x[4]=='X')&&(x[5]!='O')&&(x[5]!='X'))

c=6;

else if((x[3]=='X')&&(x[5]=='X')&&(x[4]!='O')&&(x[4]!='X'))

c=5;

else if((x[4]=='X')&&(x[5]=='X')&&(x[3]!='O')&&(x[3]!='X'))

c=4;

else if((x[6]=='X')&&(x[7]=='X')&&(x[8]!='O')&&(x[8]!='X'))

c=9;

else if((x[6]=='X')&&(x[8]=='X')&&(x[7]!='O')&&(x[7]!='X'))

c=8;

else if((x[7]=='X')&&(x[8]=='X')&&(x[6]!='O')&&(x[6]!='X'))

c=7;

else if((x[0]=='X')&&(x[4]=='X')&&(x[8]!='O')&&(x[8]!='X'))

c=9;

else if((x[0]=='X')&&(x[8]=='X')&&(x[4]!='O')&&(x[4]!='X'))

c=5;

else if((x[4]=='X')&&(x[8]=='X')&&(x[0]!='O')&&(x[0]!='X'))

c=1;

else if((x[6]=='X')&&(x[4]=='X')&&(x[2]!='O')&&(x[2]!='X'))

c=3;

else if((x[6]=='X')&&(x[2]=='X')&&(x[4]!='O')&&(x[4]!='X'))

c=5;

else if((x[4]=='X')&&(x[2]=='X')&&(x[6]!='O')&&(x[6]!='X'))

c=7;

else if((x[0]=='X')&&(x[3]=='X')&&(x[6]!='O')&&(x[6]!='X'))

c=7;

else if((x[0]=='X')&&(x[6]=='X')&&(x[3]!='O')&&(x[3]!='X'))

c=4;

else if((x[3]=='X')&&(x[6]=='X')&&(x[0]!='O')&&(x[0]!='X'))

c=1;

else if((x[1]=='X')&&(x[4]=='X')&&(x[7]!='O')&&(x[7]!='X'))

c=8;

else if((x[1]=='X')&&(x[7]=='X')&&(x[4]!='O')&&(x[4]!='X'))

c=5;

else if((x[4]=='X')&&(x[7]=='X')&&(x[1]!='O')&&(x[1]!='X'))

c=2;

else if((x[2]=='X')&&(x[5]=='X')&&(x[8]!='O')&&(x[8]!='X'))

c=9;

else if((x[2]=='X')&&(x[8]=='X')&&(x[5]!='O')&&(x[5]!='X'))

c=6;

else if((x[5]=='X')&&(x[8]=='X')&&(x[2]!='O')&&(x[2]!='X'))

c=3;

else

c=(int)(Math.random()\*9);

return c;

}

public static void hangman() throws IOException

{

File dictionary=new File("C:\\Users\\Nexgen\\Desktop\\java\\semesterproject\\engmix.txt");

Scanner textScanner=new Scanner(dictionary);

Scanner input=new Scanner(System.in);

ArrayList<String> words=new ArrayList<>();

while(textScanner.hasNextLine())

{

words.add(textScanner.nextLine());

}

String gameword=words.get((int)(Math.random()\*words.size()));

char[] gamewordarray=gameword.toCharArray();

char[] guess=new char[gamewordarray.length];

for(int i=0;i<gamewordarray.length;i++)

{

guess[i]='\_';

}

int lives=6;

while(true)

{

drawHangman(lives);

int check=0,gamecheck=0;

System.out.println("Lives = "+lives);

System.out.println("Guess a letter :");

for(int k=0;k<gamewordarray.length;k++)

{

System.out.print(guess[k]+" ");

}

System.out.println("");

String letter=input.next();

letter = letter.toLowerCase();

while(letter.length()!=1||Character.isDigit(letter.charAt(0)))

{

System.out.println("Invalid Input!\nEnter again:");

letter=input.next();

}

for(int j=0;j<gamewordarray.length;j++)

{

if(letter.charAt(0)==gamewordarray[j])

{

guess[j]=gamewordarray[j];

check++;

}

}

if(check==0)

{

lives--;

System.out.println("Wrong Guess!");

