**Source Code**

**ComputerSimulatorGroup9.java:-**

/\*

\* Project Team #9

\* Authors: Aditi,Shejal,Sree Vital,Lijadis

\* Project Title: CISC Computer ComputerSimulator

\*/

import javax.swing.\*;

import javax.swing.filechooser.FileSystemView;

import java.awt.\*;

import java.io.\*;

import java.util.logging.Level;

import java.util.logging.Logger;

public class ComputerSimulatorGroup9 extends JFrame {

private short R0; //Stores GPR0 value

private short R1; //Stores GPR1 value

private short R2; //Stores GPR2 value

private short R3; //Stores GPR3 value

private short FR0; //Stores FR0 value

private short FR1; //Stores FR1 value

private short X1; //Stores IXR1 value

private short X2; //Stores IXR2 value

private short X3; //Stores IXR3 value

private short PC; //Stores PC value

private short MAR; //Stores MAR value

private short MBR; //Stores MBR value

private short IR; //Stores IR value

private short MFR; //Stores MFR value

private short CC0; //Stores CC0-OverFlow value

private short CC1; //Stores CC1-UnderFlow value

private short CC2; //Stores CC2-DivZero value

private short CC3; //Stores CC3-EqualOrNot value

public short InputSignal; //Stores Input Signal status

public short InputVal; //Stores Input Value

public ComputerSimulatorGroup9() {

/\*\*

\* To initialize the form, this method is invoked from within the constructor.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

initComponents();

R0 = 0;

R1 = 0;

R2 = 0;

R3 = 0;

FR0 = 0;

FR1 = 0;

X1 = 0;

X2 = 0;

X3 = 0;

PC = 0;

MAR = 0;

MBR = 0;

IR = 0;

MFR = 0;

CC0 = 0;

CC1 = 0;

CC2 = 0;

CC3 = 0;

InputSignal = 0;

InputVal = 0;

}

public static short[] memory = new short[2048];

static JFrame f;

static JButton Store, Load, Clear\_All, SS, Run, Init, Custom\_Init;

static GridLayout G\_Layout, G\_Layout1, G\_Layout2, G\_Layout3;

static JLabel JLabel\_Halt, JLabel\_Run, JLabel\_R0, JLabel\_R1, JLabel\_R3, JLabel\_R2;

static JPanel JPanel\_Func, JPanel\_OPT, JPanel\_Init, Halt\_Display, JPanel\_Run, JPanel\_Display, JPanel\_Grid1, JPanel\_Grid2;

static JButton Load\_R1;

static JPanel JPanelMain, JPanel\_R1;

static JLabel JLabel\_IR;

static JPanel JPanel\_IR;

static JButton Load\_IR;

static JButton IR\_0;

static JButton IR\_1;

static JButton IR\_10;

static JButton IR\_11;

static JButton IR\_12;

static JButton IR\_13;

static JButton IR\_14;

static JButton IR\_15;

static JButton IR\_2;

static JButton IR\_3;

static JButton IR\_4;

static JButton IR\_5;

static JButton IR\_6;

static JButton IR\_7;

static JButton IR\_8;

static JButton IR\_9;

static JPanel JPanel\_R0;

static JButton Load\_R0;

static JButton R0\_0;

static JButton R0\_1;

static JButton R0\_10;

static JButton R0\_11;

static JButton R0\_12;

static JButton R0\_13;

static JButton R0\_14;

static JButton R0\_15;

static JButton R0\_2;

static JButton R0\_3;

static JButton R0\_4;

static JButton R0\_5;

static JButton R0\_6;

static JButton R0\_7;

static JButton R0\_8;

static JButton R0\_9;

static JButton R1\_0;

static JButton R1\_1;

static JButton R1\_10;

static JButton R1\_11;

static JButton R1\_12;

static JButton R1\_13;

static JButton R1\_14;

static JButton R1\_15;

static JButton R1\_2;

static JButton R1\_3;

static JButton R1\_4;

static JButton R1\_5;

static JButton R1\_6;

static JButton R1\_7;

static JButton R1\_8;

static JButton R1\_9;

static JPanel JPanel\_R2;

static JButton Load\_R2;

static JButton R2\_0;

static JButton R2\_1;

static JButton R2\_10;

static JButton R2\_11;

static JButton R2\_12;

static JButton R2\_13;

static JButton R2\_14;

static JButton R2\_15;

static JButton R2\_2;

static JButton R2\_3;

static JButton R2\_4;

static JButton R2\_5;

static JButton R2\_6;

static JButton R2\_7;

static JButton R2\_8;

static JButton R2\_9;

static JPanel JPanel\_R3;

static JButton Load\_R3;

static JButton R3\_0;

static JButton R3\_1;

static JButton R3\_10;

static JButton R3\_11;

static JButton R3\_12;

static JButton R3\_13;

static JButton R3\_14;

static JButton R3\_15;

static JButton R3\_2;

static JButton R3\_3;

static JButton R3\_4;

static JButton R3\_5;

static JButton R3\_6;

static JButton R3\_7;

static JButton R3\_8;

static JButton R3\_9;

static JLabel JLabel\_PC;

static JPanel JPanel\_PC;

static JButton Load\_PC;

static JButton PC\_0;

static JButton PC\_1;

static JButton PC\_10;

static JButton PC\_11;

static JButton PC\_2;

static JButton PC\_3;

static JButton PC\_4;

static JButton PC\_5;

static JButton PC\_6;

static JButton PC\_7;

static JButton PC\_8;

static JButton PC\_9;

static JLabel JLabel\_MAR;

static JPanel JPanel\_MAR;

static JButton Load\_MAR;

static JButton MAR\_0;

static JButton MAR\_1;

static JButton MAR\_10;

static JButton MAR\_11;

static JButton MAR\_2;

static JButton MAR\_3;

static JButton MAR\_4;

static JButton MAR\_5;

static JButton MAR\_6;

static JButton MAR\_7;

static JButton MAR\_8;

static JButton MAR\_9;

static JPanel JPanel\_MBR;

static JLabel JLabel\_MBR;

static JButton Load\_MBR;

static JButton MBR\_0;

static JButton MBR\_1;

static JButton MBR\_10;

static JButton MBR\_11;

static JButton MBR\_12;

static JButton MBR\_13;

static JButton MBR\_14;

static JButton MBR\_15;

static JButton MBR\_2;

static JButton MBR\_3;

static JButton MBR\_4;

static JButton MBR\_5;

static JButton MBR\_6;

static JButton MBR\_7;

static JButton MBR\_8;

static JButton MBR\_9;

static JToggleButton OP\_10;

static JToggleButton OP\_11;

static JToggleButton OP\_12;

static JToggleButton OP\_13;

static JToggleButton OP\_14;

static JToggleButton OP\_15;

static JToggleButton GPR\_8;

static JToggleButton GPR\_9;

static JToggleButton IXR\_6;

static JToggleButton IXR\_7;

static JToggleButton I\_5;

static JToggleButton ADDRESS\_0;

static JToggleButton ADDRESS\_1;

static JToggleButton ADDRESS\_2;

static JToggleButton ADDRESS\_3;

static JToggleButton ADDRESS\_4;

static JPanel JPanel\_OP;

static JPanel JPanel\_IXR;

static JPanel JPanel\_GPR;

static JPanel JPanel\_I;

static JPanel JPanel\_ADDRESS;

static JPanel JPanel\_INSTRUCTION;

static JPanel JPanel\_OP\_Main;

static JPanel JPanel\_GPR\_Main;

static JPanel JPanel\_IXR\_Main;

static JPanel JPanel\_I\_Main;

static JPanel JPanel\_ADDRESS\_Main;

static JPanel JPanel\_X1;

static JLabel JLabel\_X1;

static JButton Load\_X1;

static JButton X1\_0;

static JButton X1\_1;

static JButton X1\_10;

static JButton X1\_11;

static JButton X1\_12;

static JButton X1\_13;

static JButton X1\_14;

static JButton X1\_15;

static JButton X1\_2;

static JButton X1\_3;

static JButton X1\_4;

static JButton X1\_5;

static JButton X1\_6;

static JButton X1\_7;

static JButton X1\_8;

static JButton X1\_9;

static JPanel JPanel\_X2;

static JLabel JLabel\_X2;

static JButton Load\_X2;

static JButton X2\_0;

static JButton X2\_1;

static JButton X2\_10;

static JButton X2\_11;

static JButton X2\_12;

static JButton X2\_13;

static JButton X2\_14;

static JButton X2\_15;

static JButton X2\_2;

static JButton X2\_3;

static JButton X2\_4;

static JButton X2\_5;

static JButton X2\_6;

static JButton X2\_7;

static JButton X2\_8;

static JButton X2\_9;

static JPanel JPanel\_X3;

static JLabel JLabel\_X3;

static JButton Load\_X3;

static JButton X3\_0;

static JButton X3\_1;

static JButton X3\_10;

static JButton X3\_11;

static JButton X3\_12;

static JButton X3\_13;

static JButton X3\_14;

static JButton X3\_15;

static JButton X3\_2;

static JButton X3\_3;

static JButton X3\_4;

static JButton X3\_5;

static JButton X3\_6;

static JButton X3\_7;

static JButton X3\_8;

static JButton X3\_9;

static JPanel JPanel\_MFR;

static JButton MFR\_3;

static JButton MFR\_2;

static JButton MFR\_1;

static JButton MFR\_0;

static JLabel JLabel\_MFR;

static JPanel JPanel\_CC;

static JButton CC\_3;

static JButton CC\_2;

static JButton CC\_1;

static JButton CC\_0;

static JLabel JLabel\_CC;

private void Clear() { //Clear LEDs

PC\_0.setText("0");

PC\_1.setText("0");

PC\_2.setText("0");

PC\_3.setText("0");

PC\_4.setText("0");

PC\_5.setText("0");

PC\_6.setText("0");

PC\_7.setText("0");

PC\_8.setText("0");

PC\_9.setText("0");

PC\_10.setText("0");

PC\_11.setText("0");

MAR\_0.setText("0");

MAR\_1.setText("0");

MAR\_2.setText("0");

MAR\_3.setText("0");

MAR\_4.setText("0");

MAR\_5.setText("0");

MAR\_6.setText("0");

MAR\_7.setText("0");

MAR\_8.setText("0");

MAR\_9.setText("0");

MAR\_10.setText("0");

MAR\_11.setText("0");

MBR\_0.setText("0");

MBR\_1.setText("0");

MBR\_2.setText("0");

MBR\_3.setText("0");

MBR\_4.setText("0");

MBR\_5.setText("0");

MBR\_6.setText("0");

MBR\_7.setText("0");

MBR\_8.setText("0");

MBR\_9.setText("0");

MBR\_10.setText("0");

MBR\_11.setText("0");

MBR\_12.setText("0");

MBR\_13.setText("0");

MBR\_14.setText("0");

MBR\_15.setText("0");

IR\_0.setText("0");

IR\_1.setText("0");

IR\_2.setText("0");

IR\_3.setText("0");

IR\_4.setText("0");

IR\_5.setText("0");

IR\_6.setText("0");

IR\_7.setText("0");

IR\_8.setText("0");

IR\_9.setText("0");

IR\_10.setText("0");

IR\_11.setText("0");

IR\_12.setText("0");

IR\_13.setText("0");

IR\_14.setText("0");

IR\_15.setText("0");

PC\_0.setBackground(Color.white);

PC\_1.setBackground(Color.white);

PC\_2.setBackground(Color.white);

PC\_3.setBackground(Color.white);

PC\_4.setBackground(Color.white);

PC\_5.setBackground(Color.white);

PC\_6.setBackground(Color.white);

PC\_7.setBackground(Color.white);

PC\_8.setBackground(Color.white);

PC\_9.setBackground(Color.white);

PC\_10.setBackground(Color.white);

PC\_11.setBackground(Color.white);

MAR\_0.setBackground(Color.white);

MAR\_1.setBackground(Color.white);

MAR\_2.setBackground(Color.white);

MAR\_3.setBackground(Color.white);

MAR\_4.setBackground(Color.white);

MAR\_5.setBackground(Color.white);

MAR\_6.setBackground(Color.white);

MAR\_7.setBackground(Color.white);

MAR\_8.setBackground(Color.white);

MAR\_9.setBackground(Color.white);

MAR\_10.setBackground(Color.white);

MAR\_11.setBackground(Color.white);

MBR\_0.setBackground(Color.white);

MBR\_1.setBackground(Color.white);

MBR\_2.setBackground(Color.white);

MBR\_3.setBackground(Color.white);

MBR\_4.setBackground(Color.white);

MBR\_5.setBackground(Color.white);

MBR\_6.setBackground(Color.white);

MBR\_7.setBackground(Color.white);

MBR\_8.setBackground(Color.white);

MBR\_9.setBackground(Color.white);

MBR\_10.setBackground(Color.white);

MBR\_11.setBackground(Color.white);

MBR\_12.setBackground(Color.white);

MBR\_13.setBackground(Color.white);

MBR\_14.setBackground(Color.white);

MBR\_15.setBackground(Color.white);

IR\_0.setBackground(Color.white);

IR\_1.setBackground(Color.white);

IR\_2.setBackground(Color.white);

IR\_3.setBackground(Color.white);

IR\_4.setBackground(Color.white);

IR\_5.setBackground(Color.white);

IR\_6.setBackground(Color.white);

IR\_7.setBackground(Color.white);

IR\_8.setBackground(Color.white);

IR\_9.setBackground(Color.white);

IR\_10.setBackground(Color.white);

IR\_11.setBackground(Color.white);

IR\_12.setBackground(Color.white);

IR\_13.setBackground(Color.white);

IR\_14.setBackground(Color.white);

IR\_15.setBackground(Color.white);

Reset\_Toggle\_Instruction();

Halt\_Display.setBackground(Color.black);

;

}

private void LoadActionPerformed(java.awt.event.ActionEvent evt) {//Load Action performed

MBR = (short) memory[MAR];

Populate\_MBR();

}

private void Clear\_AllActionPerformed(java.awt.event.ActionEvent evt) {// Function clears all the LED's

Clear();

Reset\_Toggle\_Instruction();

}

public double convertToFloat(int content) {

double number;

String bits = String.format("%16S", Integer.toBinaryString(content).replace(' ', '0'));

bits = bits.replace(' ', '0');

int S\_Bit = Integer.parseInt(bits.substring(0, 1), 2);

int Exponent = Integer.parseInt(bits.substring(1, 8), 2) - 63;

int Mantissa = Integer.parseInt(bits.substring(8), 2);

number = Mantissa \* Math.pow(10, Exponent);

if (S\_Bit == 0) {

number = 0 - number;

}

// System.out.println(convertToFloat("33538")); // Sting 1100001100000010

return number;

}

private void SSActionPerformed(java.awt.event.ActionEvent evt) {//Performs the SS action

new Thread(() -> {

try {

ExecuteInstruction();

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

}).start();

}

private void RunActionPerformed(java.awt.event.ActionEvent evt) {// Performs the run function which executes all the instructions

new Thread(new Runnable() {

@Override

public void run() {

JPanel\_Run.setBackground(Color.red);

short flag\_Halt = 0;

while (flag\_Halt == 0) {

try {

System.out.println(IR + "y");

ExecuteInstruction();

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

if (IR == 0) {

break;

}

}

}

}).start();

}

private void Custom\_InitActionPerformed(java.awt.event.ActionEvent evt) {//Prompts the user to upload a custom Init file to load the PC

/\*// Uses this process to invoke the constructor,

// JFileChooser points to user's default directory

JFileChooser j = new JFileChooser();

// Open the save dialog box\*/

JFileChooser jfc = new JFileChooser(FileSystemView.getFileSystemView().getHomeDirectory());

int returnValue = jfc.showOpenDialog(null);

if (returnValue == JFileChooser.APPROVE\_OPTION) {

File selectedFile = jfc.getSelectedFile();

BufferedReader br = null;

try {

br = new BufferedReader(new FileReader(selectedFile));

} catch (FileNotFoundException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

String st;

try {

short flag\_PC = 0;

while ((st = br.readLine()) != null) {

String[] content = st.split(" ");

System.out.println(content[0] + content[1]);

short location = Short.parseShort(content[0], 16);

short value = Short.parseShort(content[1], 16);

System.out.println(location + " " + value);

memory[location] = value;

if (flag\_PC == 0) {

flag\_PC++;

PC = location;

Populate\_PC();

}

}

} catch (IOException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Reset\_Toggle\_Instruction();

}

}

private void InitActionPerformed(java.awt.event.ActionEvent evt) {//Reads the init.txt file and loads the pc with the first address in the file

File file = new File("Init.txt");

BufferedReader br = null;

try {

br = new BufferedReader(new FileReader(file));

} catch (FileNotFoundException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

String st;

try {

short flag\_PC = 0;

while ((st = br.readLine()) != null) {

String[] content = st.split(" ");

int location = Integer.parseInt(content[0], 16);

int value = Integer.parseInt(content[1], 16);

