MATHEFFECTS

A project report submitted in partial fulfillment of the requirements for the degree

Of

Bachelor of Computer Applications (BCA)

OF

North Eastern Hill University (NEHU)

Submitted by

Name: CHIHANSHANG KASAR

Roll No.: P1500066

Reg. No.: 14523of 2013-14

DEPARTMENT OF COMPUTER SCIENCE & APPLICATIONS

SHILLONG COLLEGE

SHILLONG 793003

MEGHALAYA.

April, 2017

CERTIFICATE

It is certified that the work contained in the project report titled ***MATHEFFECTS*** is a bonafide work done by ***Chihanshang Kasar,*** University Roll No.: ***P1500066,*** Registration No.:14523 of ***2014-14,*** has been carried out under my supervision during the final year of ***Bachelor of Computer Applications.***

IGKharmawphlang

Supervisor Miss Ibameda G.Kharmawphlang

Dept. Of Comp. Sci.&App.

Shillong College

Project seminar was held on ....17/04/2017.... at Shillong College, Shillong.

17/4/17

Smt. Aiom Mitri

HOD

Dept. Of Comp. Sci.&App.

Shillong College

Shillong.

External examiner

17/4/17

Contents

Acknowledgement

The project work may seems like a great solo effort, but the contribution towards it is what really a team work. This project would not have been possible without the exerted help of all my teachers and friends.

Specially thanks go to Miss Ibameda G. Kharmawphlang,my project guide, who has always been there to help me with my project and whatever is in need.

I also extend my gratitude and thanks to all the teacher of computer science department for their guidance, encouragement and their assistance in completing this project. And finally I like to extend a great thankfulness to my dear friends for giving me their time and support in developing this project.

2. Introduction:

A game is structure form of play, usually undertaken for enjoyment and sometimes used as an educational tool.

Game are usually carried out for remuneration and from art which is more often an expression of aesthetic or ideological elements

3. Objective:

This apps will used skill of mathematic .User are to be a part of the game the game will be extremely fun. The game will have two deferent level depending on the user skill.This app will help the user to improve and learn their skill in mathematics and also same time gives an immense entertainment.

This game is specially design for the mathematic beginner to quickly improve their skill in mathematic.

4. Technology to be used:

*Netbean IDE 8.2

**java development kit(JDK) 8*

*Java Programing Language

5. System required:

*Windows 7 and above

*RAM 1GB

*Intel Pentium above

*32 and 64 Bits processor

*Java Runtime Environment Compatible

6. Steps to play the game:

1.start the game.

2. select game mode(time/test)

3. play(guess the answer)

4.stop

7. How the game works:

1. The game will have two modes (a)Time(b)Test.
2. (a)Time = easy level (b) Test = hard level.
3. The game will have different arithmetic operation (sum,substraction.multiplication,divesion).
4. There are two random numbers generated and displayed with a random operant and a result(true/false)so the player has to guess the answer
5. The game will continue as long as the result is right.
6. The game will be over if six of the answers are wrong.
7. The game will restart again until the game is exit.

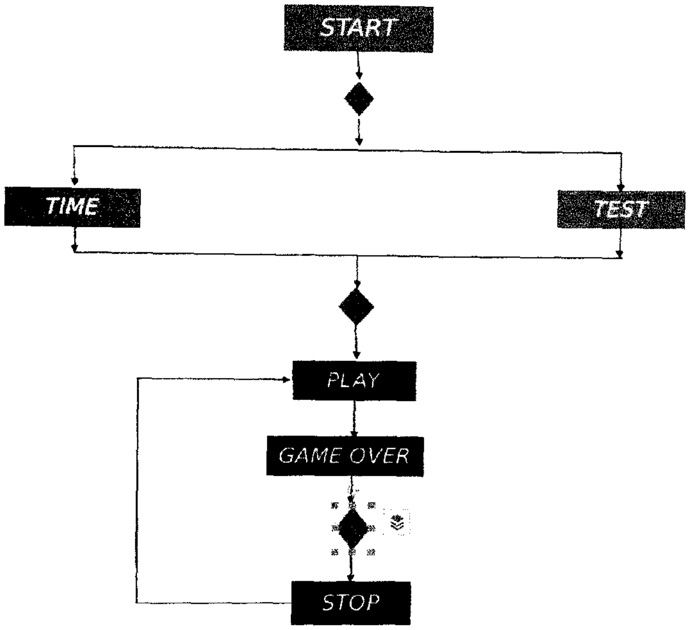
8. System development:

The game is developed in Net beans ID 8 using Java Programing Language

-applet Library

-swing Library

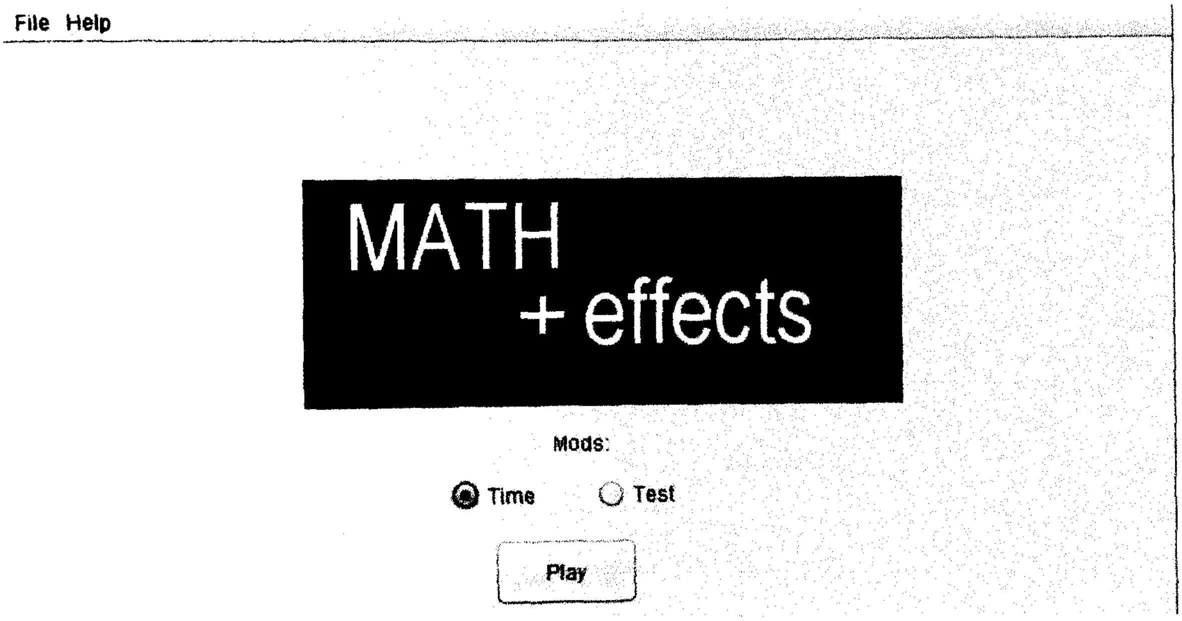
9. Game flowchart:



10. Graphical User Interface (GUI):

(i).Game start interface:

Select game mode time or test

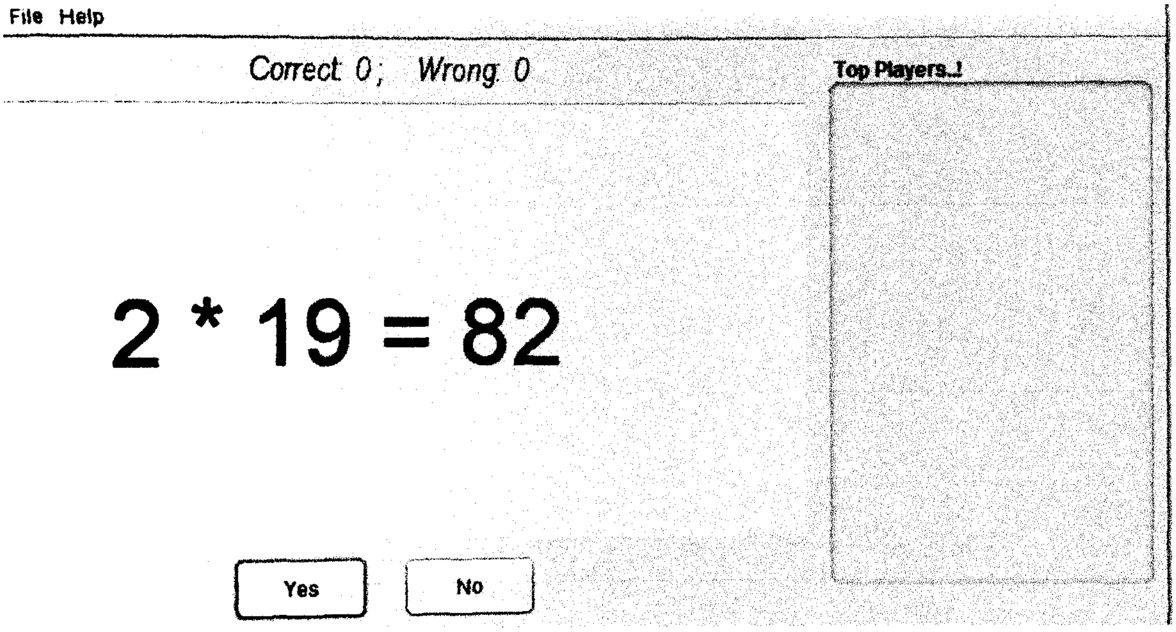


Mods:

● Time ○ Test

(II). Game _on play interface:

Guess the answer true or false

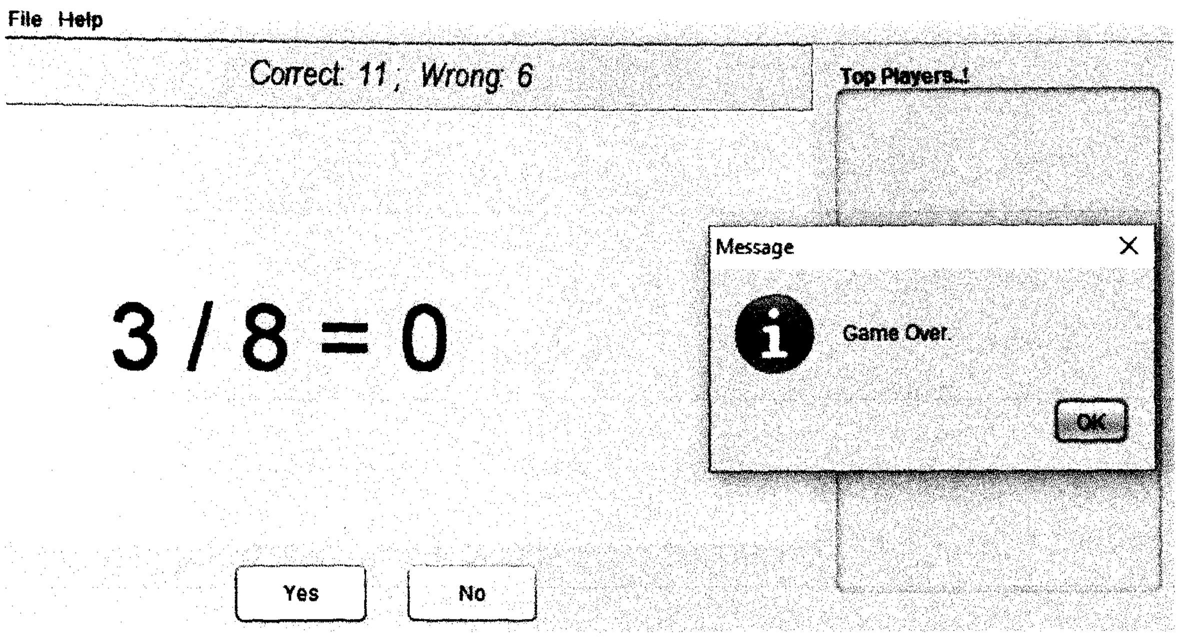


Correct 0 ; Wrong 0

2 * 19 = 82

(III). Game over interface:

Showing game over after six times wrong



Correct: 11 ; Wrong: 6

3 / 8 = 0

Game Over.

11. Algorithm:

1. Start the game
2. Select mode(Time or Test)

3. Play(guess the answer yes or no)

4. Game over

5. Repeat step 3.

12. Source Code:

(I).GameField :

```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.RenderingHints;

import javax.swing.JPanel;

public class GameField extends JPanel {

  private int[] getNumbers;

  private boolean checkk;

  public int[] getGetNumbers() {

    // getNumbers = (new GameModel).getFinalAnswer();
    return getNumbers;
  }

  public GameField() {

    setLayout(new BorderLayout());
    setVisible(true);
  }

  public void paintComponent(Graphics g) {
    super.paintComponent(g);
    Graphics2D g2 = (Graphics2D) g;

    GameModel model = new GameModel();

    int[] numbers = model.getFinalAnswer();
    String operand = model.getOperand();
    getNumbers = numbers;

    checkk = model.checkAnswer();

    int width = getWidth();
    int height = getHeight();
    Font font = new Font("AR JULIAN", Font.TRUETYPE_FONT, 55);

    g2.setFont(font);
    g2.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
        RenderingHints.VALUE_ANTIALIAS_ON);
    g.setFont(font);

    g.setColor(new Color(230, 225, 230));
    g.fillRect(0, 0, width, height);

    g.setColor(Color.RED);
    g.drawString(numbers[0] + " " + operand + " " + numbers[1] + " = "
        + numbers[2], width / 7, 3 * height / 5);

  }

  public boolean getCheck() {
    System.out.println(checkk);
    return checkk;
  }
}
```