}

for(int x=0;x<gamewordarray.length;x++)

{

if(guess[x]=='\_')

{

gamecheck++;

}

}

if(gamecheck==0)

{

System.out.println("Congratualations You have won the game!!");

break;

}

if(lives==0)

{

System.out.println("You have lost the game!!");

drawHangman(lives);

break;

}

}

System.out.println(gameword);

}

public static void drawHangman(int l)

{

switch (l) {

case 6:

System.out.println("|----------");

System.out.println("| |");

System.out.println("|");

System.out.println("|");

System.out.println("|");

System.out.println("|");

System.out.println("|");

break;

case 5:

System.out.println("|----------");

System.out.println("| |");

System.out.println("| O");

System.out.println("|");

System.out.println("|");

System.out.println("|");

System.out.println("|");

break;

case 4:

System.out.println("|----------");

System.out.println("| |");

System.out.println("| O");

System.out.println("| |");

System.out.println("|");

System.out.println("|");

System.out.println("|");

break;

case 3:

System.out.println("|----------");

System.out.println("| |");

System.out.println("| O");

System.out.println("| --|");

System.out.println("|");

System.out.println("|");

System.out.println("|");

break;

case 2:

System.out.println("|----------");

System.out.println("| |");

System.out.println("| O");

System.out.println("| -|-");

System.out.println("|");

System.out.println("|");

System.out.println("|");

break;

case 1:

System.out.println("|----------");

System.out.println("| |");

System.out.println("| O");

System.out.println("| --|--");

System.out.println("| /");

System.out.println("| /");

System.out.println("|");

break;

default:

System.out.println("|----------");

System.out.println("| |");

System.out.println("| O");

System.out.println("| --|--");

System.out.println("| / \\");

System.out.println("| / \\");

System.out.println("|");

break;

}

}

public static void mindReader()

{

Scanner input = new Scanner(System.in);

int number;

int number1;

System.out.println("Think of a word and We will guess it");

System.out.println("How many alphabets does it contain? ");

int len=input.nextInt();

int[] arg1=new int[len];

int[] arg2=new int[len];

char alp1 [][] = {{'A','B','C','D','E','F'},{'G','H','I','J','K','L'},

{'M','N','O','P','Q','R'},{'S','T','U','V','W','X'},

{'Y','Z','0','0','0','0'}};

for (int i = 0; i < alp1.length; i++)

{

System.out.print((i+1) + " ");

for (int j = 0; j < alp1[0].length; j++)

{

System.out.print(alp1[i][j] + " ");

}

System.out.println("");

}

for(int a=0;a<len;a++)

{

System.out.println("Enter Row number of Letter "+(a+1)+" of your word:");

arg1[a]=input.nextInt()-1;

}

char alp2 [][] = {{'A','G','M','S','Y'},{'B','H','N','T','Z'},

{'C','I','O','U','0'},{'D','J','P','V','0'},{'E','K','Q','W','0'},

{'F','L','R','X','0'}};

for (int i = 0; i < alp2.length; i++)

{

System.out.print((i+1) + " ");

for (int j = 0; j < alp2[0].length; j++)

{

System.out.print(alp2[i][j] + " ");

}

System.out.println("");

}

for(int a=0;a<len;a++)

{

System.out.println("Enter Row number of Letter "+(a+1)+" of your word:");

arg2[a]=input.nextInt()-1;

}

for (int i = 0; i < arg2.length; i++)

{

number = arg1[i];

number1 = arg2[i];

System.out.print(alp2[number1][number]);

}

}

public static void whatNext()

{

Scanner input= new Scanner(System.in);

int eg=100;

int a=0;

int end=0;

int choice;

while(true)

{

System.out.println("Evilen wakes up in the middle of the night to a heavy bang in the house. "

+ "Everything is pitch black. She cannot see anything."

+ "\nWhat must she do?\n1. Use her phone's flashlight.\n2.Ignore the noise.\n3.Panic and Cry ");

choice=input.nextInt();

while(true)

{

if ((choice==2)||(choice==3))

{

System.out.println("Bad choice. Try again!");

choice=input.nextInt();

}

else if(choice==1)

{

System.out.println("She picks up the torch and slowly walks to the door.");

break;

}

else

{

System.out.print("Invalid Choice. Try again.");

choice=input.nextInt();

}

}

System.out.println("A creaking sound in made as she tries to open the door."