System.out.println(location + " " + value);

memory[location] = (short) value;

if (flag\_PC == 0) {

flag\_PC++;

PC = (short) location;

Populate\_PC();

}

}

} catch (IOException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Reset\_Toggle\_Instruction();

}

public void Populate\_MFR() { //Populates the 4 BITS of MFR

String MFR\_binary = Integer.toBinaryString(MFR);

if (MFR\_binary.length() >= 1 && MFR\_binary.charAt(MFR\_binary.length() - 1) == '1') {

MFR\_0.setText("1");

MFR\_0.setBackground(Color.red);

} else {

MFR\_0.setText("0");

MFR\_0.setBackground(Color.white);

}

if (MFR\_binary.length() >= 2 && MFR\_binary.charAt(MFR\_binary.length() - 2) == '1') {

MFR\_1.setText("1");

MFR\_1.setBackground(Color.red);

} else {

MFR\_1.setText("0");

MFR\_1.setBackground(Color.white);

}

if (MFR\_binary.length() >= 3 && MFR\_binary.charAt(MFR\_binary.length() - 3) == '1') {

MFR\_2.setText("1");

MFR\_2.setBackground(Color.red);

} else {

MFR\_2.setText("0");

MFR\_2.setBackground(Color.white);

}

if (MFR\_binary.length() >= 4 && MFR\_binary.charAt(MFR\_binary.length() - 4) == '1') {

MFR\_3.setText("1");

MFR\_3.setBackground(Color.red);

} else {

MFR\_3.setText("0");

MFR\_3.setBackground(Color.white);

}

}

public void Populate\_PC() { //Populates the 12 BITS of PC

String PC\_binary = Integer.toBinaryString(PC);

if (PC\_binary.length() >= 1 && PC\_binary.charAt(PC\_binary.length() - 1) == '1') {

PC\_0.setText("1");

PC\_0.setBackground(Color.red);

} else {

PC\_0.setText("0");

PC\_0.setBackground(Color.white);

}

if (PC\_binary.length() >= 2 && PC\_binary.charAt(PC\_binary.length() - 2) == '1') {

PC\_1.setText("1");

PC\_1.setBackground(Color.red);

} else {

PC\_1.setText("0");

PC\_1.setBackground(Color.white);

}

if (PC\_binary.length() >= 3 && PC\_binary.charAt(PC\_binary.length() - 3) == '1') {

PC\_2.setText("1");

PC\_2.setBackground(Color.red);

} else {

PC\_2.setText("0");

PC\_2.setBackground(Color.white);

}

if (PC\_binary.length() >= 4 && PC\_binary.charAt(PC\_binary.length() - 4) == '1') {

PC\_3.setText("1");

PC\_3.setBackground(Color.red);

} else {

PC\_3.setText("0");

PC\_3.setBackground(Color.white);

}

if (PC\_binary.length() >= 5 && PC\_binary.charAt(PC\_binary.length() - 5) == '1') {

PC\_4.setText("1");

PC\_4.setBackground(Color.red);

} else {

PC\_4.setText("0");

PC\_4.setBackground(Color.white);

}

if (PC\_binary.length() >= 6 && PC\_binary.charAt(PC\_binary.length() - 6) == '1') {

PC\_5.setText("1");

PC\_5.setBackground(Color.red);

} else {

PC\_5.setText("0");

PC\_5.setBackground(Color.white);

}

if (PC\_binary.length() >= 7 && PC\_binary.charAt(PC\_binary.length() - 7) == '1') {

PC\_6.setText("1");

PC\_6.setBackground(Color.red);

} else {

PC\_6.setText("0");

PC\_6.setBackground(Color.white);

}

if (PC\_binary.length() >= 8 && PC\_binary.charAt(PC\_binary.length() - 8) == '1') {

PC\_7.setText("1");

PC\_7.setBackground(Color.red);

} else {

PC\_7.setText("0");

PC\_7.setBackground(Color.white);

}

if (PC\_binary.length() >= 9 && PC\_binary.charAt(PC\_binary.length() - 9) == '1') {

PC\_8.setText("1");

PC\_8.setBackground(Color.red);

} else {

PC\_8.setText("0");

PC\_8.setBackground(Color.white);

}

if (PC\_binary.length() >= 10 && PC\_binary.charAt(PC\_binary.length() - 10) == '1') {

PC\_9.setText("1");

PC\_9.setBackground(Color.red);

} else {

PC\_9.setText("0");

PC\_9.setBackground(Color.white);

}

if (PC\_binary.length() >= 11 && PC\_binary.charAt(PC\_binary.length() - 11) == '1') {

PC\_10.setText("1");

PC\_10.setBackground(Color.red);

} else {

PC\_10.setText("0");

PC\_10.setBackground(Color.white);

}

if (PC\_binary.length() >= 12 && PC\_binary.charAt(PC\_binary.length() - 12) == '1') {

PC\_11.setText("1");

PC\_11.setBackground(Color.red);

} else {

PC\_11.setText("0");

PC\_11.setBackground(Color.white);

}

}

public short getAddress(short x) { //Returns the Address of x

String x\_binary = Integer.toBinaryString(x & 0xFFFF);

if (x\_binary.length() == 32) {

x\_binary = x\_binary.substring(16, 32);

}

short address = 0;

if (x\_binary.length() >= 5 && x\_binary.charAt(x\_binary.length() - 5) == '1') {

address += Math.pow(2, 4);

}

if (x\_binary.length() >= 4 && x\_binary.charAt(x\_binary.length() - 4) == '1') {

address += Math.pow(2, 3);

}

if (x\_binary.length() >= 3 && x\_binary.charAt(x\_binary.length() - 3) == '1') {

address += Math.pow(2, 2);

}

if (x\_binary.length() >= 2 && x\_binary.charAt(x\_binary.length() - 2) == '1') {

address += Math.pow(2, 1);

}

if (x\_binary.length() >= 1 && x\_binary.charAt(x\_binary.length() - 1) == '1') {

address += Math.pow(2, 0);

}

return address;

}

public boolean memoryOutofBoundCheck(short memoryAdr) { //Checks the memory size

if (memoryAdr > 2056) {

MFR = 8;

PC = (short) (memory[1]);

Populate\_PC();

} else if (memoryAdr > 0 && memoryAdr <= 5) {

MFR = 1;

PC = (short) (memory[1]);

Populate\_PC();

} else {

return false;

}

Populate\_MFR();

return true;

}

public void Populate\_MBR() { //Populates the 16 BITS of MBR

String MBR\_binary = Integer.toBinaryString(MBR);

if (MBR\_binary.length() >= 1 && MBR\_binary.charAt(MBR\_binary.length() - 1) == '1') {

MBR\_0.setText("1");

MBR\_0.setBackground(Color.red);

} else {

MBR\_0.setText("0");

MBR\_0.setBackground(Color.white);

}

if (MBR\_binary.length() >= 2 && MBR\_binary.charAt(MBR\_binary.length() - 2) == '1') {

MBR\_1.setText("1");

MBR\_1.setBackground(Color.red);

} else {

MBR\_1.setText("0");

MBR\_1.setBackground(Color.white);

}

if (MBR\_binary.length() >= 3 && MBR\_binary.charAt(MBR\_binary.length() - 3) == '1') {

MBR\_2.setText("1");

MBR\_2.setBackground(Color.red);

} else {

MBR\_2.setText("0");

MBR\_2.setBackground(Color.white);

}

if (MBR\_binary.length() >= 4 && MBR\_binary.charAt(MBR\_binary.length() - 4) == '1') {

MBR\_3.setText("1");

MBR\_3.setBackground(Color.red);

} else {

MBR\_3.setText("0");

MBR\_3.setBackground(Color.white);

}

if (MBR\_binary.length() >= 5 && MBR\_binary.charAt(MBR\_binary.length() - 5) == '1') {

MBR\_4.setText("1");

MBR\_4.setBackground(Color.red);

} else {

MBR\_4.setText("0");

MBR\_4.setBackground(Color.white);

}

if (MBR\_binary.length() >= 6 && MBR\_binary.charAt(MBR\_binary.length() - 6) == '1') {

MBR\_5.setText("1");

MBR\_5.setBackground(Color.red);

} else {

MBR\_5.setText("0");

MBR\_5.setBackground(Color.white);

}

if (MBR\_binary.length() >= 7 && MBR\_binary.charAt(MBR\_binary.length() - 7) == '1') {

MBR\_6.setText("1");

MBR\_6.setBackground(Color.red);

} else {

MBR\_6.setText("0");

MBR\_6.setBackground(Color.white);

}

if (MBR\_binary.length() >= 8 && MBR\_binary.charAt(MBR\_binary.length() - 8) == '1') {

MBR\_7.setText("1");

MBR\_7.setBackground(Color.red);

} else {

MBR\_7.setText("0");

MBR\_7.setBackground(Color.white);

}

if (MBR\_binary.length() >= 9 && MBR\_binary.charAt(MBR\_binary.length() - 9) == '1') {

MBR\_8.setText("1");

MBR\_8.setBackground(Color.red);

} else {

MBR\_8.setText("0");

}

if (MBR\_binary.length() >= 10 && MBR\_binary.charAt(MBR\_binary.length() - 10) == '1') {

MBR\_9.setText("1");

MBR\_9.setBackground(Color.red);

} else {

MBR\_9.setText("0");

MBR\_9.setBackground(Color.white);

}

if (MBR\_binary.length() >= 11 && MBR\_binary.charAt(MBR\_binary.length() - 11) == '1') {

MBR\_10.setText("1");

MBR\_10.setBackground(Color.red);

} else {

MBR\_10.setText("0");

MBR\_10.setBackground(Color.white);

}

if (MBR\_binary.length() >= 12 && MBR\_binary.charAt(MBR\_binary.length() - 12) == '1') {

MBR\_11.setText("1");

MBR\_11.setBackground(Color.red);

} else {

MBR\_11.setText("0");

MBR\_11.setBackground(Color.white);

}

if (MBR\_binary.length() >= 13 && MBR\_binary.charAt(MBR\_binary.length() - 13) == '1') {

MBR\_12.setText("1");

MBR\_12.setBackground(Color.red);

} else {

MBR\_12.setText("0");

MBR\_12.setBackground(Color.white);

}

if (MBR\_binary.length() >= 14 && MBR\_binary.charAt(MBR\_binary.length() - 14) == '1') {

MBR\_13.setText("1");

MBR\_13.setBackground(Color.red);

} else {

MBR\_13.setText("0");

MBR\_13.setBackground(Color.white);

}

if (MBR\_binary.length() >= 15 && MBR\_binary.charAt(MBR\_binary.length() - 15) == '1') {

MBR\_14.setText("1");

MBR\_14.setBackground(Color.red);

} else {

MBR\_14.setText("0");

MBR\_14.setBackground(Color.white);

}

if (MBR\_binary.length() >= 16 && MBR\_binary.charAt(MBR\_binary.length() - 16) == '1') {

MBR\_15.setText("1");

MBR\_15.setBackground(Color.red);

} else {

MBR\_15.setText("0");

MBR\_15.setBackground(Color.white);

}

}

private void StoreActionPerformed(java.awt.event.ActionEvent evt) {//Store operation is performed

memory[MAR] = MBR;

Clear();

}

private void Load\_X3ActionPerformed(java.awt.event.ActionEvent evt) {//X3 is loaded with values

X3 = 0;

if (ADDRESS\_0.isSelected()) {

X3\_0.setText("1");

X3 += Math.pow(2, 0);

X3\_0.setBackground(Color.red);

} else {

X3\_0.setText("0");

X3\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

X3 += Math.pow(2, 1);

X3\_1.setText("1");

X3\_1.setBackground(Color.red);

} else {

X3\_1.setText("0");

X3\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

X3 += Math.pow(2, 2);

X3\_2.setText("1");

X3\_2.setBackground(Color.red);

} else {

X3\_2.setText("0");

X3\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

X3 += Math.pow(2, 3);

X3\_3.setText("1");

X3\_3.setBackground(Color.red);

} else {

X3\_3.setText("0");

X3\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

X3 += Math.pow(2, 4);

X3\_4.setText("1");

X3\_4.setBackground(Color.red);

} else {

X3\_4.setText("0");

X3\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

X3 += Math.pow(2, 5);

X3\_5.setText("1");

X3\_5.setBackground(Color.red);

} else {

X3\_5.setText("0");

X3\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

X3 += Math.pow(2, 6);

X3\_6.setText("1");

X3\_6.setBackground(Color.red);

} else {

X3\_6.setText("0");

X3\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

X3 += Math.pow(2, 7);

X3\_7.setText("1");

X3\_7.setBackground(Color.red);

} else {

X3\_7.setText("0");

X3\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

X3 += Math.pow(2, 8);

X3\_8.setText("1");

X3\_8.setBackground(Color.red);

} else {

X3\_8.setText("0");

X3\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

X3 += Math.pow(2, 9);

X3\_9.setText("1");

X3\_9.setBackground(Color.red);

} else {

X3\_9.setText("0");

X3\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

X3 += Math.pow(2, 10);

X3\_10.setText("1");

X3\_10.setBackground(Color.red);

} else {

X3\_10.setText("0");

X3\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

X3 += Math.pow(2, 11);

X3\_11.setText("1");

X3\_11.setBackground(Color.red);

} else {

X3\_11.setText("0");

X3\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

X3 += Math.pow(2, 12);

X3\_12.setText("1");

X3\_12.setBackground(Color.red);

} else {

X3\_12.setText("0");

X3\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

X3 += Math.pow(2, 13);

X3\_13.setText("1");

X3\_13.setBackground(Color.red);

} else {

X3\_13.setText("0");

X3\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

X3 += Math.pow(2, 14);

X3\_14.setText("1");

X3\_14.setBackground(Color.red);

} else {

X3\_14.setText("0");

X3\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

X3 += Math.pow(2, 15);

X3\_15.setText("1");

X3\_15.setBackground(Color.red);

} else {

X3\_15.setText("0");

X3\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_X2ActionPerformed(java.awt.event.ActionEvent evt) {//X2 is loaded with values

X2 = 0;

if (ADDRESS\_0.isSelected()) {

X2\_0.setText("1");

X2 += Math.pow(2, 0);

X2\_0.setBackground(Color.red);

} else {

X2\_0.setText("0");

X2\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

X2 += Math.pow(2, 1);

X2\_1.setText("1");

X2\_1.setBackground(Color.red);

} else {

X2\_1.setText("0");

X2\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

X2 += Math.pow(2, 2);

X2\_2.setText("1");

X2\_2.setBackground(Color.red);

} else {

X2\_2.setText("0");

X2\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

X2 += Math.pow(2, 3);

X2\_3.setText("1");

X2\_3.setBackground(Color.red);

} else {

X2\_3.setText("0");

X2\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

X2 += Math.pow(2, 4);

X2\_4.setText("1");

X2\_4.setBackground(Color.red);

} else {

X2\_4.setText("0");

X2\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

X2 += Math.pow(2, 5);

X2\_5.setText("1");

X2\_5.setBackground(Color.red);

} else {

X2\_5.setText("0");

X2\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

X2 += Math.pow(2, 6);

X2\_6.setText("1");

X2\_6.setBackground(Color.red);

} else {

X2\_6.setText("0");

X2\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

X2 += Math.pow(2, 7);

X2\_7.setText("1");

X2\_7.setBackground(Color.red);

} else {

X2\_7.setText("0");

X2\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

X2 += Math.pow(2, 8);

X2\_8.setText("1");

X2\_8.setBackground(Color.red);

} else {

X2\_8.setText("0");

X2\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

X2 += Math.pow(2, 9);

X2\_9.setText("1");

X2\_9.setBackground(Color.red);

} else {

X2\_9.setText("0");

X2\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

X2 += Math.pow(2, 10);

X2\_10.setText("1");

X2\_10.setBackground(Color.red);

} else {

X2\_10.setText("0");

X2\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

X2 += Math.pow(2, 11);

X2\_11.setText("1");

X2\_11.setBackground(Color.red);

} else {

X2\_11.setText("0");

X2\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

X2 += Math.pow(2, 12);

X2\_12.setText("1");

X2\_12.setBackground(Color.red);

} else {

X2\_12.setText("0");

X2\_11.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

X2 += Math.pow(2, 13);

X2\_13.setText("1");

X2\_13.setBackground(Color.red);

} else {

X2\_13.setText("0");

X2\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

X2 += Math.pow(2, 14);

X2\_14.setText("1");

X2\_14.setBackground(Color.red);

} else {

X2\_14.setText("0");

X2\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

X2 += Math.pow(2, 15);

X2\_15.setText("1");

X2\_15.setBackground(Color.red);

} else {

X2\_15.setText("0");

X2\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

public int getIntValofFloat(double number) {

String s = String.valueOf(number);

int n = s.length();