(II)GameModel :

```
import java.util.Random;
```

```
public class GameModel {

  private static Random ran;
  public int randomNumb1;
  public int randomNumb2;
  private String operand = " ";
  private static int[] threeNumbers;
  private static int[] finalAnswer;
  public static boolean check;
  public boolean lastCheck;

  public GameModel() {
    solutionChances();
    solution();
  }

  public String getOperand() {
    return operand;
  }

  public int[] ranNumbers() {

    int count = MainFrame.getCount();
    int[] numbers = new int[2];
    if (count < 10) {
      ran = new Random();
      numbers[0] = ran.nextInt(20);
      ran = new Random();
      numbers[1] = ran.nextInt(20);

    } else if (count < 30) {
      ran = new Random();
      numbers[0] = ran.nextInt(50);
      ran = new Random();
      numbers[1] = ran.nextInt(50);

    } else if (count < 50) {
      ran = new Random();
      numbers[0] = ran.nextInt(70);
      ran = new Random();
      numbers[1] = ran.nextInt(70);

    } else {
      ran = new Random();
    numbers[0] = ran.nextInt(100);
    ran = new Random();
    numbers[1] = ran.nextInt(100);
  }
  return numbers;
}

public static String operatori() {

  ran = new Random();
  int stringRan = ran.nextInt(4);

  String[] operatoret = { "+", "-", "*", "/" };
  String answer = "";

  switch (stringRan) {
  case 0:
    answer = operatoret[0];
    break;
  case 1:
    answer = operatoret[1];
    break;
  case 2:
    answer = operatoret[2];
    break;
  case 3:
    answer = operatoret[3];
    break;
  }

  return answer;
}

//answer[0]
//answer[2]

public int[] solutionChances() {
  ran = new Random();

  finalAnswer = new int[3];

  int ranNumb = ran.nextInt(2) + 1;

  int[] answer = new int[3];
```

```
int[] ranNumbers = new int[3];
ranNumbers = ranNumbers();
operand = operatori();
switch (ranNumb) {
// zgjidhja reale dmth true
case 1:
  switch (operand) {
  case "+":
    answer[2] = ranNumbers[0] + ranNumbers[1];
    operand = "+";
    break;
  case "-":
    answer[2] = ranNumbers[0] - ranNumbers[1];
    operand = "-";
    break;
  case "*":
    answer[2] = (ranNumbers[0]) * ranNumbers[1];
    operand = "*";
    break;
  case "/":
    if (ranNumbers[1] == 0) {
      ranNumbers[1] += 1;

      answer[2] = ranNumbers[0] / ranNumbers[1];
    } else
      answer[2] = ranNumbers[0] / ranNumbers[1];
    operand = "/";
    break;
  }

  break;

//false
case 2:
  switch (operand) {
  case "+":
    answer[2] = (ranNumbers[0]) + ranNumbers[1] + ran.nextInt(50);
    operand = "+";
    break;
  case "-":
    answer[2] = ranNumbers[0] - ranNumbers[1] + ran.nextInt(50);
    operand = "-";
    break;
  case "*":
        answer[2] = ranNumbers[0] * ranNumbers[1] + ran.nextInt(50);
        operand = "*";
        break;
      case "/":
        if (ranNumbers[1] == 0)
{
                  ranNumbers[1] += 1;
          answer[2] = ranNumbers[0] / ranNumbers[1];
        } else {
          answer[2] = ranNumbers[0] / ranNumbers[1];
        }

        operand = "/";
        break;
      }
      break;
    }
    answer[0] = ranNumbers[0];
    answer[1] = ranNumbers[1];

    finalAnswer = answer;

    return answer;
}

public int[] getFinalAnswer() {

  return finalAnswer;
}

public boolean solution() {

  int[] solutionsNumbers = new int[3];
  int[] numbers = getFinalAnswer();

  threeNumbers = new int[3];
  threeNumbers = numbers;
  solutionsNumbers = threeNumbers;

  int numb1 = solutionsNumbers[0];
  int numb2 = solutionsNumbers[1];
  int solution = solutionsNumbers[2];
```

```
      boolean answer = false;

      switch (operand) {
     case "+":
        answer = (numb1 + numb2) == solution;
        break;
      case "-":
        answer = (numb1 - numb2) == solution;
        break;
      case "*":
        answer = (numb1 * numb2) == solution;
        break;
      case "/":
        answer = (numb1 / numb2) == solution;
        break;
      }

      check = answer;
      return answer;
    }

    public static boolean checkAnswer() {
      return check;
    }


}
```