+ " She looks outside to see no one, not a single family member."

+ "\nWhat to do next?\n1. Try to find them.\n2. Call the Police."

+ "\n3.Pick something solid to protect her.\n");

choice=input.nextInt();

while(true)

{

if(choice==2)

{

System.out.println("No Service!\nTry again.\n");

choice=input.nextInt();

}

else if(choice==1)

{

System.out.println("As she slowly tip toes down the stairs, she suddenly sees a bat flying "

+ "towards her through the broken window of the lounge. She has nothing to protect her herself."

+ "She fells down the stairs, hits her head and loses concious. She wakes up to find herself at the bottom of the staircase, her head filled with blood"

+ "and her phone broken. Her head is hurting. She struggles to get and walks towards the kitchen in pain.");

eg=30;

break;

}

else if(choice==3)

{

System.out.println("As she slowly tip toes down the stairs, she suddenly sees a bat flying "

+ "towards her through the broken window of the lounge.She swings the vase she picked up earlier at the bat and it flies away. Her phone battery dies. She slowly walks down and towards the kitchen.");

break;

}

else

{

System.out.println("Invalid choice. Try again.\n");

choice=input.nextInt();

}

}

while(true)

{

if(eg<100)

{

System.out.println("She is low on energy. What would she like to eat?\n1. Grapes.\n2. Cookie.\n3. Water.\n");

choice=input.nextInt();

if(choice==1)

{

eg=80;

System.out.println("Her energy is now "+eg+"%.");

break;

}

else if(choice==2)

{

eg=100;

System.out.println("Her energy is now "+eg+"% because she had sugar.");

break;

}

else if(choice==3)

{

eg=50;

System.out.println("Her energy is now "+eg+"%.");

break;

}

else

{

System.out.println("Invalid choice. Try Again!\n");

}

}

break;

}

System.out.println("It is pitch black,very difficult to see.\nSuddenly she hears someone's footsteps walking towards her.\nWhat should you do?"

+ "\n1. Pick up a knife.\n2. Hide.\n3. Run.\n");

choice=input.nextInt();

while(true)

{

if(choice==2)

{

System.out.println("He finds her, knocks her out and takes her to the basement.");

a=1;

break;

}

else if(choice==3)

{

System.out.println("He finds her, knocks her out and takes her to the basement.");

a=1;

break;

}

else if(choice==1)

{

System.out.println("She hides behind the door and when the footsteps come close to her"

+ ", she tries to defence attack.");

break;

}

else

{

System.out.println("Invalid choice. Try again!\n");

choice=input.nextInt();

}

}

if(a==1)

break;

System.out.println("1. Hits neck.\n2. Hits upper body.\n"

+ "This time Computer will choose either 1 or 2 randomly upon her luck. ");

Random r=new Random();

int num=r.nextInt(2)+1;

System.out.println("\nComputer chose: "+num);

if(num==1)

{

System.out.print("The knifes strikes his neck and the man dies. She runs towards her neighbor's house for safety."

+ "Police is called. The man is nowhere to be found and he has killed her parents. Police finds out"

+ "he had just escaped prison and was a serial killer. They were his new victims.\nThe End");

end=1;

}

if(end==1)

break;

if (num==1)

System.out.print("The knifes strikes his upper body which buys her time to run.");

System.out.print("She is in fear and her energy is getting low.");

eg=eg-50;

if(eg<=0)

{

System.out.println("She runs out of energy and dies.\nGame over.\nYou killed her because of your bad choices.");

end=1;

}

if(end==1)

break;

System.out.print("She runs towards the back door but it is locked. So she runs upstairs but she hears the man slowly following her."

+ "\nWhere to hide?\n1. Closet.\n2. Washroom.\n");

choice=input.nextInt();

while(true)

{

if(choice==2)

{

System.out.println("He finds her, knocks her out and takes her to the basement.");

a=1;

break;

}

else if(choice==1)

{

System.out.println("He doesnt find her so he leaves the room to go look for her."