StringBuilder num\_string = new StringBuilder();

int i, j, exponent, c;

for (i = 0; s.charAt(i) == '0' || s.charAt(i) == '.'; i++);

for (j = n - 1; s.charAt(j) == '0' || s.charAt(j) == '.'; j--);

c = s.indexOf('.');

if (c == -1) {

c = n;

}

num\_string.append(s.charAt(i));

if (i != j) {

num\_string.append('.');

}

for (int k = i + 1; k <= j; k++) {

if (s.charAt(k) != '.') {

num\_string.append(s.charAt(k));

}

}

if (i < c) {

exponent = c - i - 1;

} else {

exponent = c - i;

}

i = 0;

StringBuilder new\_num\_string = new StringBuilder();

while (true) {

if (num\_string.charAt(i) != '.') {

new\_num\_string.append(num\_string.charAt(i));

}

if (new\_num\_string.length() == 3) {

break;

}

i++;

}

exponent = exponent - 2;

int mantissa = Integer.parseInt(new\_num\_string.toString());

if (mantissa > 255) {

mantissa = Integer.parseInt(String.valueOf(mantissa).substring(0, 2));

exponent++;

}

String Final\_Num\_Bin = "";

if (number < 0) {

Final\_Num\_Bin = Final\_Num\_Bin + "0";

} else {

Final\_Num\_Bin = Final\_Num\_Bin + "1";

}

if (exponent == 0) {

Final\_Num\_Bin = Final\_Num\_Bin + "0000000";

} else if (exponent > 0) {

Final\_Num\_Bin = Final\_Num\_Bin + String.format("%7S", Integer.toBinaryString(64 + exponent).replace(' ', '0'));

} else if (exponent < 0) {

Final\_Num\_Bin = Final\_Num\_Bin + "0";

Final\_Num\_Bin = Final\_Num\_Bin + String.format("%6S", Integer.toBinaryString(64 - Math.abs(exponent)).replace(' ', '0'));

}

Final\_Num\_Bin = Final\_Num\_Bin + String.format("%8S", Integer.toBinaryString(mantissa).replace(' ', '0'));

Final\_Num\_Bin = Final\_Num\_Bin.substring(Final\_Num\_Bin.length() - 16);

return Integer.parseInt(Final\_Num\_Bin, 2);

}

private void Load\_X1ActionPerformed(java.awt.event.ActionEvent evt) {//X1 is loaded with values

X1 = 0;

if (ADDRESS\_0.isSelected()) {

X1\_0.setText("1");

X1 += Math.pow(2, 0);

X1\_0.setBackground(Color.red);

} else {

X1\_0.setText("0");

X2\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

X1 += Math.pow(2, 1);

X1\_1.setText("1");

X1\_1.setBackground(Color.red);

} else {

X1\_1.setText("0");

X2\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

X1 += Math.pow(2, 2);

X1\_2.setText("1");

X1\_2.setBackground(Color.red);

}

else {

X1\_2.setText("0");

X2\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

X1 += Math.pow(2, 3);

X1\_3.setText("1");

X1\_3.setBackground(Color.red);

} else {

X1\_3.setText("0");

X1\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

X1 += Math.pow(2, 4);

X1\_4.setText("1");

X1\_4.setBackground(Color.red);

} else {

X1\_4.setText("0");

X1\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

X1 += Math.pow(2, 5);

X1\_5.setText("1");

X1\_5.setBackground(Color.red);

} else {

X1\_5.setText("0");

X1\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

X1 += Math.pow(2, 6);

X1\_6.setText("1");

X1\_6.setBackground(Color.red);

}

else {

X1\_6.setText("0");

X1\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

X1 += Math.pow(2, 7);

X1\_7.setText("1");

X1\_7.setBackground(Color.red);

}

else {

X1\_7.setText("0");

X1\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

X1 += Math.pow(2, 8);

X1\_8.setText("1");

X1\_8.setBackground(Color.red);

}

else {

X1\_8.setText("0");

X1\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

X1 += Math.pow(2, 9);

X1\_9.setText("1");

X1\_9.setBackground(Color.red);

}

else {

X1\_9.setText("0");

X1\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

X1 += Math.pow(2, 10);

X1\_10.setText("1");

X1\_10.setBackground(Color.red);

}

else {

X1\_10.setText("0");

X1\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

X1 += Math.pow(2, 11);

X1\_11.setText("1");

X1\_11.setBackground(Color.red);

}

else {

X1\_11.setText("0");

X1\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

X1 += Math.pow(2, 12);

X1\_12.setText("1");

X1\_12.setBackground(Color.red);

}

else {

X1\_12.setText("0");

X1\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

X1 += Math.pow(2, 13);

X1\_13.setText("1");

X1\_13.setBackground(Color.red);

}

else {

X1\_13.setText("0");

X1\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

X1 += Math.pow(2, 14);

X1\_14.setText("1");

X1\_14.setBackground(Color.red);

}

else {

X1\_14.setText("0");

X1\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

X1 += Math.pow(2, 15);

X1\_15.setText("1");

X1\_15.setBackground(Color.red);

}

else {

X1\_15.setText("0");

X1\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_MARActionPerformed(java.awt.event.ActionEvent evt) {//MAR is loaded with values

MAR = 0;

if (ADDRESS\_0.isSelected()) {

MAR += Math.pow(2, 0);

MAR\_0.setText("1");

MAR\_0.setBackground(Color.red);

} else {

MAR\_0.setText("0");

MAR\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

MAR += Math.pow(2, 1);

MAR\_1.setText("1");

MAR\_1.setBackground(Color.red);

} else {

MAR\_1.setText("0");

MAR\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

MAR += Math.pow(2, 2);

MAR\_2.setText("1");

MAR\_2.setBackground(Color.red);

} else {

MAR\_2.setText("0");

MAR\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

MAR += Math.pow(2, 3);

MAR\_3.setText("1");

MAR\_3.setBackground(Color.red);

} else {

MAR\_3.setText("0");

MAR\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

MAR += Math.pow(2, 4);

MAR\_4.setText("1");

MAR\_4.setBackground(Color.red);

} else {

MAR\_4.setText("0");

MAR\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

MAR += Math.pow(2, 5);

MAR\_5.setText("1");

MAR\_5.setBackground(Color.red);

} else {

MAR\_5.setText("0");

MAR\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

MAR += Math.pow(2, 6);

MAR\_6.setText("1");

MAR\_6.setBackground(Color.red);

} else {

MAR\_6.setText("0");

MAR\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

MAR += Math.pow(2, 7);

MAR\_7.setText("1");

MAR\_7.setBackground(Color.red);

} else {

MAR\_7.setText("0");

MAR\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

MAR += Math.pow(2, 8);

MAR\_8.setText("1");

MAR\_8.setBackground(Color.red);

} else {

MAR\_8.setText("0");

MAR\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

MAR += Math.pow(2, 9);

MAR\_9.setText("1");

MAR\_9.setBackground(Color.red);

} else {

MAR\_9.setText("0");

MAR\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

MAR += Math.pow(2, 10);

MAR\_10.setText("1");

MAR\_10.setBackground(Color.red);

} else {

MAR\_10.setText("0");

MAR\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

MAR += Math.pow(2, 11);

MAR\_11.setText("1");

MAR\_11.setBackground(Color.red);

} else {

MAR\_11.setText("0");

MAR\_11.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_MBRActionPerformed(java.awt.event.ActionEvent evt) {//MBR is loaded with values

MBR = 0;

if (ADDRESS\_0.isSelected()) {

MBR\_0.setText("1");

MBR += Math.pow(2, 0);

MBR\_0.setBackground(Color.red);

} else {

MBR\_0.setText("0");

MBR\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

MBR += Math.pow(2, 1);

MBR\_1.setText("1");

MBR\_1.setBackground(Color.red);

} else {

MBR\_1.setText("0");

MBR\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

MBR += Math.pow(2, 2);

MBR\_2.setText("1");

MBR\_2.setBackground(Color.red);

} else {

MBR\_2.setText("0");

MBR\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

MBR += Math.pow(2, 3);

MBR\_3.setText("1");

MBR\_3.setBackground(Color.red);

} else {

MBR\_3.setText("0");

MBR\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

MBR += Math.pow(2, 4);

MBR\_4.setText("1");

MBR\_4.setBackground(Color.red);

} else {

MBR\_4.setText("0");

MBR\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

MBR += Math.pow(2, 5);

MBR\_5.setText("1");

MBR\_5.setBackground(Color.red);

} else {

MBR\_5.setText("0");

MBR\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

MBR += Math.pow(2, 6);

MBR\_6.setText("1");

MBR\_6.setBackground(Color.red);

} else {

MBR\_6.setText("0");

MBR\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

MBR += Math.pow(2, 7);

MBR\_7.setText("1");

MBR\_7.setBackground(Color.red);

} else {

MBR\_7.setText("0");

MBR\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

MBR += Math.pow(2, 8);

MBR\_8.setText("1");

MBR\_8.setBackground(Color.red);

} else {

MBR\_8.setText("0");

MBR\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

MBR += Math.pow(2, 9);

MBR\_9.setText("1");

MBR\_9.setBackground(Color.red);

} else {

MBR\_9.setText("0");

MBR\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

MBR += Math.pow(2, 10);

MBR\_10.setText("1");

MBR\_10.setBackground(Color.red);

} else {

MBR\_10.setText("0");

MBR\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

MBR += Math.pow(2, 11);

MBR\_11.setText("1");

MBR\_11.setBackground(Color.red);

} else {

MBR\_11.setText("0");

MBR\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

MBR += Math.pow(2, 12);

MBR\_12.setText("1");

MBR\_12.setBackground(Color.red);

} else {

MBR\_12.setText("0");

MBR\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

MBR += Math.pow(2, 13);

MBR\_13.setText("1");

MBR\_13.setBackground(Color.red);

} else {

MBR\_13.setText("0");

MBR\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

MBR += Math.pow(2, 14);

MBR\_14.setText("1");

MBR\_14.setBackground(Color.red);

} else {

MBR\_14.setText("0");

MBR\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

MBR += Math.pow(2, 15);

MBR\_15.setText("1");

MBR\_15.setBackground(Color.red);

} else {

MBR\_15.setText("0");

MBR\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_PCActionPerformed(java.awt.event.ActionEvent evt) {//PC is loaded with values

PC = 0;

if (ADDRESS\_0.isSelected()) {

PC += Math.pow(2, 0);

PC\_0.setText("1");

PC\_0.setBackground(Color.red);

} else {

PC\_0.setText("0");

PC\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

PC += Math.pow(2, 1);

PC\_1.setText("1");

PC\_1.setBackground(Color.red);

} else {

PC\_1.setText("0");

PC\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

PC += Math.pow(2, 2);

PC\_2.setText("1");

PC\_2.setBackground(Color.red);

} else {

PC\_2.setText("0");

PC\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

PC += Math.pow(2, 3);

PC\_3.setText("1");

PC\_3.setBackground(Color.red);

} else {

PC\_3.setText("0");

PC\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

PC += Math.pow(2, 4);

PC\_4.setText("1");

PC\_4.setBackground(Color.red);

} else {

PC\_4.setText("0");

PC\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

PC += Math.pow(2, 5);

PC\_5.setText("1");

PC\_5.setBackground(Color.red);

} else {

PC\_5.setText("0");

PC\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

PC += Math.pow(2, 6);

PC\_6.setText("1");

PC\_6.setBackground(Color.red);

} else {

PC\_6.setText("0");

PC\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

PC += Math.pow(2, 7);

PC\_7.setText("1");

PC\_7.setBackground(Color.red);

} else {

PC\_7.setText("0");

PC\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

PC += Math.pow(2, 8);

PC\_8.setText("1");

PC\_8.setBackground(Color.red);

} else {

PC\_8.setText("0");

PC\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

PC += Math.pow(2, 9);

PC\_9.setText("1");

PC\_9.setBackground(Color.red);

} else {

PC\_9.setText("0");

PC\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

PC += Math.pow(2, 10);

PC\_10.setText("1");

PC\_10.setBackground(Color.red);

} else {

PC\_10.setText("0");

PC\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

PC += Math.pow(2, 11);

PC\_11.setText("1");

PC\_11.setBackground(Color.red);

} else {

PC\_11.setText("0");

PC\_11.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_R3ActionPerformed(java.awt.event.ActionEvent evt) {//R3 is loaded with values

R3 = 0;

if (ADDRESS\_0.isSelected()) {

R3\_0.setText("1");

R3 += Math.pow(2, 0);

R3\_0.setBackground(Color.red);

} else {

R3\_0.setText("0");

R3\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

R3 += Math.pow(2, 1);

R3\_1.setText("1");

R3\_1.setBackground(Color.red);

} else {

R3\_1.setText("0");

R3\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

R3 += Math.pow(2, 2);

R3\_2.setText("1");

R3\_2.setBackground(Color.red);

} else {

R3\_2.setText("0");

R3\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

R3 += Math.pow(2, 3);

R3\_3.setText("1");

R3\_3.setBackground(Color.red);

} else {

R3\_3.setText("0");

R3\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

R3 += Math.pow(2, 4);

R3\_4.setText("1");

R3\_4.setBackground(Color.red);

} else {

R3\_4.setText("0");

R3\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

R3 += Math.pow(2, 5);

R3\_5.setText("1");

R3\_5.setBackground(Color.red);

} else {

R3\_5.setText("0");

R3\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

R3 += Math.pow(2, 6);

R3\_6.setText("1");

R3\_6.setBackground(Color.red);

} else {

R3\_6.setText("0");

R3\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

R3 += Math.pow(2, 7);

R3\_7.setText("1");

R3\_7.setBackground(Color.red);

} else {

R3\_7.setText("0");

R3\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

R3 += Math.pow(2, 8);

R3\_8.setText("1");

R3\_8.setBackground(Color.red);

} else {

R3\_8.setText("0");

R3\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

R3 += Math.pow(2, 9);

R3\_9.setText("1");

R3\_9.setBackground(Color.red);

} else {

R3\_9.setText("0");

R3\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

R3 += Math.pow(2, 10);

R3\_10.setText("1");

R3\_10.setBackground(Color.red);

} else {

R3\_10.setText("0");

R3\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

R3 += Math.pow(2, 11);

R3\_11.setText("1");

R3\_11.setBackground(Color.red);

} else {

R3\_11.setText("0");

R3\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

R3 += Math.pow(2, 12);

R3\_12.setText("1");

R3\_12.setBackground(Color.red);

} else {

R3\_12.setText("0");

R3\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

R3 += Math.pow(2, 13);

R3\_13.setText("1");

R3\_13.setBackground(Color.red);

} else {

R3\_13.setText("0");

R3\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

R3 += Math.pow(2, 14);

R3\_14.setText("1");

R3\_14.setBackground(Color.red);

} else {

R3\_14.setText("0");

R3\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

R3 += Math.pow(2, 15);

R3\_15.setText("1");

R3\_15.setBackground(Color.red);

} else {

R3\_15.setText("0");

R3\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_R2ActionPerformed(java.awt.event.ActionEvent evt) {//R2 is loaded with values

R2 = 0;

if (ADDRESS\_0.isSelected()) {

R2\_0.setText("1");

R2 += Math.pow(2, 0);

R2\_0.setBackground(Color.red);

} else {

R2\_0.setText("0");

R2\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

R2 += Math.pow(2, 1);

R2\_1.setText("1");

R2\_1.setBackground(Color.red);

} else {

R2\_1.setText("0");

R2\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

R2 += Math.pow(2, 2);

R2\_2.setText("1");

R2\_2.setBackground(Color.red);

} else {

R2\_2.setText("0");

R2\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

R2 += Math.pow(2, 3);

R2\_3.setText("1");

R2\_3.setBackground(Color.red);

} else {

R2\_3.setText("0");

R2\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

R2 += Math.pow(2, 4);

R2\_4.setText("1");

R2\_4.setBackground(Color.red);

} else {

R2\_4.setText("0");

R2\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

R2 += Math.pow(2, 5);

R2\_5.setText("1");

R2\_5.setBackground(Color.red);

} else {

R2\_5.setText("0");

R2\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

R2 += Math.pow(2, 6);

R2\_6.setText("1");

R2\_6.setBackground(Color.red);

} else {

R2\_6.setText("0");

R2\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

R2 += Math.pow(2, 7);

R2\_7.setText("1");

R2\_7.setBackground(Color.red);

} else {

R2\_7.setText("0");

R2\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

R2 += Math.pow(2, 8);

R2\_8.setText("1");

R2\_8.setBackground(Color.red);

} else {

R2\_8.setText("0");

R2\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

R2 += Math.pow(2, 9);

R2\_9.setText("1");

R2\_9.setBackground(Color.red);

} else {

R2\_9.setText("0");

R2\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

R2 += Math.pow(2, 10);

R2\_10.setText("1");

R2\_10.setBackground(Color.red);

} else {

R2\_10.setText("0");