(III)LeaderLabel :

```
import java.awt.Dimension;

import javax.swing.BorderFactory;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.border.Border;

public class LeaderLabel extends JPanel {
  private JLabel label;

  public LeaderLabel() {

    Dimension dim = getPreferredSize();


    int width = 350;

    dim.width = width;
    setPreferredSize(dim);

    setBorder(BorderFactory.createEtchedBorder());

    Border innerBorder = BorderFactory.createTitledBorder("Top Players..!");
    Border outerBorder = BorderFactory.createEmptyBorder(10, 10, 10, 10);

    setBorder(BorderFactory.createCompoundBorder(outerBorder, innerBorder));
  }
}
```

(IV)MainFrame :

```
import java.awt.Component;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;

public class MainFrame extends JFrame {

  private LeaderLabel leader;
  private GameField gameField;
  private Score score;
  private boolean check;
  private static int count;
  String name = null;
  public static String name1;

  public MainFrame() {
    super("Math Effects");

    setSize(700, 400);
    setVisible(true);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setResizable(false);

    leader = new LeaderLabel();
    gameField = new GameField();

    score = new Score();

    setLayout(null);

   JButton Yes = new JButton("Yes");

    Yes.addActionListener(new ActionListener() {

      public void actionPerformed(ActionEvent e) {
        check = GameModel.checkAnswer();
        if (check) {
          score.incCorrect();
        score.repaint();
        gameField.repaint();
        count++;

     } else {
        score.incWrong();
        score.repaint();
        gameField.repaint();

       count++;
     }
     if (Score.getWrong() > 5) {
      JOptionPane.showMessageDialog(null, "Game Over");
       score.reset();
       score.repaint();
       count = 0;
       gameField.repaint();
     }

  }

});

JButton No = new JButton("No");

No.addActionListener(new ActionListener() {

  public void actionPerformed(ActionEvent e) {

    check = GameModel.checkAnswer();
    if (check) {
      score.incWrong();
      score.repaint();
     gameField.repaint();
      count++;
    } else {
      score.incCorrect();
      score.repaint();
      gameField.repaint();

      count++;
    }

    if (Score.getWrong() > 5) {
        JOptionPane.showMessageDialog(null, "Game Over.");
        score.reset();
        count = 0;
        gameField.repaint();
      }
    }

  });

  Yes.setBounds(140, 305, 80, 37);
  add(Yes);
  No.setBounds(240, 305, 80, 37);
  add(No);

  leader.setBounds(480, 0, 220, 340);
  add(leader);

  score.setBounds(0, 0, 482, 40);
  add(score);

  gameField.setBounds(0, 40, 480, 255);
  add(gameField);

  JMenuBar mb = new JMenuBar();
  setJMenuBar(mb);
  JMenu file = new JMenu("File");
  mb.add(file);
  JMenuItem exit = new JMenuItem("Exit");
  file.add(exit);

  exit.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
      System.exit(0);
    }

  });

  JMenu help = new JMenu("Help");
  mb.add(help);

  JMenuItem howToPlay = new JMenuItem("How To Play");
  help.add(howToPlay);

  final String howTo = "Two random numbers are generated together with the
```

```
operand(random).\n"
        + "Then some answer is generated. If its true and you press Yes, you have +1
Corrected,\n and if its true and you press No, you have +1 Wronged.\n"
        + "\n"
        + "Warning: All calculations are done with integers, so divide operator('/') is done
with whole down part.\n"
        + "For Example if we have:  5/7, the result will be 0, because 5/7 is less than 1 and
when we take the whole down part the result would be 0.\n"
        + "Another example: 10/3 = 3,  15/4 = 3, 20/3 = 6 etc.";

    howToPlay.addActionListener(new ActionListener() {
      public void actionPerformed(ActionEvent arg0) {
        JOptionPane.showMessageDialog(null, howTo);
      }

    });

    JMenuItem about = new JMenuItem("About");
    help.add(about);

    about.addActionListener(new ActionListener() {
      public void actionPerformed(ActionEvent arg0) {
        JOptionPane
            .showMessageDialog(
                null,
                "- Math +effects -\nME V1.0\n"
                    + " by chihan.\n To be continued...!");
      }

    });
  }
  public static int getCount() {
    return count;
  }
  public static void main(String[] args) {
    new MainFrame();
  }

}
```