+ "All the doors and windows are sealed shut so she can escape.\nSo "

+ "she cannot try to break out because he will hear her and find her.");

break;

}

else

{

System.out.println("Invalid choice. Try again!\n");

choice=input.nextInt();

}

}

System.out.println("What does she do now?\n1. Stay there whole night."

+ "\n2. Try going to the Attic.\n");

choice=input.nextInt();

while(true)

{

if(choice==1)

{

System.out.println("He finds her while she sleeps and abducts her and takes her to the basement.");

a=1;

break;

}

else if(choice==2)

{

System.out.println("As she walks in the Attic, she finds out that the window is not locked. She climbs"

+ "out on the roof trying to make no noise and climbs down the tree besides the roof."

+ "She runs towards her neighbor's house for safety. Police is called. The man is no where to be found"

+ "and has killed her parents. Police later finds out he had just escaped prison and is a serial killer. They were"

+ "his next victims.\nThe End!");

end=1;

break;

}

else

{

System.out.println("Invalid choice. Try again!\n");

choice=input.nextInt();

}

}

if(end==1)

break;

if(a==1)

break;

}

if(a==1)

{

System.out.println("She wakes up in the basement and is tied to a chair. Her head hurts. She slowly gains concious only"

+ "to see her parents laying dead in the corner of the basement. She bursts into tears."

+ "\nDid she take self defence class?\n1. Yes\n2. No");

choice=input.nextInt();

while(true)

{

if(choice==1)

{

System.out.print("Wrong Answer because she didnot take them. This was trick question.\nYou Lose because of your poor decision making skills!");

break;

}

else if(choice==2)

{

System.out.println("She looks around to see a screw driver on the ground. She tries to move the chair and crawls somehow"

+ "to reach it. Her energy is now more low. Yet somehow she manages to cut the ropes and rushes to her parents'."

+ " dead bodies. She is filled with tears but its no time to mourne.\n What What does she do now?"

+ "\n1. Run up stairs.\n2. Wait for him to come down.\n ");

choice=input.nextInt();

while(true)

{

if(choice==1)

{

System.out.println("As she is running up the stairs, he comes at the top stair and pushes her and she fells down the stairs,"

+ "snaps her neck and dies.\nGame over.\nYou lose because of your bad choices.");

break;

}

else if(choice==2)

{

System.out.println("She stands besides the stair case waiting for him"

+ "to come down. As soon as he does, she stabs the screw driver in his neck and he bleeds to death."

+ "\nShe runs to her neighbor's house for safety. Police is called. The man has killed her parents. Police later finds out he has just escaped prison and is serial killer."

+ "They were his new victims.\nGame over!");

break;

}

else

System.out.println("Invalid choice. Try again!\n");

choice=input.nextInt();

}

}

else

System.out.println("Invalid choice. Try again!\n");

choice=input.nextInt();

}

}

}

public static void wordCross()

{

Scanner input = new Scanner(System.in);

int count = 0;

int hints[] = {7, 3, 17, 14, 7, 1, 14, 13, 7, 8, 11, 7, 14, 1, 20, 12, 9, 17, 15, 6};

String ans[] = {"egg", "candle", "sponge", "promise", "towel", "bank", "darkness",

"piano", "needle", "seven", "short", "envelope", "silence",

"river", "light", "mirror", "footsteps", "nothing", "queue", "few"};

String newans[] = new String[50];

try

{

FileOutputStream riddles = new FileOutputStream("C:\\Users\\Nexgen\\Desktop\\java\\semesterproject\\Riddles.txt");