R2\_10.setBackground((Color.white));

}

if (OP\_11.isSelected()) {

R2 += Math.pow(2, 11);

R2\_11.setText("1");

R2\_11.setBackground(Color.red);

} else {

R2\_11.setText("0");

R2\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

R2 += Math.pow(2, 12);

R2\_12.setText("1");

R2\_12.setBackground(Color.red);

} else {

R2\_12.setText("0");

R2\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

R2 += Math.pow(2, 13);

R2\_13.setText("1");

R2\_13.setBackground(Color.red);

} else {

R2\_13.setText("0");

R2\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

R2 += Math.pow(2, 14);

R2\_14.setText("1");

R2\_14.setBackground(Color.red);

} else {

R2\_14.setText("0");

R2\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

R2 += Math.pow(2, 15);

R2\_15.setText("1");

R2\_15.setBackground(Color.red);

} else {

R2\_15.setText("0");

R2\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_R1ActionPerformed(java.awt.event.ActionEvent evt) {//R1 is loaded with values

R1 = 0;

if (ADDRESS\_0.isSelected()) {

R1\_0.setText("1");

R1 += Math.pow(2, 0);

R1\_0.setBackground(Color.red);

} else {

R1\_0.setText("0");

R1\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

R1 += Math.pow(2, 1);

R1\_1.setText("1");

R1\_1.setBackground(Color.red);

} else {

R1\_1.setText("0");

R1\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

R1 += Math.pow(2, 2);

R1\_2.setText("1");

R1\_2.setBackground(Color.red);

} else {

R1\_2.setText("0");

R1\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

R1 += Math.pow(2, 3);

R1\_3.setText("1");

R1\_3.setBackground(Color.red);

} else {

R1\_3.setText("0");

R1\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

R1 += Math.pow(2, 4);

R1\_4.setText("1");

R1\_4.setBackground(Color.red);

} else {

R1\_4.setText("0");

R1\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

R1 += Math.pow(2, 5);

R1\_5.setText("1");

R1\_5.setBackground(Color.red);

} else {

R1\_5.setText("0");

R1\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

R1 += Math.pow(2, 6);

R1\_6.setText("1");

R1\_6.setBackground(Color.red);

} else {

R1\_6.setText("0");

R1\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

R1 += Math.pow(2, 7);

R1\_7.setText("1");

R1\_7.setBackground(Color.red);

} else {

R1\_7.setText("0");

R1\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

R1 += Math.pow(2, 8);

R1\_8.setText("1");

R1\_8.setBackground(Color.red);

} else {

R1\_8.setText("0");

R1\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

R1 += Math.pow(2, 9);

R1\_9.setText("1");

R1\_9.setBackground(Color.red);

} else {

R1\_9.setText("0");

R1\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

R1 += Math.pow(2, 10);

R1\_10.setText("1");

R1\_10.setBackground(Color.red);

} else {

R1\_10.setText("0");

R1\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

R1 += Math.pow(2, 11);

R1\_11.setText("1");

R1\_11.setBackground(Color.red);

} else {

R1\_11.setText("0");

R1\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

R1 += Math.pow(2, 12);

R1\_12.setText("1");

R1\_12.setBackground(Color.red);

} else {

R1\_12.setText("0");

R1\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

R1 += Math.pow(2, 13);

R1\_13.setText("1");

R1\_13.setBackground(Color.red);

} else {

R1\_13.setText("0");

R1\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

R1 += Math.pow(2, 14);

R1\_14.setText("1");

R1\_14.setBackground(Color.red);

} else {

R1\_14.setText("0");

R1\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

R1 += Math.pow(2, 15);

R1\_15.setText("1");

R1\_15.setBackground(Color.red);

} else {

R1\_15.setText("0");

R1\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_R0ActionPerformed(java.awt.event.ActionEvent evt) {//R0 is loaded with values

R0 = 0;

if (ADDRESS\_0.isSelected()) {

R0\_0.setText("1");

R0 += Math.pow(2, 0);

R0\_0.setBackground(Color.red);

} else {

R0\_0.setText("0");

R0\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

R0 += Math.pow(2, 1);

R0\_1.setText("1");

R0\_1.setBackground(Color.red);

} else {

R0\_1.setText("0");

R0\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

R0 += Math.pow(2, 2);

R0\_2.setText("1");

R0\_2.setBackground(Color.red);

} else {

R0\_2.setText("0");

R0\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

R0 += Math.pow(2, 3);

R0\_3.setText("1");

R0\_3.setBackground(Color.red);

} else {

R0\_3.setText("0");

R0\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

R0 += Math.pow(2, 4);

R0\_4.setText("1");

R0\_4.setBackground(Color.red);

} else {

R0\_4.setText("0");

R0\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

R0 += Math.pow(2, 5);

R0\_5.setText("1");

R0\_5.setBackground(Color.red);

} else {

R0\_5.setText("0");

R0\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

R0 += Math.pow(2, 6);

R0\_6.setText("1");

R0\_6.setBackground(Color.red);

} else {

R0\_6.setText("0");

R0\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

R0 += Math.pow(2, 7);

R0\_7.setText("1");

R0\_7.setBackground(Color.red);

} else {

R0\_7.setText("0");

R0\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

R0 += Math.pow(2, 8);

R0\_8.setText("1");

R0\_8.setBackground(Color.red);

} else {

R0\_8.setText("0");

R0\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

R0 += Math.pow(2, 9);

R0\_9.setText("1");

R0\_9.setBackground(Color.red);

} else {

R0\_9.setText("0");

R0\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

R0 += Math.pow(2, 10);

R0\_10.setText("1");

R0\_10.setBackground(Color.red);

} else {

R0\_10.setText("0");

R0\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

R0 += Math.pow(2, 11);

R0\_11.setText("1");

R0\_11.setBackground(Color.red);

} else {

R0\_11.setText("0");

R0\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

R0 += Math.pow(2, 12);

R0\_12.setText("1");

R0\_12.setBackground(Color.red);

} else {

R0\_12.setText("0");

R0\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

R0 += Math.pow(2, 13);

R0\_13.setText("1");

R0\_13.setBackground(Color.red);

} else {

R0\_13.setText("0");

R0\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

R0 += Math.pow(2, 14);

R0\_14.setText("1");

R0\_14.setBackground(Color.red);

} else {

R0\_14.setText("0");

R0\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

R0 += Math.pow(2, 15);

R0\_15.setText("1");

R0\_15.setBackground(Color.red);

} else {

R0\_15.setText("0");

R0\_15.setBackground(Color.white);

}

Reset\_Toggle\_Instruction();

}

private void Load\_IRActionPerformed(java.awt.event.ActionEvent evt) {//IR is loaded with values

IR = 0;

if (ADDRESS\_0.isSelected()) {

IR += Math.pow(2, 0);

IR\_0.setText("1");

IR\_0.setBackground(Color.red);

} else {

IR\_0.setText("0");

IR\_0.setBackground(Color.white);

}

if (ADDRESS\_1.isSelected()) {

IR += Math.pow(2, 1);

IR\_1.setText("1");

IR\_1.setBackground(Color.red);

} else {

IR\_1.setText("0");

IR\_1.setBackground(Color.white);

}

if (ADDRESS\_2.isSelected()) {

IR += Math.pow(2, 2);

IR\_2.setText("1");

IR\_2.setBackground(Color.red);

} else {

IR\_2.setText("0");

IR\_2.setBackground(Color.white);

}

if (ADDRESS\_3.isSelected()) {

IR += Math.pow(2, 3);

IR\_3.setText("1");

IR\_3.setBackground(Color.red);

} else {

IR\_3.setText("0");

IR\_3.setBackground(Color.white);

}

if (ADDRESS\_4.isSelected()) {

IR += Math.pow(2, 4);

IR\_4.setText("1");

IR\_4.setBackground(Color.red);

} else {

IR\_4.setText("0");

IR\_4.setBackground(Color.white);

}

if (I\_5.isSelected()) {

IR += Math.pow(2, 5);

IR\_5.setText("1");

IR\_5.setBackground(Color.red);

} else {

IR\_5.setText("0");

IR\_5.setBackground(Color.white);

}

if (IXR\_6.isSelected()) {

IR += Math.pow(2, 6);

IR\_6.setText("1");

IR\_6.setBackground(Color.red);

} else {

IR\_6.setText("0");

IR\_6.setBackground(Color.white);

}

if (IXR\_7.isSelected()) {

IR += Math.pow(2, 7);

IR\_7.setText("1");

IR\_7.setBackground(Color.red);

} else {

IR\_7.setText("0");

IR\_7.setBackground(Color.white);

}

if (GPR\_8.isSelected()) {

IR += Math.pow(2, 8);

IR\_8.setText("1");

IR\_8.setBackground(Color.red);

} else {

IR\_8.setText("0");

IR\_8.setBackground(Color.white);

}

if (GPR\_9.isSelected()) {

IR += Math.pow(2, 9);

IR\_9.setText("1");

IR\_9.setBackground(Color.red);

} else {

IR\_9.setText("0");

IR\_9.setBackground(Color.white);

}

if (OP\_10.isSelected()) {

IR += Math.pow(2, 10);

IR\_10.setText("1");

IR\_10.setBackground(Color.red);

} else {

IR\_10.setText("0");

IR\_10.setBackground(Color.white);

}

if (OP\_11.isSelected()) {

IR += Math.pow(2, 11);

IR\_11.setText("1");

IR\_11.setBackground(Color.red);

} else {

IR\_11.setText("0");

IR\_11.setBackground(Color.white);

}

if (OP\_12.isSelected()) {

IR += Math.pow(2, 12);

IR\_12.setText("1");

IR\_12.setBackground(Color.red);

} else {

IR\_12.setText("0");

IR\_12.setBackground(Color.white);

}

if (OP\_13.isSelected()) {

IR += Math.pow(2, 13);

IR\_13.setText("1");

IR\_13.setBackground(Color.red);

} else {

IR\_13.setText("0");

IR\_13.setBackground(Color.white);

}

if (OP\_14.isSelected()) {

IR += Math.pow(2, 14);

IR\_14.setText("1");

IR\_14.setBackground(Color.red);

} else {

IR\_14.setText("0");

IR\_14.setBackground(Color.white);

}

if (OP\_15.isSelected()) {

IR += Math.pow(2, 15);

IR\_15.setText("1");

IR\_15.setBackground(Color.red);

} else {

IR\_15.setText("0");

IR\_15.setBackground(Color.white);

}

System.out.println(IR);

Reset\_Toggle\_Instruction();

}

public void Populate\_CC() {//Populate 4 bits of CC

if (CC0 == 1) {

CC\_0.setText("1");

CC\_0.setBackground(Color.red);

} else {

CC\_0.setText("0");

CC\_0.setBackground(Color.white);

}

if (CC1 == 1) {

CC\_1.setText("1");

CC\_1.setBackground(Color.red);

} else {

CC\_1.setText("0");

CC\_1.setBackground(Color.white);

}

if (CC2 == 1) {

CC\_2.setText("1");

CC\_2.setBackground(Color.red);

} else {

CC\_2.setText("0");

CC\_2.setBackground(Color.white);

}

if (CC3 == 1) {

CC\_3.setText("1");

CC\_3.setBackground(Color.red);

} else {

CC\_3.setText("0");

CC\_3.setBackground(Color.white);

}

}

public void Populate\_R0() {//Populate 16 bits of R0

String R0\_binary = Integer.toBinaryString(R0);

if (R0\_binary.length() >= 1 && R0\_binary.charAt(R0\_binary.length() - 1) == '1') {

R0\_0.setText("1");

R0\_0.setBackground(Color.red);

} else {

R0\_0.setText("0");

R0\_0.setBackground(Color.white);

}

if (R0\_binary.length() >= 2 && R0\_binary.charAt(R0\_binary.length() - 2) == '1') {

R0\_1.setText("1");

R0\_1.setBackground(Color.red);

} else {

R0\_1.setText("0");

R0\_1.setBackground(Color.white);

}

if (R0\_binary.length() >= 3 && R0\_binary.charAt(R0\_binary.length() - 3) == '1') {

R0\_2.setText("1");

R0\_2.setBackground(Color.red);

} else {

R0\_2.setText("0");

R0\_2.setBackground(Color.white);

}

if (R0\_binary.length() >= 4 && R0\_binary.charAt(R0\_binary.length() - 4) == '1') {

R0\_3.setText("1");

R0\_3.setBackground(Color.red);

} else {

R0\_3.setText("0");

R0\_3.setBackground(Color.white);

}

if (R0\_binary.length() >= 5 && R0\_binary.charAt(R0\_binary.length() - 5) == '1') {

R0\_4.setText("1");

R0\_4.setBackground(Color.red);

} else {

R0\_4.setText("0");

R0\_4.setBackground(Color.white);

}

if (R0\_binary.length() >= 6 && R0\_binary.charAt(R0\_binary.length() - 6) == '1') {

R0\_5.setText("1");

R0\_5.setBackground(Color.red);

} else {

R0\_5.setText("0");

R0\_5.setBackground(Color.white);

}

if (R0\_binary.length() >= 7 && R0\_binary.charAt(R0\_binary.length() - 7) == '1') {

R0\_6.setText("1");

R0\_6.setBackground(Color.red);

} else {

R0\_6.setText("0");

R0\_6.setBackground(Color.white);

}

if (R0\_binary.length() >= 8 && R0\_binary.charAt(R0\_binary.length() - 8) == '1') {

R0\_7.setText("1");

R0\_7.setBackground(Color.red);

} else {

R0\_7.setText("0");

R0\_7.setBackground(Color.white);

}

if (R0\_binary.length() >= 9 && R0\_binary.charAt(R0\_binary.length() - 9) == '1') {

R0\_8.setText("1");

R0\_8.setBackground(Color.red);

} else {

R0\_8.setText("0");

R0\_8.setBackground(Color.white);

}

if (R0\_binary.length() >= 10 && R0\_binary.charAt(R0\_binary.length() - 10) == '1') {

R0\_9.setText("1");

R0\_9.setBackground(Color.red);

} else {

R0\_9.setText("0");

R0\_9.setBackground(Color.white);

}

if (R0\_binary.length() >= 11 && R0\_binary.charAt(R0\_binary.length() - 11) == '1') {

R0\_10.setText("1");

R0\_10.setBackground(Color.red);

} else {

R0\_10.setText("0");

R0\_10.setBackground(Color.white);

}

if (R0\_binary.length() >= 12 && R0\_binary.charAt(R0\_binary.length() - 12) == '1') {

R0\_11.setText("1");

R0\_11.setBackground(Color.red);

} else {

R0\_11.setText("0");

R0\_11.setBackground(Color.white);

}

if (R0\_binary.length() >= 13 && R0\_binary.charAt(R0\_binary.length() - 13) == '1') {

R0\_12.setText("1");

R0\_12.setBackground(Color.red);

} else {

R0\_12.setText("0");

R0\_12.setBackground(Color.white);

}

if (R0\_binary.length() >= 14 && R0\_binary.charAt(R0\_binary.length() - 14) == '1') {

R0\_13.setText("1");

R0\_13.setBackground(Color.red);

} else {

R0\_13.setText("0");

R0\_13.setBackground(Color.white);

}

if (R0\_binary.length() >= 15 && R0\_binary.charAt(R0\_binary.length() - 15) == '1') {

R0\_14.setText("1");

R0\_14.setBackground(Color.red);

} else {

R0\_14.setText("0");

R0\_14.setBackground(Color.white);

}

if (R0\_binary.length() >= 16 && R0\_binary.charAt(R0\_binary.length() - 16) == '1') {

R0\_15.setText("1");

R0\_15.setBackground(Color.red);

} else {

R0\_15.setText("0");

R0\_15.setBackground(Color.white);

}

}

public void Populate\_R1() {//Populate 16 bits of R1

String R1\_binary = Integer.toBinaryString(R1);

if (R1\_binary.length() >= 1 && R1\_binary.charAt(R1\_binary.length() - 1) == '1') {

R1\_0.setText("1");

R1\_0.setBackground(Color.red);

} else {

R1\_0.setText("0");

R1\_0.setBackground(Color.white);

}

if (R1\_binary.length() >= 2 && R1\_binary.charAt(R1\_binary.length() - 2) == '1') {

R1\_1.setText("1");

R1\_1.setBackground(Color.red);

} else {

R1\_1.setText("0");

R1\_1.setBackground(Color.white);

}

if (R1\_binary.length() >= 3 && R1\_binary.charAt(R1\_binary.length() - 3) == '1') {

R1\_2.setText("1");

R1\_2.setBackground(Color.red);

} else {

R1\_2.setText("0");

R1\_2.setBackground(Color.white);

}

if (R1\_binary.length() >= 4 && R1\_binary.charAt(R1\_binary.length() - 4) == '1') {

R1\_3.setText("1");

R1\_3.setBackground(Color.red);