(V)Score :

```
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.RenderingHints;

import javax.swing.BorderFactory;
import javax.swing.JLabel;
import javax.swing.JPanel;

public class Score extends JPanel {

  private int correct;
  private static int wrong;

  public void paintComponent(Graphics g) {
    super.paintComponent(g);
    Graphics2D g2 = (Graphics2D) g;
    setBorder(BorderFactory.createEtchedBorder());

    Dimension dim = getPreferredSize();

    int height = 25;

    dim.height = height;
    setPreferredSize(dim);

    g.setColor(Color.black);

    Font font = new Font("Arial Narrow", Font.ITALIC, 20);

    g2.setFont(font);
    g2.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
        RenderingHints.VALUE_ANTIALIAS_ON);

    g.drawString("Wrong: " + wrong, 250, 25);
    g.drawString("Correct: " + correct + " ;     ", 150, 25);

  }

  public void reset(){
    correct = 0;
    wrong = 0;
    repaint();
  }

  public void incCorrect() {
    correct++;
  }

  public void incWrong() {
    wrong++;
  }

  public static int getWrong() {
    return wrong;
  }


}
```

(VI)StartFrame :

```
import java.awt.CardLayout;
import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.RenderingHints;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.ButtonGroup;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JRadioButton;

public class StartFrame extends JPanel {
  private JRadioButton modTime;
  private JRadioButton testMod;
  private ButtonGroup genderGroup;

  public StartFrame() {

    modTime = new JRadioButton("Time");
    testMod = new JRadioButton("Test");
    genderGroup = new ButtonGroup();

    JButton play = new JButton("Play");

    play.setBounds(285, 290, 85, 40);
    play.addActionListener(new ActionListener() {
      public void actionPerformed(ActionEvent e) {
        new MainFrame();
      }
    });

    add(play);

    // set up gender ratios
    genderGroup.add(modTime);
    genderGroup.add(testMod);
    modTime.setSelected(true);

    setLayout(null);

    // ModTime radio button
    // first mod

    modTime.setBounds(260, 240, 50, 50);
    add(modTime);

    // second mod
    testMod.setBounds(345, 240, 50, 50);
    add(testMod);
  }

  public void paintComponent(Graphics g) {
    super.paintComponent(g);
   Graphics2D g2 = (Graphics2D) g;

   int width = 700;
    int height = 400;

    g.setColor((new Color(230, 235, 240)));
    g.fillRect(0, 0, width, height);
    g.setColor(Color.RED);
    g.fillRect(width / 4, height / 5, width / 2, height / 3);

    g.setColor(Color.black);
    g.drawString("Mods:", width / 2 - 30, 3 * height / 5);

    g.setColor(Color.white);

    Font font = new Font("Arial Narrow", Font.TRUETYPE_FONT, 55);

    g2.setFont(font);
    g2.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
        RenderingHints.VALUE_ANTIALIAS_ON);

   g.drawString("MATH", 2 * width / 7, 4 * height / 12);
   g.drawString("+ effects", 3 * width / 7, 4 * height / 9);
 }

 public static void main(String[] args) {
    StartFrame panel = new StartFrame();

    JFrame frame = new JFrame("Math Effects");

    JMenuBar mb = new JMenuBar();
    frame.setJMenuBar(mb);
    JMenu file = new JMenu("File");
    mb.add(file);
    JMenuItem exit = new JMenuItem("Exit");
    file.add(exit);

    exit.addActionListener(new ActionListener() {
      public void actionPerformed(ActionEvent arg0) {
        System.exit(0);
      }
    });

    JMenu help = new JMenu("Help");
    mb.add(help);
    JMenuItem about = new JMenuItem("About");
    help.add(about);

    about.addActionListener(new ActionListener() {
      public void actionPerformed(ActionEvent arg0) {
        JOptionPane
            .showMessageDialog(
                null,
                "- Math +effects -\nME V1.0\nby chihan.\n"
                    + " To be continued...!");
      }
    });
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.add(panel);
    frame.setSize(700, 400);
    frame.setVisible(true);

                }

}
```

13. Conclusion:

This game is specially design for the student. This game will help in improving mathematic skill and the game will be easy to play and learn.