PrintStream writer = new PrintStream(riddles);

writer.println("What has to be broken before you use it?");

writer.println("I’m tall when I’m young and I’m short when I’m old. What am I?");

writer.println("What is full of holes but still holds water?");

writer.println("What can break even if you don’t touch it or pick it up? ");

writer.println("What gets wet while drying?");

writer.println("I have branches but no fruit, trunk or leaves? ");

writer.println("The more of this there is, the less you see. What is it?");

writer.println("What has many keys but can’t open a single lock? ");

writer.println("What has one eye, but can’t see?");

writer.println("I am an odd number. Take away a letter and I become even. What number am I? ");

writer.println("What five-letter word becomes shorter when you add two letters to it? ");

writer.println("What begins with an “e” and only contains one letter?");

writer.println("What is so fragile that saying its name breaks it?");

writer.println("What can run but never walks, has a mouth but never talks, has a head but never weeps, has a bed but never sleeps?");

writer.println("What can fill a room but takes up no space? ");

writer.println("If you drop me, I’m sure to crack, but give me a smile and I’ll always smile back. What am I?");

writer.println("The more you take, the more you leave behind. What are they? ");

writer.println("What does man love more than life, hate more than death or mortal strife; that which contented men desire; the poor have," +

" the rich require; the miser spends, the spendthrift saves, and all men carry to their graves? ");

writer.println("What word is pronounced the same if you take away four of its five letters?");

writer.println("I am a word of letters three; add two and fewer there will be. What word am I?");

riddles.close();

}

catch (Exception e)

{

System.out.println("Error Occured");

}

char grid[][] = {{'B', 'P', 'D', 'B', 'W', 'J', 'D', 'R', 'E', 'V', 'I', 'R'},

{'A', 'O', 'E', 'E', 'L', 'I', 'T', 'E', 'Y', 'A', 'R', 'X'},

{'N', 'L', 'N', 'I', 'L', 'C', 'C', 'A', 'N', 'D', 'L', 'E'},

{'K', 'E', 'U', 'D', 'V', 'D', 'X', 'W', 'B', 'Q', 'E', 'G'},

{'T', 'V', 'M', 'G', 'E', 'H', 'E', 'C', 'U', 'G', 'W', 'F'},

{'N', 'N', 'G', 'W', 'K', 'F', 'R', 'E', 'Z', 'H', 'O', 'B'},

{'E', 'E', 'L', 'A', 'F', 'S', 'S', 'A', 'N', 'O', 'T', 'R'},

{'I', 'G', 'N', 'E', 'V', 'E', 'S', 'V', 'Q', 'R', 'E', 'O'},

{'K', 'Y', 'Z', 'L', 'O', 'V', 'E', 'F', 'F', 'S', 'O', 'R'},

{'C', 'S', 'P', 'B', 'C', 'S', 'N', 'O', 'Z', 'H', 'I', 'R'},

{'O', 'G', 'A', 'J', 'E', 'N', 'K', 'O', 'X', 'M', 'S', 'I'},

{'N', 'N', 'J', 'B', 'V', 'P', 'R', 'T', 'A', 'P', 'B', 'M'},

{'K', 'I', 'V', 'Q', 'E', 'P', 'A', 'S', 'O', 'R', 'K', 'P'},

{'N', 'H', 'H', 'T', 'V', 'I', 'D', 'T', 'E', 'P', 'Y', 'S'},

{'O', 'D', 'T', 'L', 'Q', 'A', 'L', 'E', 'A', 'M', 'T', 'I'},

{'C', 'O', 'K', 'J', 'B', 'N', 'I', 'P', 'N', 'R', 'T', 'L'},

{'K', 'N', 'E', 'G', 'N', 'O', 'P', 'S', 'N', 'H', 'Y', 'E'},

{'O', 'L', 'Q', 'K', 'V', 'L', 'E', 'X', 'G', 'R', 'Y', 'N'},

{'W', 'H', 'A', 'P', 'P', 'Y', 'L', 'I', 'A', 'M', 'D', 'C'},

{'T', 'U', 'C', 'A', 'L', 'L', 'L', 'M', 'Z', 'I', 'V', 'E'}};

for (int i = 0; i < grid.length; i++)

{

if (i >= 9)

{

System.out.print((i + 1) + " ");

}

else

{

System.out.print((i + 1) + " ");

}

for (int j = 0; j < grid[0].length; j++)

{

System.out.print(grid[i][j] + " ");

}

System.out.println("");

}

try

{

File file = new File("C:\\Users\\Nexgen\\Desktop\\java\\semesterproject\\Riddles.txt");