} else {

R1\_3.setText("0");

R1\_3.setBackground(Color.white);

}

if (R1\_binary.length() >= 5 && R1\_binary.charAt(R1\_binary.length() - 5) == '1') {

R1\_4.setText("1");

R1\_4.setBackground(Color.red);

} else {

R1\_4.setText("0");

R1\_4.setBackground(Color.white);

}

if (R1\_binary.length() >= 6 && R1\_binary.charAt(R1\_binary.length() - 6) == '1') {

R1\_5.setText("1");

R1\_5.setBackground(Color.red);

} else {

R1\_5.setText("0");

R1\_5.setBackground(Color.white);

}

if (R1\_binary.length() >= 7 && R1\_binary.charAt(R1\_binary.length() - 7) == '1') {

R1\_6.setText("1");

R1\_6.setBackground(Color.red);

} else {

R1\_6.setText("0");

R1\_6.setBackground(Color.white);

}

if (R1\_binary.length() >= 8 && R1\_binary.charAt(R1\_binary.length() - 8) == '1') {

R1\_7.setText("1");

R1\_7.setBackground(Color.red);

} else {

R1\_7.setText("0");

R1\_7.setBackground(Color.white);

}

if (R1\_binary.length() >= 9 && R1\_binary.charAt(R1\_binary.length() - 9) == '1') {

R1\_8.setText("1");

R1\_8.setBackground(Color.red);

} else {

R1\_8.setText("0");

R1\_8.setBackground(Color.white);

}

if (R1\_binary.length() >= 10 && R1\_binary.charAt(R1\_binary.length() - 10) == '1') {

R1\_9.setText("1");

R1\_9.setBackground(Color.red);

} else {

R1\_9.setText("0");

R1\_9.setBackground(Color.white);

}

if (R1\_binary.length() >= 11 && R1\_binary.charAt(R1\_binary.length() - 11) == '1') {

R1\_10.setText("1");

R1\_10.setBackground(Color.red);

} else {

R1\_10.setText("0");

R1\_10.setBackground(Color.white);

}

if (R1\_binary.length() >= 12 && R1\_binary.charAt(R1\_binary.length() - 12) == '1') {

R1\_11.setText("1");

R1\_11.setBackground(Color.red);

} else {

R1\_11.setText("0");

R1\_11.setBackground(Color.white);

}

if (R1\_binary.length() >= 13 && R1\_binary.charAt(R1\_binary.length() - 13) == '1') {

R1\_12.setText("1");

R1\_12.setBackground(Color.red);

} else {

R1\_12.setText("0");

R1\_12.setBackground(Color.white);

}

if (R1\_binary.length() >= 14 && R1\_binary.charAt(R1\_binary.length() - 14) == '1') {

R1\_13.setText("1");

R1\_13.setBackground(Color.red);

} else {

R1\_13.setText("0");

R1\_13.setBackground(Color.white);

}

if (R1\_binary.length() >= 15 && R1\_binary.charAt(R1\_binary.length() - 15) == '1') {

R1\_14.setText("1");

R1\_14.setBackground(Color.red);

} else {

R1\_14.setText("0");

R1\_14.setBackground(Color.white);

}

if (R1\_binary.length() >= 16 && R1\_binary.charAt(R1\_binary.length() - 16) == '1') {

R1\_15.setText("1");

R1\_15.setBackground(Color.red);

} else {

R1\_15.setText("0");

R1\_15.setBackground(Color.white);

}

}

public void Populate\_R2() {//Populate 16 bits of R2

String R2\_binary = Integer.toBinaryString(R2);

if (R2\_binary.length() >= 1 && R2\_binary.charAt(R2\_binary.length() - 1) == '1') {

R2\_0.setText("1");

R2\_0.setBackground(Color.red);

} else {

R2\_0.setText("0");

R2\_0.setBackground(Color.white);

}

if (R2\_binary.length() >= 2 && R2\_binary.charAt(R2\_binary.length() - 2) == '1') {

R2\_1.setText("1");

R2\_1.setBackground(Color.red);

} else {

R2\_1.setText("0");

R2\_1.setBackground(Color.white);

}

if (R2\_binary.length() >= 3 && R2\_binary.charAt(R2\_binary.length() - 3) == '1') {

R2\_2.setText("1");

R2\_2.setBackground(Color.red);

} else {

R2\_2.setText("0");

R2\_2.setBackground(Color.white);

}

if (R2\_binary.length() >= 4 && R2\_binary.charAt(R2\_binary.length() - 4) == '1') {

R2\_3.setText("1");

R2\_3.setBackground(Color.red);

} else {

R2\_3.setText("0");

R2\_3.setBackground(Color.white);

}

if (R2\_binary.length() >= 5 && R2\_binary.charAt(R2\_binary.length() - 5) == '1') {

R2\_4.setText("1");

R2\_4.setBackground(Color.red);

} else {

R2\_4.setText("0");

R2\_4.setBackground(Color.white);

}

if (R2\_binary.length() >= 6 && R2\_binary.charAt(R2\_binary.length() - 6) == '1') {

R2\_5.setText("1");

R2\_5.setBackground(Color.red);

} else {

R2\_5.setText("0");

R2\_5.setBackground(Color.white);

}

if (R2\_binary.length() >= 7 && R2\_binary.charAt(R2\_binary.length() - 7) == '1') {

R2\_6.setText("1");

R2\_6.setBackground(Color.red);

} else {

R2\_6.setText("0");

R2\_6.setBackground(Color.white);

}

if (R2\_binary.length() >= 8 && R2\_binary.charAt(R2\_binary.length() - 8) == '1') {

R2\_7.setText("1");

R2\_7.setBackground(Color.red);

} else {

R2\_7.setText("0");

R2\_7.setBackground(Color.white);

}

if (R2\_binary.length() >= 9 && R2\_binary.charAt(R2\_binary.length() - 9) == '1') {

R2\_8.setText("1");

R2\_8.setBackground(Color.red);

} else {

R2\_8.setText("0");

R2\_8.setBackground(Color.white);

}

if (R2\_binary.length() >= 10 && R2\_binary.charAt(R2\_binary.length() - 10) == '1') {

R2\_9.setText("1");

R2\_9.setBackground(Color.red);

} else {

R2\_9.setText("0");

R2\_9.setBackground(Color.white);

}

if (R2\_binary.length() >= 11 && R2\_binary.charAt(R2\_binary.length() - 11) == '1') {

R2\_10.setText("1");

R2\_10.setBackground(Color.red);

} else {

R2\_10.setText("0");

R2\_10.setBackground(Color.white);

}

if (R2\_binary.length() >= 12 && R2\_binary.charAt(R2\_binary.length() - 12) == '1') {

R2\_11.setText("1");

R2\_11.setBackground(Color.red);

} else {

R2\_11.setText("0");

R2\_11.setBackground(Color.white);

}

if (R2\_binary.length() >= 13 && R2\_binary.charAt(R2\_binary.length() - 13) == '1') {

R2\_12.setText("1");

R2\_12.setBackground(Color.red);

} else {

R2\_12.setText("0");

R2\_12.setBackground(Color.white);

}

if (R2\_binary.length() >= 14 && R2\_binary.charAt(R2\_binary.length() - 14) == '1') {

R2\_13.setText("1");

R2\_13.setBackground(Color.red);

} else {

R2\_13.setText("0");

R2\_13.setBackground(Color.white);

}

if (R2\_binary.length() >= 15 && R2\_binary.charAt(R2\_binary.length() - 15) == '1') {

R2\_14.setText("1");

R2\_14.setBackground(Color.red);

} else {

R2\_14.setText("0");

R2\_14.setBackground(Color.white);

}

if (R2\_binary.length() >= 16 && R2\_binary.charAt(R2\_binary.length() - 16) == '1') {

R2\_15.setText("1");

R2\_15.setBackground(Color.red);

} else {

R2\_15.setText("0");

R2\_15.setBackground(Color.white);

}

}

public void Populate\_R3() {//Populate 16 bits of R3

String R3\_binary = Integer.toBinaryString(R3);

if (R3\_binary.length() >= 1 && R3\_binary.charAt(R3\_binary.length() - 1) == '1') {

R3\_0.setText("1");

R3\_0.setBackground(Color.red);

} else {

R3\_0.setText("0");

R3\_0.setBackground(Color.white);

}

if (R3\_binary.length() >= 2 && R3\_binary.charAt(R3\_binary.length() - 2) == '1') {

R3\_1.setText("1");

R3\_1.setBackground(Color.red);

} else {

R3\_1.setText("0");

R3\_1.setBackground(Color.white);

}

if (R3\_binary.length() >= 3 && R3\_binary.charAt(R3\_binary.length() - 3) == '1') {

R3\_2.setText("1");

R3\_2.setBackground(Color.red);

} else {

R3\_2.setText("0");

R3\_2.setBackground(Color.white);

}

if (R3\_binary.length() >= 4 && R3\_binary.charAt(R3\_binary.length() - 4) == '1') {

R3\_3.setText("1");

R3\_3.setBackground(Color.red);

} else {

R3\_3.setText("0");

R3\_3.setBackground(Color.white);

}

if (R3\_binary.length() >= 5 && R3\_binary.charAt(R3\_binary.length() - 5) == '1') {

R3\_4.setText("1");

R3\_4.setBackground(Color.red);

} else {

R3\_4.setText("0");

R3\_4.setBackground(Color.white);

}

if (R3\_binary.length() >= 6 && R3\_binary.charAt(R3\_binary.length() - 6) == '1') {

R3\_5.setText("1");

R3\_5.setBackground(Color.red);

} else {

R3\_5.setText("0");

R3\_5.setBackground(Color.white);

}

if (R3\_binary.length() >= 7 && R3\_binary.charAt(R3\_binary.length() - 7) == '1') {

R3\_6.setText("1");

R3\_6.setBackground(Color.red);

} else {

R3\_6.setText("0");

R3\_6.setBackground(Color.white);

}

if (R3\_binary.length() >= 8 && R3\_binary.charAt(R3\_binary.length() - 8) == '1') {

R3\_7.setText("1");

R3\_7.setBackground(Color.red);

} else {

R3\_7.setText("0");

R3\_7.setBackground(Color.white);

}

if (R3\_binary.length() >= 9 && R3\_binary.charAt(R3\_binary.length() - 9) == '1') {

R3\_8.setText("1");

R3\_8.setBackground(Color.red);

} else {

R3\_8.setText("0");

R3\_8.setBackground(Color.white);

}

if (R3\_binary.length() >= 10 && R3\_binary.charAt(R3\_binary.length() - 10) == '1') {

R3\_9.setText("1");

R3\_9.setBackground(Color.red);

} else {

R3\_9.setText("0");

R3\_9.setBackground(Color.white);

}

if (R3\_binary.length() >= 11 && R3\_binary.charAt(R3\_binary.length() - 11) == '1') {

R3\_10.setText("1");

R3\_10.setBackground(Color.red);

} else {

R3\_10.setText("0");

R3\_10.setBackground(Color.white);

}

if (R3\_binary.length() >= 12 && R3\_binary.charAt(R3\_binary.length() - 12) == '1') {

R3\_11.setText("1");

R3\_11.setBackground(Color.red);

} else {

R3\_11.setText("0");

R3\_11.setBackground(Color.white);

}

if (R3\_binary.length() >= 13 && R3\_binary.charAt(R3\_binary.length() - 13) == '1') {

R3\_12.setText("1");

R3\_12.setBackground(Color.red);

} else {

R3\_12.setText("0");

R3\_12.setBackground(Color.white);

}

if (R3\_binary.length() >= 14 && R3\_binary.charAt(R3\_binary.length() - 14) == '1') {

R3\_13.setText("1");

R3\_13.setBackground(Color.red);

} else {

R3\_13.setText("0");

R3\_13.setBackground(Color.white);

}

if (R3\_binary.length() >= 15 && R3\_binary.charAt(R3\_binary.length() - 15) == '1') {

R3\_14.setText("1");

R3\_14.setBackground(Color.red);

} else {

R3\_14.setText("0");

R3\_14.setBackground(Color.white);

}

if (R3\_binary.length() >= 16 && R3\_binary.charAt(R3\_binary.length() - 16) == '1') {

R3\_15.setText("1");

R3\_15.setBackground(Color.red);

} else {

R3\_15.setText("0");

R3\_15.setBackground(Color.white);

}

}

public void Populate\_X1() {//Populate 16 bits of X1

String X1\_binary = Integer.toBinaryString(X1);

if (X1\_binary.length() >= 1 && X1\_binary.charAt(X1\_binary.length() - 1) == '1') {

X1\_0.setText("1");

X1\_0.setBackground(Color.red);

} else {

X1\_0.setText("0");

X1\_0.setBackground(Color.white);

}

if (X1\_binary.length() >= 2 && X1\_binary.charAt(X1\_binary.length() - 2) == '1') {

X1\_1.setText("1");

X1\_1.setBackground(Color.red);

} else {

X1\_1.setText("0");

X1\_1.setBackground(Color.white);

}

if (X1\_binary.length() >= 3 && X1\_binary.charAt(X1\_binary.length() - 3) == '1') {

X1\_2.setText("1");

X1\_2.setBackground(Color.red);

} else {

X1\_2.setText("0");

X1\_2.setBackground(Color.white);

}

if (X1\_binary.length() >= 4 && X1\_binary.charAt(X1\_binary.length() - 4) == '1') {

X1\_3.setText("1");

X1\_3.setBackground(Color.red);

} else {

X1\_3.setText("0");

X1\_3.setBackground(Color.white);

}

if (X1\_binary.length() >= 5 && X1\_binary.charAt(X1\_binary.length() - 5) == '1') {

X1\_4.setText("1");

X1\_4.setBackground(Color.red);

} else {

X1\_4.setText("0");

X1\_4.setBackground(Color.white);

}

if (X1\_binary.length() >= 6 && X1\_binary.charAt(X1\_binary.length() - 6) == '1') {

X1\_5.setText("1");

X1\_5.setBackground(Color.red);

} else {

X1\_5.setText("0");

X1\_5.setBackground(Color.white);

}

if (X1\_binary.length() >= 7 && X1\_binary.charAt(X1\_binary.length() - 7) == '1') {

X1\_6.setText("1");

X1\_6.setBackground(Color.red);

} else {

X1\_6.setText("0");

X1\_6.setBackground(Color.white);

}

if (X1\_binary.length() >= 8 && X1\_binary.charAt(X1\_binary.length() - 8) == '1') {

X1\_7.setText("1");

X1\_7.setBackground(Color.red);

} else {

X1\_7.setText("0");

X1\_7.setBackground(Color.white);

}

if (X1\_binary.length() >= 9 && X1\_binary.charAt(X1\_binary.length() - 9) == '1') {

X1\_8.setText("1");

X1\_8.setBackground(Color.red);

} else {

X1\_8.setText("0");

X1\_8.setBackground(Color.white);

}

if (X1\_binary.length() >= 10 && X1\_binary.charAt(X1\_binary.length() - 10) == '1') {

X1\_9.setText("1");

X1\_9.setBackground(Color.red);

} else {

X1\_9.setText("0");

X1\_9.setBackground(Color.white);

}

if (X1\_binary.length() >= 11 && X1\_binary.charAt(X1\_binary.length() - 11) == '1') {

X1\_10.setText("1");

X1\_10.setBackground(Color.red);

} else {

X1\_10.setText("0");

X1\_10.setBackground(Color.white);

}

if (X1\_binary.length() >= 12 && X1\_binary.charAt(X1\_binary.length() - 12) == '1') {

X1\_11.setText("1");

X1\_11.setBackground(Color.red);

} else {

X1\_11.setText("0");

X1\_11.setBackground(Color.white);

}

if (X1\_binary.length() >= 13 && X1\_binary.charAt(X1\_binary.length() - 13) == '1') {

X1\_12.setText("1");

X1\_12.setBackground(Color.red);

} else {

X1\_12.setText("0");

X1\_12.setBackground(Color.white);

}

if (X1\_binary.length() >= 14 && X1\_binary.charAt(X1\_binary.length() - 14) == '1') {

X1\_13.setText("1");

X1\_13.setBackground(Color.red);

} else {

X1\_13.setText("0");

X1\_13.setBackground(Color.white);

}

if (X1\_binary.length() >= 15 && X1\_binary.charAt(X1\_binary.length() - 15) == '1') {

X1\_14.setText("1");

X1\_14.setBackground(Color.red);

} else {

X1\_14.setText("0");

X1\_14.setBackground(Color.white);

}

if (X1\_binary.length() >= 16 && X1\_binary.charAt(X1\_binary.length() - 16) == '1') {

X1\_15.setText("1");

X1\_15.setBackground(Color.red);

} else {

X1\_15.setText("0");

X1\_15.setBackground(Color.white);

}

}

public void Populate\_X2() {//Populate 16 bits of X2

String X2\_binary = Integer.toBinaryString(X2);

if (X2\_binary.length() >= 1 && X2\_binary.charAt(X2\_binary.length() - 1) == '1') {