Scanner i = new Scanner (file);

int lives = 6;

int choice;

int number = 0;

int answer;

int play;

System.out.println("There are total 20 Riddles.");

while (i.hasNext())

{

Scanner sc=new Scanner(System.in);

if(lives==0)

break;

System.out.println(i.nextLine());

System.out.println("Do you want a hint? (you will lose a life)"

+ "\n1. Yes\n2. No\nlives= " + lives);

choice = input.nextInt();

if (choice == 1)

{

lives--;

System.out.println("First Letter of the Word is in " + hints[number] + " row");

System.out.println("Enter a single word answer in lowercase alphabets: ");

newans[number] = sc.nextLine();

}

else

{

System.out.println("Enter a single word answer in lowercase alphabets: ");

newans[number] = sc.nextLine();

}

System.out.println("Do you want to continue playing?"

+ "\n1. Yes\n2. No ");

play=input.nextInt();

if (ans[number].equalsIgnoreCase(newans[number]))

count++;

if(play==2)

{

break;

}

number++;

}

if (lives == 0)

{

System.out.println("You ran out of lives.\nYou lose!");

}

System.out.println("You got " + count + " out of 20 correct.");

}

catch (Exception e)

{

System.out.println("Error found");

}

}

public static void rps()

{

//rock paper scissor method starts

Scanner input = new Scanner (System.in);

Random r= new Random();

System.out.println("Welcome to \n 'Rock , Paper , Scissors'\n Best out of 5 first to dominate wins!! ");

int counter=0;

int userscore=0;

int compscore=0;

int draw=0;

while(counter<5)

{

System.out.println("Please input the element of your choice! \n 1 for Rock \n 2 for paper \n 3 for scissors ");

int user=input.nextInt();

String choice="";

int comp= r.nextInt(3);

if (user>0 && user<4)

{

//if user enters valid number game loop is entered

if(user==1)

{

//user choice is matched with computer choice and scores are awarded on the basis of rules of the game!!!

if(comp==0)

{ choice="Rock";

System.out.print("Computer choice : "+choice+"\n User choice : Rock "+"\n It is a draw !");

draw+=1;

}

if(comp==1)

{ choice="Paper";

System.out.print("Computer choice : "+choice+"\n User choice : Rock "+"\n Paper beats rock \n You have lost !");

compscore+=1;

}

if(comp==2)

{ choice="Scissor";

System.out.print("Computer choice : "+choice+"\n User choice : Rock "+"\n Rock beats Scissors \n You have won !");

userscore+=1;

}

System.out.print("\n");

System.out.println("--------------------------");

}

if(user==2)

{

if(comp==0)

{ choice="Rock";

System.out.print("Computer choice : "+choice+"\n User choice : Paper "+"\n Paper beats rock \n You have won ! !");

userscore+=1;

}

if(comp==1)

{ choice="Paper";

System.out.print("Computer choice : "+choice+"\n User choice : Paper "+"\n It is a draw !");

draw+=1;

}

if(comp==2)

{ choice="Scissor";

System.out.print("Computer choice : "+choice+"\n User choice : Paper "+"\n Scissors beats Paper \n You have lost !");

compscore+=1;

}

System.out.print("\n");

System.out.println("--------------------------");

}

if(user==3)

{

if(comp==0)

{ choice="Rock";

System.out.print("Computer choice : "+choice+"\n User choice : Scissors "+"\n Rock beats Scissors \n You have lost ! !");

compscore+=1;

}

if(comp==1)

{ choice="Paper";

System.out.print("Computer choice : "+choice+"\n User choice : Scissors "+"\n Scissor beats paper \n You have won !");

userscore+=1;

}

if(comp==2)

{ choice="Scissor";

System.out.print("Computer choice : "+choice+"\n User choice : Scissors "+"\n It is a draw!");

draw+=1;

}

System.out.println("\n");

System.out.println("--------------------------");

}

}

else

{

//if user enters invalid choice then counter is decreased as if to try again

System.out.println("Your input is invalid !!! Please enter a valid number ");

counter-=1;

}

counter+=1;

} //displaying score

System.out.println("user score : "+userscore);

System.out.println("computer score : "+compscore);