X2\_0.setText("1");

X2\_0.setBackground(Color.red);

} else {

X2\_0.setText("0");

X2\_0.setBackground(Color.white);

}

if (X2\_binary.length() >= 2 && X2\_binary.charAt(X2\_binary.length() - 2) == '1') {

X2\_1.setText("1");

X2\_1.setBackground(Color.red);

} else {

X2\_1.setText("0");

X2\_1.setBackground(Color.white);

}

if (X2\_binary.length() >= 3 && X2\_binary.charAt(X2\_binary.length() - 3) == '1') {

X2\_2.setText("1");

X2\_2.setBackground(Color.red);

} else {

X2\_2.setText("0");

X2\_2.setBackground(Color.white);

}

if (X2\_binary.length() >= 4 && X2\_binary.charAt(X2\_binary.length() - 4) == '1') {

X2\_3.setText("1");

X2\_3.setBackground(Color.red);

} else {

X2\_3.setText("0");

X2\_3.setBackground(Color.white);

}

if (X2\_binary.length() >= 5 && X2\_binary.charAt(X2\_binary.length() - 5) == '1') {

X2\_4.setText("1");

X2\_4.setBackground(Color.red);

} else {

X2\_4.setText("0");

X2\_4.setBackground(Color.white);

}

if (X2\_binary.length() >= 6 && X2\_binary.charAt(X2\_binary.length() - 6) == '1') {

X2\_5.setText("1");

X2\_5.setBackground(Color.red);

} else {

X2\_5.setText("0");

X2\_5.setBackground(Color.white);

}

if (X2\_binary.length() >= 7 && X2\_binary.charAt(X2\_binary.length() - 7) == '1') {

X2\_6.setText("1");

X2\_6.setBackground(Color.red);

} else {

X2\_6.setText("0");

X2\_6.setBackground(Color.white);

}

if (X2\_binary.length() >= 8 && X2\_binary.charAt(X2\_binary.length() - 8) == '1') {

X2\_7.setText("1");

X2\_7.setBackground(Color.red);

} else {

X2\_7.setText("0");

X2\_7.setBackground(Color.white);

}

if (X2\_binary.length() >= 9 && X2\_binary.charAt(X2\_binary.length() - 9) == '1') {

X2\_8.setText("1");

X2\_8.setBackground(Color.red);

} else {

X2\_8.setText("0");

X2\_8.setBackground(Color.white);

}

if (X2\_binary.length() >= 10 && X2\_binary.charAt(X2\_binary.length() - 10) == '1') {

X2\_9.setText("1");

X2\_9.setBackground(Color.red);

} else {

X2\_9.setText("0");

X2\_9.setBackground(Color.white);

}

if (X2\_binary.length() >= 11 && X2\_binary.charAt(X2\_binary.length() - 11) == '1') {

X2\_10.setText("1");

X2\_10.setBackground(Color.red);

} else {

X2\_10.setText("0");

X2\_10.setBackground(Color.white);

}

if (X2\_binary.length() >= 12 && X2\_binary.charAt(X2\_binary.length() - 12) == '1') {

X2\_11.setText("1");

X2\_11.setBackground(Color.red);

} else {

X2\_11.setText("0");

X2\_11.setBackground(Color.white);

}

if (X2\_binary.length() >= 13 && X2\_binary.charAt(X2\_binary.length() - 13) == '1') {

X2\_12.setText("1");

X2\_12.setBackground(Color.red);

} else {

X2\_12.setText("0");

X2\_12.setBackground(Color.white);

}

if (X2\_binary.length() >= 14 && X2\_binary.charAt(X2\_binary.length() - 14) == '1') {

X2\_13.setText("1");

X2\_13.setBackground(Color.red);

} else {

X2\_13.setText("0");

X2\_13.setBackground(Color.white);

}

if (X2\_binary.length() >= 15 && X2\_binary.charAt(X2\_binary.length() - 15) == '1') {

X2\_14.setText("1");

X2\_14.setBackground(Color.red);

} else {

X2\_14.setText("0");

X2\_14.setBackground(Color.white);

}

if (X2\_binary.length() >= 16 && X2\_binary.charAt(X2\_binary.length() - 16) == '1') {

X2\_15.setText("1");

X2\_15.setBackground(Color.red);

} else {

X2\_15.setText("0");

X2\_15.setBackground(Color.white);

}

}

public void Populate\_X3() {//Populate 16 bits of X3

String X3\_binary = Integer.toBinaryString(X3);

if (X3\_binary.length() >= 1 && X3\_binary.charAt(X3\_binary.length() - 1) == '1') {

X3\_0.setText("1");

X3\_0.setBackground(Color.red);

} else {

X3\_0.setText("0");

X3\_0.setBackground(Color.white);

}

if (X3\_binary.length() >= 2 && X3\_binary.charAt(X3\_binary.length() - 2) == '1') {

X3\_1.setText("1");

X3\_1.setBackground(Color.red);

} else {

X3\_1.setText("0");

X3\_1.setBackground(Color.white);

}

if (X3\_binary.length() >= 3 && X3\_binary.charAt(X3\_binary.length() - 3) == '1') {

X3\_2.setText("1");

X3\_2.setBackground(Color.red);

} else {

X3\_2.setText("0");

X3\_2.setBackground(Color.white);

}

if (X3\_binary.length() >= 4 && X3\_binary.charAt(X3\_binary.length() - 4) == '1') {

X3\_3.setText("1");

X3\_3.setBackground(Color.red);

} else {

X3\_3.setText("0");

X3\_3.setBackground(Color.white);

}

if (X3\_binary.length() >= 5 && X3\_binary.charAt(X3\_binary.length() - 5) == '1') {

X3\_4.setText("1");

X3\_4.setBackground(Color.red);

} else {

X3\_4.setText("0");

X3\_4.setBackground(Color.white);

}

if (X3\_binary.length() >= 6 && X3\_binary.charAt(X3\_binary.length() - 6) == '1') {

X3\_5.setText("1");

X3\_5.setBackground(Color.red);

} else {

X3\_5.setText("0");

X3\_5.setBackground(Color.white);

}

if (X3\_binary.length() >= 7 && X3\_binary.charAt(X3\_binary.length() - 7) == '1') {

X3\_6.setText("1");

X3\_6.setBackground(Color.red);

} else {

X3\_6.setText("0");

X3\_6.setBackground(Color.white);

}

if (X3\_binary.length() >= 8 && X3\_binary.charAt(X3\_binary.length() - 8) == '1') {

X3\_7.setText("1");

X3\_7.setBackground(Color.red);

} else {

X3\_7.setText("0");

X3\_7.setBackground(Color.white);

}

if (X3\_binary.length() >= 9 && X3\_binary.charAt(X3\_binary.length() - 9) == '1') {

X3\_8.setText("1");

X3\_8.setBackground(Color.red);

} else {

X3\_8.setText("0");

X3\_8.setBackground(Color.white);

}

if (X3\_binary.length() >= 10 && X3\_binary.charAt(X3\_binary.length() - 10) == '1') {

X3\_9.setText("1");

X3\_9.setBackground(Color.red);

} else {

X3\_9.setText("0");

X3\_9.setBackground(Color.white);

}

if (X3\_binary.length() >= 11 && X3\_binary.charAt(X3\_binary.length() - 11) == '1') {

X3\_10.setText("1");

X3\_10.setBackground(Color.red);

} else {

X3\_10.setText("0");

X3\_10.setBackground(Color.white);

}

if (X3\_binary.length() >= 12 && X3\_binary.charAt(X3\_binary.length() - 12) == '1') {

X3\_11.setText("1");

X3\_11.setBackground(Color.red);

} else {

X3\_11.setText("0");

X3\_11.setBackground(Color.white);

}

if (X3\_binary.length() >= 13 && X3\_binary.charAt(X3\_binary.length() - 13) == '1') {

X3\_12.setText("1");

X3\_12.setBackground(Color.red);

} else {

X3\_12.setText("0");

X3\_12.setBackground(Color.white);

}

if (X3\_binary.length() >= 14 && X3\_binary.charAt(X3\_binary.length() - 14) == '1') {

X3\_13.setText("1");

X3\_13.setBackground(Color.red);

} else {

X3\_13.setText("0");

X3\_13.setBackground(Color.white);

}

if (X3\_binary.length() >= 15 && X3\_binary.charAt(X3\_binary.length() - 15) == '1') {

X3\_14.setText("1");

X3\_14.setBackground(Color.red);

} else {

X3\_14.setText("0");

X3\_14.setBackground(Color.white);

}

if (X3\_binary.length() >= 16 && X3\_binary.charAt(X3\_binary.length() - 16) == '1') {

X3\_15.setText("1");

X3\_15.setBackground(Color.red);

} else {

X3\_15.setText("0");

X3\_15.setBackground(Color.white);

}

}

public void ExecuteInstruction() throws InterruptedException { // Function to execute the instructions

System.out.println(PC);

Clear();

Populate\_PC();

IR = memory[PC];

System.out.println(IR);

Populate\_IR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

short flag\_GPR = 0, flag\_IXR = 0, flag\_I = 0;

//MAR = PC;

System.out.println(IR);

if (IR == 0) //Halt

{

MAR = PC;

Populate\_MAR();

for (short i = 0; i < 20; i++)

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Halt\_Display.setBackground(Color.red);

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

PC++;

Populate\_PC();

JPanel\_Run.setBackground(Color.black);

return;

}

String IR\_binary = Integer.toBinaryString(IR & 0xFFFF);

System.out.println(IR\_binary);

if (IR\_binary.length() == 32) {

IR\_binary = IR\_binary.substring(16, 32);

}

if (IR\_binary.length() <= 10) {

MBR = IR;

MAR = PC;

Populate\_MAR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MBR();

} else {

//GPR

if (IR\_binary.charAt(IR\_binary.length() - 10) == '0' && IR\_binary.charAt(IR\_binary.length() - 9) == '0') {

flag\_GPR = 1;

} else if (IR\_binary.charAt(IR\_binary.length() - 10) == '0' && IR\_binary.charAt(IR\_binary.length() - 9) == '1') {

flag\_GPR = 2;

} else if (IR\_binary.charAt(IR\_binary.length() - 10) == '1' && IR\_binary.charAt(IR\_binary.length() - 9) == '0') {

flag\_GPR = 3;

} else {

flag\_GPR = 4;

}

//IXR

if (IR\_binary.charAt(IR\_binary.length() - 8) == '0' && IR\_binary.charAt(IR\_binary.length() - 7) == '1') {

flag\_IXR = 1;

} else if (IR\_binary.charAt(IR\_binary.length() - 8) == '1' && IR\_binary.charAt(IR\_binary.length() - 7) == '0') {

flag\_IXR = 2;

} else if (IR\_binary.charAt(IR\_binary.length() - 8) == '1' && IR\_binary.charAt(IR\_binary.length() - 7) == '1') {

flag\_IXR = 3;

}

if (IR\_binary.charAt(IR\_binary.length() - 6) == '1') {

flag\_I = 1;

}

if (IR\_binary.length() == 11 && IR\_binary.charAt(0) == '1') {

System.out.println("LDR ");

//I = 0

if (flag\_I == 0) {

//No Indexing

if (flag\_IXR == 0) {

System.out.println("no Indexing I=0 ");

MAR = getAddress(IR);

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1 -> {

System.out.println("IX1 I=0 ");

MAR = (short) (getAddress(IR) + X1);

if (memoryOutofBoundCheck(MAR)) return;

}

case 2 -> {

System.out.println("IX2 I=0 ");

MAR = (short) (getAddress(IR) + X2);

if (memoryOutofBoundCheck(MAR)) return;

}

case 3 -> {

System.out.println("IX3 I=0 ");

MAR = (short) (getAddress(IR) + X3);

if (memoryOutofBoundCheck(MAR)) return;

}

default -> {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

switch (flag\_GPR) {

case 1 -> {

System.out.println("R0\n");

R0 = memory[MAR];

Populate\_R0();

}

case 2 -> {

System.out.println("R1\n");

R1 = memory[MAR];

Populate\_R1();

}

case 3 -> {

System.out.println("R2\n");

R2 = memory[MAR];

Populate\_R2();

}

case 4 -> {

System.out.println("R3\n");

R3 = memory[MAR];

Populate\_R3();

}

}

} else //I = 1

{

//No Indexing

if (flag\_IXR == 0) {

System.out.println("no Indexing I=1 ");

MAR = memory[getAddress(IR)];

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1:

System.out.println("IX1 I=1 ");

MAR = memory[(getAddress(IR) + X1)];

if (memoryOutofBoundCheck(MAR)) return;

break;

case 2:

System.out.println("IX2 I=1 ");

MAR = memory[getAddress(IR) + X2];

if (memoryOutofBoundCheck(MAR)) return;

break;

case 3:

System.out.println("IX3 I=1 ");

MAR = memory[getAddress(IR) + X3];

if (memoryOutofBoundCheck(MAR)) return;

break;

default: {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

switch (flag\_GPR) {

case 1 -> {

System.out.println("R0\n");

R0 = memory[MAR];

Populate\_R0();

}

case 2 -> {

System.out.println("R1\n");

R1 = memory[MAR];

Populate\_R1();

}

case 3 -> {

System.out.println("R2\n");

R2 = memory[MAR];

Populate\_R2();

}

case 4 -> {

System.out.println("R3\n");

R3 = memory[MAR];

Populate\_R3();

}

}

}

}

//STR

else if (IR\_binary.length() == 12 &&

IR\_binary.charAt(0) == '1' &&

IR\_binary.charAt(1) == '0') {

System.out.println("STR ");

//I = 0

if (flag\_I == 0) {

System.out.println("I=0 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("No Indexing ");

MAR = getAddress(IR);

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1 -> {

System.out.println("IX1 ");

MAR = (short) (getAddress(IR) + X1);

if (memoryOutofBoundCheck(MAR)) return;

}

case 2 -> {

System.out.println("IX2 ");

MAR = (short) (getAddress(IR) + X2);

if (memoryOutofBoundCheck(MAR)) return;

}

case 3 -> {

System.out.println("IX3 ");

MAR = (short) (getAddress(IR) + X3);

if (memoryOutofBoundCheck(MAR)) return;

}

default -> {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

switch (flag\_GPR) {

case 1 -> {

System.out.println("R0\n");

memory[MAR] = R0;

}

case 2 -> {

System.out.println("R1\n");

memory[MAR] = R1;

}

case 3 -> {

System.out.println("R2\n");

memory[MAR] = R2;

}

case 4 -> {

System.out.println("R3\n");

memory[MAR] = R3;

}

}

MBR = memory[MAR];

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MBR();

} else //I = 1

{

System.out.println("I=1 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("no indexing ");

MAR = memory[getAddress(IR)];

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1:

MAR = memory[getAddress(IR) + X1];

System.out.println("IX1 ");

if (memoryOutofBoundCheck(MAR)) return;

case 2:

MAR = memory[getAddress(IR) + X2];

System.out.println("IX2 ");

if (memoryOutofBoundCheck(MAR)) return;

case 3:

MAR = memory[getAddress(IR) + X3];

System.out.println("IX3 ");

if (memoryOutofBoundCheck(MAR)) return;

default: {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

switch (flag\_GPR) {

case 1 -> {

System.out.println("R0\n");

memory[MAR] = R0;

}

case 2 -> {

System.out.println("R1\n");

memory[MAR] = R1;

}

case 3 -> {

System.out.println("R2\n");

memory[MAR] = R2;

}

case 4 -> {

System.out.println("R3\n");

memory[MAR] = R3;

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

}

}

//LDA

else if (IR\_binary.length() == 12 &&

IR\_binary.charAt(0) == '1' &&

IR\_binary.charAt(1) == '1') {

//I = 0

System.out.println("LDA ");

if (flag\_I == 0) {

System.out.println("I=0 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("no indexing ");

MAR = getAddress(IR);

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1 -> {

MAR = (short) (getAddress(IR) + X1);

if (memoryOutofBoundCheck(MAR)) return;

System.out.println("IX1 ");

}

case 2 -> {

MAR = (short) (getAddress(IR) + X2);

if (memoryOutofBoundCheck(MAR)) return;

System.out.println("IX2 ");

}

case 3 -> {

MAR = (short) (getAddress(IR) + X3);

if (memoryOutofBoundCheck(MAR)) return;

System.out.println("IX3 ");

}

default -> {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

switch (flag\_GPR) {

case 1 -> {

R0 = MAR;

Populate\_R0();

System.out.println("R0\n");

}

case 2 -> {

R1 = MAR;

Populate\_R1();

System.out.println("R1\n");

}

case 3 -> {

R2 = MAR;

Populate\_R2();

System.out.println("R2\n");

}

case 4 -> {

R3 = MAR;

Populate\_R3();

System.out.println("R3\n");

}

}

} else //I = 1

{

System.out.println("I=1 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("No Indexing ");

MAR = memory[getAddress(IR)];

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1:

MAR = memory[getAddress(IR) + X1];

System.out.println("IX1 ");

if (memoryOutofBoundCheck(MAR)) return;

case 2:

MAR = memory[getAddress(IR) + X2];

System.out.println("IX2 ");

if (memoryOutofBoundCheck(MAR)) return;

case 3:

MAR = memory[getAddress(IR) + X3];

System.out.println("IX3 ");

if (memoryOutofBoundCheck(MAR)) return;

default: {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

switch (flag\_GPR) {

case 1 -> {

R0 = MAR;

Populate\_R0();

System.out.println("R0\n");

}

case 2 -> {

R1 = MAR;

Populate\_R1();

System.out.println("R1\n");

}

case 3 -> {

R2 = MAR;

Populate\_R2();

System.out.println("R2\n");

}

case 4 -> {

R3 = MAR;

Populate\_R3();

System.out.println("R3\n");

}

}

}

}

//LDX

else if (IR\_binary.length() == 16 &&

IR\_binary.charAt(0) == '1' &&

IR\_binary.charAt(1) == '0' &&

IR\_binary.charAt(2) == '1' &&

IR\_binary.charAt(3) == '0' &&

IR\_binary.charAt(4) == '0' &&

IR\_binary.charAt(5) == '1') {

System.out.println("LDX ");

//I = 0

if (flag\_I == 0) {

System.out.println("I=0 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("No Indexing ");

MAR = getAddress(IR);

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1 -> {

MAR = (short) (getAddress(IR));

if (memoryOutofBoundCheck(MAR)) return;

System.out.println("IX1\n");

}

case 2 -> {

MAR = (short) (getAddress(IR));

if (memoryOutofBoundCheck(MAR)) return;

System.out.println("IX2\n");

}

case 3 -> {

MAR = (short) (getAddress(IR));

if (memoryOutofBoundCheck(MAR)) return;

System.out.println("IX3\n");

}

default -> {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

MBR = memory[MAR];

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MBR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

switch (flag\_IXR) {

case 1 -> {

X1 = MBR;

Populate\_X1();

}

case 2 -> {

X2 = MBR;

Populate\_X2();

}

case 3 -> {

X3 = MBR;

Populate\_X3();

}

}

} else //I = 1

{

System.out.println("I=1 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("No Indexing ");

MAR = memory[getAddress(IR)];

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1:

MAR = memory[getAddress(IR)];

System.out.println("IX1\n");

if (memoryOutofBoundCheck(MAR)) return;

case 2:

MAR = memory[getAddress(IR)];

System.out.println("IX2\n");

if (memoryOutofBoundCheck(MAR)) return;

case 3:

MAR = memory[getAddress(IR)];

System.out.println("IX3\n");

if (memoryOutofBoundCheck(MAR)) return;

default: {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

switch (flag\_IXR) {

case 1 -> {

X1 = MAR;

Populate\_X1();

}

case 2 -> {

X2 = MAR;

Populate\_X2();

}

case 3 -> {

X3 = MAR;

Populate\_X3();

}

}

}

}

//STX

else if (IR\_binary.length() == 16 &&

IR\_binary.charAt(0) == '1' &&

IR\_binary.charAt(1) == '0' &&

IR\_binary.charAt(2) == '1' &&

IR\_binary.charAt(3) == '0' &&

IR\_binary.charAt(4) == '1' &&

IR\_binary.charAt(5) == '0') {

System.out.println("STX ");

//I = 0

if (flag\_I == 0) {

System.out.println("I=0 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("No Indexing ");

MAR = getAddress(IR);

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1 -> {

MAR = (short) (getAddress(IR) + X1);

System.out.println("IX1\n");

if (memoryOutofBoundCheck(MAR)) return;

}

case 2 -> {

MAR = (short) (getAddress(IR) + X2);

System.out.println("IX2\n");

if (memoryOutofBoundCheck(MAR)) return;

}

case 3 -> {

MAR = (short) (getAddress(IR)+3);

System.out.println("IX3\n");

if (memoryOutofBoundCheck(MAR)) return;

}

default -> {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

switch (flag\_IXR) {

case 1 -> {

memory[MAR] = X1;

}

case 2 -> {

memory[MAR] = X2;

}

case 3 -> {

memory[MAR] = X3;

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

} else //I = 1

{

System.out.println("I=1 ");

//No Indexing

if (flag\_IXR == 0) {

System.out.println("No Indexing ");

MAR = memory[getAddress(IR)];

if (memoryOutofBoundCheck(MAR)) return;

} else //Indexed

{

switch (flag\_IXR) {

case 1:

MAR = memory[getAddress(IR) + X1];

System.out.println("IX1\n");

if (memoryOutofBoundCheck(MAR)) return;

case 2:

MAR = memory[getAddress(IR) + X2];

System.out.println("IX2\n");

if (memoryOutofBoundCheck(MAR)) return;

case 3:

MAR = memory[getAddress(IR) + X3];

System.out.println("IX3\n");

if (memoryOutofBoundCheck(MAR)) return;

default: {

}

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

Populate\_MAR();

switch (flag\_IXR) {

case 1 -> {

memory[MAR] = X1;

}

case 2 -> {

memory[MAR] = X2;

}

case 3 -> {

memory[MAR] = X3;

}

}

try {

Thread.sleep(10);

} catch (InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

MBR = memory[MAR];

Populate\_MBR();

}

}

}

try {

Thread.sleep(10);

} catch (

InterruptedException ex) {

Logger.getLogger(ComputerSimulatorGroup9.class.getName()).log(Level.SEVERE, null, ex);

}

PC++;

Populate\_PC();

}

public void Populate\_MAR() { //Populate 16 bits of MAR

String MAR\_binary = Integer.toBinaryString(MAR);

if (MAR\_binary.length() >= 1 && MAR\_binary.charAt(MAR\_binary.length() - 1) == '1') {

MAR\_0.setText("1");

MAR\_0.setBackground(Color.red);

} else {

MAR\_0.setText("0");

MAR\_0.setBackground(Color.white);

}

if (MAR\_binary.length() >= 2 && MAR\_binary.charAt(MAR\_binary.length() - 2) == '1') {

MAR\_1.setText("1");

MAR\_1.setBackground(Color.red);

} else {

MAR\_1.setText("0");

MAR\_1.setBackground(Color.white);

}

if (MAR\_binary.length() >= 3 && MAR\_binary.charAt(MAR\_binary.length() - 3) == '1') {

MAR\_2.setText("1");

MAR\_2.setBackground(Color.red);

} else {

MAR\_2.setText("0");

MAR\_2.setBackground(Color.white);

}

if (MAR\_binary.length() >= 4 && MAR\_binary.charAt(MAR\_binary.length() - 4) == '1') {

MAR\_3.setText("1");

MAR\_3.setBackground(Color.red);

} else {

MAR\_3.setText("0");

MAR\_3.setBackground(Color.white);

}

if (MAR\_binary.length() >= 5 && MAR\_binary.charAt(MAR\_binary.length() - 5) == '1') {

MAR\_4.setText("1");

MAR\_4.setBackground(Color.red);

} else {

MAR\_4.setText("0");

MAR\_4.setBackground(Color.white);

}

if (MAR\_binary.length() >= 6 && MAR\_binary.charAt(MAR\_binary.length() - 6) == '1') {

MAR\_5.setText("1");

MAR\_5.setBackground(Color.red);

} else {

MAR\_5.setText("0");

MAR\_5.setBackground(Color.white);

}

if (MAR\_binary.length() >= 7 && MAR\_binary.charAt(MAR\_binary.length() - 7) == '1') {

MAR\_6.setText("1");

MAR\_6.setBackground(Color.red);

} else {

MAR\_6.setText("0");

MAR\_6.setBackground(Color.white);

}

if (MAR\_binary.length() >= 8 && MAR\_binary.charAt(MAR\_binary.length() - 8) == '1') {

MAR\_7.setText("1");

MAR\_7.setBackground(Color.red);

} else {

MAR\_7.setText("0");

MAR\_7.setBackground(Color.white);

}

if (MAR\_binary.length() >= 9 && MAR\_binary.charAt(MAR\_binary.length() - 9) == '1') {

MAR\_8.setText("1");

MAR\_8.setBackground(Color.red);

} else {

MAR\_8.setText("0");

MAR\_8.setBackground(Color.white);

}

if (MAR\_binary.length() >= 10 && MAR\_binary.charAt(MAR\_binary.length() - 10) == '1') {

MAR\_9.setText("1");

MAR\_9.setBackground(Color.red);

} else {

MAR\_9.setText("0");

MAR\_9.setBackground(Color.white);

}

if (MAR\_binary.length() >= 11 && MAR\_binary.charAt(MAR\_binary.length() - 11) == '1') {

MAR\_10.setText("1");

MAR\_10.setBackground(Color.red);

} else {

MAR\_10.setText("0");

MAR\_10.setBackground(Color.white);

}

if (MAR\_binary.length() >= 12 && MAR\_binary.charAt(MAR\_binary.length() - 12) == '1') {

MAR\_11.setText("1");

MAR\_11.setBackground(Color.red);

} else {

MAR\_11.setText("0");

MAR\_11.setBackground(Color.white);

}

}

public void Populate\_IR() { //Populate 16 bits of IR

String IR\_binary = Integer.toBinaryString(IR);

if (IR\_binary.length() >= 1 && IR\_binary.charAt(IR\_binary.length() - 1) == '1') {

IR\_0.setText("1");

IR\_0.setBackground(Color.red);

} else {

IR\_0.setText("0");

IR\_0.setBackground(Color.white);

}

if (IR\_binary.length() >= 2 && IR\_binary.charAt(IR\_binary.length() - 2) == '1') {

IR\_1.setText("1");

IR\_1.setBackground(Color.red);

} else {

IR\_1.setText("0");

IR\_1.setBackground(Color.white);

}

if (IR\_binary.length() >= 3 && IR\_binary.charAt(IR\_binary.length() - 3) == '1') {

IR\_2.setText("1");

IR\_2.setBackground(Color.red);

} else {

IR\_2.setText("0");

IR\_2.setBackground(Color.white);

}

if (IR\_binary.length() >= 4 && IR\_binary.charAt(IR\_binary.length() - 4) == '1') {

IR\_3.setText("1");

IR\_3.setBackground(Color.red);

} else {

IR\_3.setText("0");

IR\_3.setBackground(Color.white);

}

if (IR\_binary.length() >= 5 && IR\_binary.charAt(IR\_binary.length() - 5) == '1') {

IR\_4.setText("1");

IR\_4.setBackground(Color.red);

} else {

IR\_4.setText("0");

IR\_4.setBackground(Color.white);

}

if (IR\_binary.length() >= 6 && IR\_binary.charAt(IR\_binary.length() - 6) == '1') {

IR\_5.setText("1");

IR\_5.setBackground(Color.red);

} else {

IR\_5.setText("0");

IR\_5.setBackground(Color.white);

}

if (IR\_binary.length() >= 7 && IR\_binary.charAt(IR\_binary.length() - 7) == '1') {

IR\_6.setText("1");

IR\_6.setBackground(Color.red);

} else {

IR\_6.setText("0");

IR\_6.setBackground(Color.white);

}

if (IR\_binary.length() >= 8 && IR\_binary.charAt(IR\_binary.length() - 8) == '1') {

IR\_7.setText("1");

IR\_7.setBackground(Color.red);

} else {

IR\_7.setText("0");

IR\_7.setBackground(Color.white);

}

if (IR\_binary.length() >= 9 && IR\_binary.charAt(IR\_binary.length() - 9) == '1') {

IR\_8.setText("1");

IR\_8.setBackground(Color.red);

} else {

IR\_8.setText("0");

IR\_8.setBackground(Color.white);

}

if (IR\_binary.length() >= 10 && IR\_binary.charAt(IR\_binary.length() - 10) == '1') {

IR\_9.setText("1");

IR\_9.setBackground(Color.red);

} else {

IR\_9.setText("0");

IR\_9.setBackground(Color.white);

}

if (IR\_binary.length() >= 11 && IR\_binary.charAt(IR\_binary.length() - 11) == '1') {

IR\_10.setText("1");

IR\_10.setBackground(Color.red);

} else {

IR\_10.setText("0");

IR\_10.setBackground(Color.white);

}

if (IR\_binary.length() >= 12 && IR\_binary.charAt(IR\_binary.length() - 12) == '1') {

IR\_11.setText("1");

IR\_11.setBackground(Color.red);

} else {

IR\_11.setText("0");

IR\_11.setBackground(Color.white);

}

if (IR\_binary.length() >= 13 && IR\_binary.charAt(IR\_binary.length() - 13) == '1') {

IR\_12.setText("1");

IR\_12.setBackground(Color.red);

} else {

IR\_12.setText("0");

IR\_12.setBackground(Color.white);

}

if (IR\_binary.length() >= 14 && IR\_binary.charAt(IR\_binary.length() - 14) == '1') {

IR\_13.setText("1");

IR\_13.setBackground(Color.red);

} else {

IR\_13.setText("0");

IR\_13.setBackground(Color.white);

}

if (IR\_binary.length() >= 15 && IR\_binary.charAt(IR\_binary.length() - 15) == '1') {

IR\_14.setText("1");

IR\_14.setBackground(Color.red);

} else {

IR\_14.setText("0");

IR\_14.setBackground(Color.white);

}

if (IR\_binary.length() >= 16 && IR\_binary.charAt(IR\_binary.length() - 16) == '1') {

IR\_15.setText("1");

IR\_15.setBackground(Color.red);

} else {

IR\_15.setText("0");

IR\_15.setBackground(Color.white);

}

}

public void Reset\_Toggle\_Instruction() { //Reset Input Instruction Toggle Buttons to Initial value

OP\_15.setSelected(false);

OP\_14.setSelected(false);

OP\_13.setSelected(false);

OP\_12.setSelected(false);

OP\_11.setSelected(false);

OP\_10.setSelected(false);

GPR\_9.setSelected(false);

GPR\_8.setSelected(false);

IXR\_7.setSelected(false);

IXR\_6.setSelected(false);

I\_5.setSelected(false);

ADDRESS\_4.setSelected(false);

ADDRESS\_3.setSelected(false);

ADDRESS\_2.setSelected(false);

ADDRESS\_1.setSelected(false);

ADDRESS\_0.setSelected(false);

OP\_15.setBackground(Color.white);

OP\_14.setBackground(Color.white);

OP\_13.setBackground(Color.white);

I\_5.setBackground(Color.white);

OP\_12.setBackground(Color.white);

OP\_11.setBackground(Color.white);

OP\_10.setBackground(Color.white);

GPR\_9.setBackground(Color.white);

GPR\_8.setBackground(Color.white);

IXR\_7.setBackground(Color.white);

IXR\_6.setBackground(Color.white);

ADDRESS\_4.setBackground(Color.white);

ADDRESS\_3.setBackground(Color.white);

ADDRESS\_2.setBackground(Color.white);

ADDRESS\_1.setBackground(Color.white);

ADDRESS\_0.setBackground(Color.white);

}

@SuppressWarnings("unchecked")

private void initComponents() {

f = new JFrame("Group 9 | Computer Simulator");

f.setSize(4000, 4000);

JPanelMain = new JPanel();

JPanel\_IR = new JPanel();

JLabel\_IR = new JLabel();

IR\_15 = new JButton();

IR\_14 = new JButton();

IR\_13 = new JButton();

IR\_12 = new JButton();

IR\_11 = new JButton();

IR\_10 = new JButton();

IR\_9 = new JButton();

IR\_8 = new JButton();

IR\_7 = new JButton();

IR\_6 = new JButton();

IR\_5 = new JButton();

IR\_3 = new JButton();

IR\_4 = new JButton();

IR\_2 = new JButton();

IR\_1 = new JButton();

IR\_0 = new JButton();

Load\_IR = new JButton();

JPanel\_R0 = new JPanel();

JLabel\_R0 = new JLabel();

R0\_15 = new JButton();

R0\_14 = new JButton();

R0\_13 = new JButton();

R0\_12 = new JButton();

R0\_11 = new JButton();

R0\_10 = new JButton();

R0\_9 = new JButton();

R0\_8 = new JButton();

R0\_7 = new JButton();

R0\_6 = new JButton();

R0\_5 = new JButton();

R0\_3 = new JButton();

R0\_4 = new JButton();

R0\_2 = new JButton();

R0\_1 = new JButton();

R0\_0 = new JButton();

Load\_R0 = new JButton();

JPanel\_R1 = new JPanel();

JLabel\_R1 = new JLabel();

R1\_15 = new JButton();

R1\_14 = new JButton();

R1\_13 = new JButton();

R1\_12 = new JButton();

R1\_11 = new JButton();

R1\_10 = new JButton();