System.out.println("drawn games : "+draw);

//checking for who has won

if(userscore>compscore)

{

System.out.println("you have won !!!");

}

else if (userscore==compscore)

{

System.out.println("It is a draw !!!");

}

else

{

System.out.println("You have lost!!!");

}

}

public static void handCricket()

{

Scanner input =new Scanner(System.in);

Random r=new Random();

System.out.println("Welcome to hand cricket ! \n A game of our childhood");

System.out.println("Rules : \n You will play against computer !!");

System.out.println("first is toss enter number from 1 to 5 also choose heads or tails");

System.out.println("if you win you can choose to bat or bowl");

System.out.println("battting or bowling enter a number between 1 to 6");

System.out.println("if number matches you are out if batting or computer is out if bowling");

System.out.println("Welcome to toss !! ");

boolean z=true; //boolean used to check whether user choice is in range or not

int usertoss=0;

while(z=true)

{

System.out.println("Choose heads or tails \n 1 for heads & 2 for tails");

usertoss=input.nextInt();

if (usertoss==1 || usertoss==2)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

int usertoss1=0;

while(z=true)

{

System.out.println("Enter a number from 1 to 5");

usertoss1=input.nextInt();

if (usertoss1>0 && usertoss1<6)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

int comptoss1=r.nextInt(6);

int check=0;

boolean x=true;// used to check score while the game is being played

boolean y=true;// used to check who has won outside game loop

int usernum=0;//user entered number from 1 to 6

int userscore=0;// user score

int compchoice=1;//comp choice from 1 to 6 random numbers are generated from 1 to 6

int compscore=0;//computer score

int tosstotal=usertoss1+comptoss1;// both user and comp choice are added !!

if(usertoss==1)

{

//nested loop to check who has won the toss

//iteration of check is used to decide which loop to enter after decision id declared of toss

//if comp wins check is iterated +3 and if player wins it is iterated +1

if(tosstotal%2==0)

{

System.out.println("User choice : "+usertoss+"\nUser input : "+usertoss1);

System.out.println("Computer input : "+comptoss1);

System.out.println("It is tails !! you have lost the toss ");

check+=3;

}

else

{

System.out.println("User choice : "+usertoss+"\nUser input : "+usertoss1);

System.out.println("Computer input : "+comptoss1);

System.out.println("It is heads !! you have won the toss ");

check+=1;

}

}

else if(usertoss==2)

{

if(tosstotal%2==0)

{

System.out.println("User choice : "+usertoss+"\nUser input : "+usertoss1);

System.out.println("Computer input : "+comptoss1);

System.out.println("It is tails !! you have won the toss ");

check+=1;

}

else

{

System.out.println("User choice : "+usertoss+"\nUser input : "+usertoss1);

System.out.println("Computer input : "+comptoss1);

System.out.println("It is heads !! you have lost the toss ");

check+=3;

}

}

else

{

System.out.println("Invalid Input");

}

int userchoice=0;

if(check<2)

{

//user has won the toss loop starts

//in this condition user has won the toss

while(z==true)

{

//loop for decision of batting or bowling

System.out.println("you have won the toss what do you want to do? \n enter 1 for batting \n 2 for balling? ");

userchoice=input.nextInt();

if (userchoice==1 || userchoice==2)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

if(userchoice==1)

{

//user decides to bat and game starts

System.out.println("You have decided to bat");

while(x=true)

{

while(z=true)

{

// check if user has entered right choice

System.out.println("Enter your number you want to score between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

System.out.println("user score is "+userscore);

x=false;

break;

//user is out

}

else

{

//user score iterated by user choice if user is not out

userscore+=usernum;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nUser score :: "+userscore +"\n");

}

}

System.out.println("user total score is "+userscore);

System.out.println("2nd innings"+"computer needs "+userscore+" to win");

while(x=true)

{

//2nd innings start and is similiar to above loop

// the only difference is that here comp score is iterated

//because here comp is chasing the target

while(z=true)

{

System.out.println("Enter your number you want to bowl between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

x=false;

break;

}

else

{

compscore+=compchoice;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nComputer score :: "+compscore +"\n");

}

if(userscore==compscore || compscore>userscore)

{

x=false;

System.out.print("You have lost the match!!! !!!");

break;

}

}

System.out.println("comp total score is :: "+compscore);

while(y=true)

{

if(compscore==userscore|| compscore>userscore)

{

y=false;

break;

}

else

{

System.out.print("Hurray ! You have won the game !!!");

y=false;

break;

}

}

}

else if(userchoice==2)

{

System.out.println("You have decided to bowl first!! ");

while(x=true)

{

while(z=true)

{

System.out.println("Enter your number you want to bowl between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

x=false;

break;

}

else

{

compscore+=compchoice;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nComputer score :: "+compscore +"\n");

}

}

System.out.println("comp total score is "+compscore);

System.out.println("2nd innings"+" you need "+compscore+" to win");

while(x=true)

{

while(z=true)

{

System.out.println("Enter your number you want to score between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

System.out.println("user score is "+userscore);

x=false;

break;

}

else

{

userscore+=usernum;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nUser score :: "+userscore +"\n");

}

if(userscore==compscore || userscore>compscore)

{

x=false;

System.out.print("You have won the match!!! !!!");

break;

}

}

System.out.println("user total score is"+userscore);

while(y=true)

{

if(compscore==userscore|| userscore>compscore)

{

y=false;

break;

}

else

{

System.out.print(" You have lost the game !!!");

y=false;

break;

}

}

}

}//user has won the toss loop ends

else if(check>2)

{

//computer has won the toss loop starts

System.out.println("Computer has won the toss !!! ");

int compchoicetoss=r.nextInt(2);

if(compchoicetoss==0)

{

System.out.print("Computer has decided to bat !! ");

while(x=true)

{

while(z=true)

{

System.out.println("Enter your number you want to bowl between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

System.out.println("comp score is"+compscore);

x=false;

break;

}

else

{ compscore+=compchoice;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nComputer score :: "+compscore +"\n");

}

}

System.out.println("2nd innings "+"you need "+compscore+" to win");

while(x=true)

{

while(z=true)

{

System.out.println("Enter your number you want to score between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

System.out.println("user score is "+userscore);

x=false;

break;

}

else

{

userscore+=usernum;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nUser score :: "+userscore +"\n");

}

if(userscore==compscore || userscore>compscore)

{

x=false;

System.out.print("You have won the match!!! !!!");

break;

}

}

while(y=true)

{

if(compscore==userscore|| userscore>compscore)

{

y=false;

break;

}

else

{

System.out.print(" You have lost the game !!!");

y=false;

break;

}

}

}

else if(compchoicetoss==1)

{

System.out.println("computer has decided to bowl");

while(x=true)

{

while(z=true)

{

System.out.println("Enter your number you want to score between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

System.out.println("user score is "+userscore);

x=false;

break;

}

else

{

userscore+=usernum;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nUser score :: "+userscore +"\n");

}

}

System.out.println("user total score is"+userscore);

System.out.println("2nd innings"+"computer needs "+userscore+" to win");

while(x=true)

{

while(z=true)

{

System.out.println("Enter your number you want to bowl between 1-6 !");

usernum=input.nextInt();

if (usernum>0 && usernum<7)

{

z=false;

break;

}

else

{

System.out.println("Your choice is invalid !!! Try again !!");

}

}

compchoice=r.nextInt(6)+1;

if(usernum==compchoice)

{

x=false;

break;

}

else

{

compscore+=compchoice;

System.out.print("Computer choice :: "+compchoice+"\n"+"User choice :: "+usernum+"\nComputer score :: "+compscore +"\n");

}

if(userscore==compscore || compscore>userscore)

{

x=false;

System.out.print("You have lost the match!!! !!!");

break;

}

}

System.out.println("comp total score is"+compscore);

while(y=true)

{

if(compscore==userscore|| compscore>userscore)

{

y=false;

break;

}

else

{

System.out.print(" You have won the game !!!");

y=false;

break;

}

}

}

else

{

System.out.println("Invalid input !!! ");

} // this else is to complete if else loop

} // here computer wins toss loop ends

} // here hand cricket method ends

}