R1\_9 = new JButton();

R1\_8 = new JButton();

R1\_7 = new JButton();

R1\_6 = new JButton();

R1\_5 = new JButton();

R1\_3 = new JButton();

R1\_4 = new JButton();

R1\_2 = new JButton();

R1\_1 = new JButton();

R1\_0 = new JButton();

Load\_R1 = new JButton();

JPanel\_R2 = new JPanel();

JLabel\_R2 = new JLabel();

R2\_15 = new JButton();

R2\_14 = new JButton();

R2\_13 = new JButton();

R2\_12 = new JButton();

R2\_11 = new JButton();

R2\_10 = new JButton();

R2\_9 = new JButton();

R2\_8 = new JButton();

R2\_7 = new JButton();

R2\_6 = new JButton();

R2\_5 = new JButton();

R2\_3 = new JButton();

R2\_4 = new JButton();

R2\_2 = new JButton();

R2\_1 = new JButton();

R2\_0 = new JButton();

Load\_R2 = new JButton();

JPanel\_R3 = new JPanel();

JLabel\_R3 = new JLabel();

R3\_15 = new JButton();

R3\_14 = new JButton();

R3\_13 = new JButton();

R3\_12 = new JButton();

R3\_11 = new JButton();

R3\_10 = new JButton();

R3\_9 = new JButton();

R3\_8 = new JButton();

R3\_7 = new JButton();

R3\_6 = new JButton();

R3\_5 = new JButton();

R3\_3 = new JButton();

R3\_4 = new JButton();

R3\_2 = new JButton();

R3\_1 = new JButton();

R3\_0 = new JButton();

Load\_R3 = new JButton();

JPanel\_PC = new JPanel();

JLabel\_PC = new JLabel();

PC\_11 = new JButton();

PC\_10 = new JButton();

PC\_9 = new JButton();

PC\_8 = new JButton();

PC\_7 = new JButton();

PC\_6 = new JButton();

PC\_5 = new JButton();

PC\_4 = new JButton();

PC\_3 = new JButton();

PC\_2 = new JButton();

PC\_1 = new JButton();

PC\_0 = new JButton();

Load\_PC = new JButton();

JPanel\_MAR = new JPanel();

JLabel\_MAR = new JLabel();

MAR\_11 = new JButton();

MAR\_10 = new JButton();

MAR\_9 = new JButton();

MAR\_8 = new JButton();

MAR\_7 = new JButton();

MAR\_6 = new JButton();

MAR\_5 = new JButton();

MAR\_4 = new JButton();

MAR\_3 = new JButton();

MAR\_2 = new JButton();

MAR\_1 = new JButton();

MAR\_0 = new JButton();

Load\_MAR = new JButton();

JPanel\_MBR = new JPanel();

JLabel\_MBR = new JLabel();

MBR\_15 = new JButton();

MBR\_14 = new JButton();

MBR\_13 = new JButton();

MBR\_12 = new JButton();

MBR\_11 = new JButton();

MBR\_10 = new JButton();

MBR\_9 = new JButton();

MBR\_8 = new JButton();

MBR\_7 = new JButton();

MBR\_6 = new JButton();

MBR\_5 = new JButton();

MBR\_3 = new JButton();

MBR\_4 = new JButton();

MBR\_2 = new JButton();

MBR\_1 = new JButton();

MBR\_0 = new JButton();

Load\_MBR = new JButton();

JPanel\_INSTRUCTION = new JPanel();

JPanel\_OP = new JPanel();

OP\_15 = new JToggleButton();

OP\_14 = new JToggleButton();

OP\_13 = new JToggleButton();

OP\_12 = new JToggleButton();

OP\_11 = new JToggleButton();

OP\_10 = new JToggleButton();

JPanel\_GPR = new JPanel();

GPR\_9 = new JToggleButton();

GPR\_8 = new JToggleButton();

JPanel\_IXR = new JPanel();

IXR\_7 = new JToggleButton();

IXR\_6 = new JToggleButton();

JPanel\_I = new JPanel();

I\_5 = new JToggleButton();

JPanel\_ADDRESS = new JPanel();

ADDRESS\_4 = new JToggleButton();

ADDRESS\_3 = new JToggleButton();

ADDRESS\_2 = new JToggleButton();

ADDRESS\_1 = new JToggleButton();

ADDRESS\_0 = new JToggleButton();

JPanel\_OP\_Main = new JPanel();

JPanel\_I\_Main = new JPanel();

JPanel\_GPR\_Main = new JPanel();

JPanel\_ADDRESS\_Main = new JPanel();

JPanel\_IXR\_Main = new JPanel();

JPanel\_X1 = new JPanel();

JPanel\_X2 = new JPanel();

JPanel\_X3 = new JPanel();

JLabel\_X1 = new JLabel();

JLabel\_X2 = new JLabel();

JLabel\_X3 = new JLabel();

X1\_15 = new JButton();

X1\_14 = new JButton();

X1\_13 = new JButton();

X1\_12 = new JButton();

X1\_11 = new JButton();

X1\_10 = new JButton();

X1\_9 = new JButton();

X1\_8 = new JButton();

X1\_7 = new JButton();

X1\_6 = new JButton();

X1\_5 = new JButton();

X1\_4 = new JButton();

X1\_3 = new JButton();

X1\_2 = new JButton();

X1\_1 = new JButton();

X1\_0 = new JButton();

Load\_X1 = new JButton();

X2\_12 = new JButton();

X2\_3 = new JButton();

X2\_2 = new JButton();

X2\_15 = new JButton();

X2\_5 = new JButton();

Load\_X2 = new JButton();

X2\_7 = new JButton();

X2\_14 = new JButton();

X2\_6 = new JButton();

X2\_8 = new JButton();

X2\_4 = new JButton();

X2\_1 = new JButton();

X2\_10 = new JButton();

X2\_13 = new JButton();

X2\_11 = new JButton();

X2\_9 = new JButton();

X3\_12 = new JButton();

X3\_3 = new JButton();

X3\_2 = new JButton();

X3\_15 = new JButton();

X3\_5 = new JButton();

Load\_X3 = new JButton();

X3\_7 = new JButton();

X3\_14 = new JButton();

X3\_6 = new JButton();

X3\_8 = new JButton();

X3\_4 = new JButton();

X3\_1 = new JButton();

X3\_10 = new JButton();

X3\_13 = new JButton();

X3\_11 = new JButton();

X3\_9 = new JButton();

X2\_0 = new JButton();

X3\_0 = new JButton();

JPanel\_MFR = new JPanel();

JLabel\_MFR = new JLabel();

MFR\_3 = new JButton();

MFR\_2 = new JButton();

MFR\_1 = new JButton();

MFR\_0 = new JButton();

JPanel\_CC = new JPanel();

JLabel\_CC = new JLabel();

CC\_3 = new JButton();

CC\_2 = new JButton();

CC\_1 = new JButton();

CC\_0 = new JButton();

Store = new JButton();

Load = new JButton();

Clear\_All = new JButton();

Init = new JButton();

Custom\_Init = new JButton();

SS = new JButton();

Run = new JButton();

JPanel\_Func = new JPanel();

JPanel\_OPT = new JPanel();

JPanel\_Init = new JPanel();

JLabel\_Halt = new JLabel();

JLabel\_Run = new JLabel();

Halt\_Display = new JPanel();

JPanel\_Run = new JPanel();

JPanel\_Display = new JPanel();

R0\_0.setBackground(Color.white);

R0\_1.setBackground(Color.white);

R0\_2.setBackground(Color.white);

R0\_3.setBackground(Color.white);

R0\_4.setBackground(Color.white);

R0\_5.setBackground(Color.white);

R0\_6.setBackground(Color.white);

R0\_7.setBackground(Color.white);

R0\_8.setBackground(Color.white);

R0\_9.setBackground(Color.white);

R0\_10.setBackground(Color.white);

R0\_11.setBackground(Color.white);

R0\_12.setBackground(Color.white);

R0\_13.setBackground(Color.white);

R0\_14.setBackground(Color.white);

R0\_15.setBackground(Color.white);

R0\_0.setBackground(Color.white);

R0\_1.setBackground(Color.white);

R0\_2.setBackground(Color.white);

R0\_3.setBackground(Color.white);

R0\_4.setBackground(Color.white);

R0\_5.setBackground(Color.white);

R0\_6.setBackground(Color.white);

R0\_7.setBackground(Color.white);

R0\_8.setBackground(Color.white);

R0\_9.setBackground(Color.white);

R0\_10.setBackground(Color.white);

R0\_11.setBackground(Color.white);

R0\_12.setBackground(Color.white);

R0\_13.setBackground(Color.white);

R0\_14.setBackground(Color.white);

R0\_15.setBackground(Color.white);

R1\_0.setBackground(Color.white);

R1\_1.setBackground(Color.white);

R1\_2.setBackground(Color.white);

R1\_3.setBackground(Color.white);

R1\_4.setBackground(Color.white);

R1\_5.setBackground(Color.white);

R1\_6.setBackground(Color.white);

R1\_7.setBackground(Color.white);

R1\_8.setBackground(Color.white);

R1\_9.setBackground(Color.white);

R1\_10.setBackground(Color.white);

R1\_11.setBackground(Color.white);

R1\_12.setBackground(Color.white);

R1\_13.setBackground(Color.white);

R1\_14.setBackground(Color.white);

R1\_15.setBackground(Color.white);

R2\_0.setBackground(Color.white);

R2\_1.setBackground(Color.white);

R2\_2.setBackground(Color.white);

R2\_3.setBackground(Color.white);

R2\_4.setBackground(Color.white);

R2\_5.setBackground(Color.white);

R2\_6.setBackground(Color.white);

R2\_7.setBackground(Color.white);

R2\_8.setBackground(Color.white);

R2\_9.setBackground(Color.white);

R2\_10.setBackground(Color.white);

R2\_11.setBackground(Color.white);

R2\_12.setBackground(Color.white);

R2\_13.setBackground(Color.white);

R2\_14.setBackground(Color.white);

R2\_15.setBackground(Color.white);

R3\_0.setBackground(Color.white);

R3\_1.setBackground(Color.white);

R3\_2.setBackground(Color.white);

R3\_3.setBackground(Color.white);

R3\_4.setBackground(Color.white);

R3\_5.setBackground(Color.white);

R3\_6.setBackground(Color.white);

R3\_7.setBackground(Color.white);

R3\_8.setBackground(Color.white);

R3\_9.setBackground(Color.white);

R3\_10.setBackground(Color.white);

R3\_11.setBackground(Color.white);

R3\_12.setBackground(Color.white);

R3\_13.setBackground(Color.white);

R3\_14.setBackground(Color.white);

R3\_15.setBackground(Color.white);

X1\_0.setBackground(Color.white);

X1\_1.setBackground(Color.white);

X1\_2.setBackground(Color.white);

X1\_3.setBackground(Color.white);

X1\_4.setBackground(Color.white);

X1\_5.setBackground(Color.white);

X1\_6.setBackground(Color.white);

X1\_7.setBackground(Color.white);

X1\_8.setBackground(Color.white);

X1\_9.setBackground(Color.white);

X1\_10.setBackground(Color.white);

X1\_11.setBackground(Color.white);

X1\_12.setBackground(Color.white);

X1\_13.setBackground(Color.white);

X1\_14.setBackground(Color.white);

X1\_15.setBackground(Color.white);

X2\_0.setBackground(Color.white);

X2\_1.setBackground(Color.white);

X2\_2.setBackground(Color.white);

X2\_3.setBackground(Color.white);

X2\_4.setBackground(Color.white);

X2\_5.setBackground(Color.white);

X2\_6.setBackground(Color.white);

X2\_7.setBackground(Color.white);

X2\_8.setBackground(Color.white);

X2\_9.setBackground(Color.white);

X2\_10.setBackground(Color.white);

X2\_11.setBackground(Color.white);

X2\_12.setBackground(Color.white);

X2\_13.setBackground(Color.white);

X2\_14.setBackground(Color.white);

X2\_15.setBackground(Color.white);

X3\_0.setBackground(Color.white);

X3\_1.setBackground(Color.white);

X3\_2.setBackground(Color.white);

X3\_3.setBackground(Color.white);

X3\_4.setBackground(Color.white);

X3\_5.setBackground(Color.white);

X3\_6.setBackground(Color.white);

X3\_7.setBackground(Color.white);

X3\_8.setBackground(Color.white);

X3\_9.setBackground(Color.white);

X3\_10.setBackground(Color.white);

X3\_11.setBackground(Color.white);

X3\_12.setBackground(Color.white);

X3\_13.setBackground(Color.white);

X3\_14.setBackground(Color.white);

X3\_15.setBackground(Color.white);

OP\_15.setBackground(Color.white);

OP\_14.setBackground(Color.white);

OP\_13.setBackground(Color.white);

I\_5.setBackground(Color.white);

OP\_12.setBackground(Color.white);

OP\_11.setBackground(Color.white);

OP\_10.setBackground(Color.white);

GPR\_9.setBackground(Color.white);

GPR\_8.setBackground(Color.white);

IXR\_7.setBackground(Color.white);

IXR\_6.setBackground(Color.white);

ADDRESS\_4.setBackground(Color.white);

ADDRESS\_3.setBackground(Color.white);

ADDRESS\_2.setBackground(Color.white);

ADDRESS\_1.setBackground(Color.white);

ADDRESS\_0.setBackground(Color.white);

PC\_0.setBackground(Color.white);

PC\_1.setBackground(Color.white);

PC\_2.setBackground(Color.white);

PC\_3.setBackground(Color.white);

PC\_4.setBackground(Color.white);

PC\_5.setBackground(Color.white);

PC\_6.setBackground(Color.white);

PC\_7.setBackground(Color.white);

PC\_8.setBackground(Color.white);

PC\_9.setBackground(Color.white);

PC\_10.setBackground(Color.white);

PC\_11.setBackground(Color.white);

MAR\_0.setBackground(Color.white);

MAR\_1.setBackground(Color.white);

MAR\_2.setBackground(Color.white);

MAR\_3.setBackground(Color.white);

MAR\_4.setBackground(Color.white);

MAR\_5.setBackground(Color.white);

MAR\_6.setBackground(Color.white);

MAR\_7.setBackground(Color.white);

MAR\_8.setBackground(Color.white);

MAR\_9.setBackground(Color.white);

MAR\_10.setBackground(Color.white);

MAR\_11.setBackground(Color.white);

MBR\_0.setBackground(Color.white);

MBR\_1.setBackground(Color.white);

MBR\_2.setBackground(Color.white);

MBR\_3.setBackground(Color.white);

MBR\_4.setBackground(Color.white);

MBR\_5.setBackground(Color.white);

MBR\_6.setBackground(Color.white);

MBR\_7.setBackground(Color.white);

MBR\_8.setBackground(Color.white);

MBR\_9.setBackground(Color.white);

MBR\_10.setBackground(Color.white);

MBR\_11.setBackground(Color.white);

MBR\_12.setBackground(Color.white);

MBR\_13.setBackground(Color.white);

MBR\_14.setBackground(Color.white);

MBR\_15.setBackground(Color.white);

IR\_0.setBackground(Color.white);

IR\_1.setBackground(Color.white);

IR\_2.setBackground(Color.white);

IR\_3.setBackground(Color.white);

IR\_4.setBackground(Color.white);

IR\_5.setBackground(Color.white);

IR\_6.setBackground(Color.white);

IR\_7.setBackground(Color.white);

IR\_8.setBackground(Color.white);

IR\_9.setBackground(Color.white);

IR\_10.setBackground(Color.white);

IR\_11.setBackground(Color.white);

IR\_12.setBackground(Color.white);

IR\_13.setBackground(Color.white);

IR\_14.setBackground(Color.white);

IR\_15.setBackground(Color.white);

MFR\_0.setBackground(Color.white);

MFR\_1.setBackground(Color.white);

MFR\_2.setBackground(Color.white);

MFR\_3.setBackground(Color.white);

CC\_0.setBackground(Color.white);

CC\_1.setBackground(Color.white);

CC\_2.setBackground(Color.white);

CC\_3.setBackground(Color.white);

JLabel\_IR.setText("IR");

JPanel\_IR.add(JLabel\_IR);

IR\_15.setText("0");

JPanel\_IR.add(IR\_15);

IR\_14.setText("0");

JPanel\_IR.add(IR\_14);

IR\_13.setText("0");

JPanel\_IR.add(IR\_13);

IR\_12.setText("0");

JPanel\_IR.add(IR\_12);

IR\_11.setText("0");

JPanel\_IR.add(IR\_11);

IR\_10.setText("0");

JPanel\_IR.add(IR\_10);

IR\_9.setText("0");

JPanel\_IR.add(IR\_9);

IR\_8.setText("0");

JPanel\_IR.add(IR\_8);

IR\_7.setText("0");

JPanel\_IR.add(IR\_7);

IR\_6.setText("0");

JPanel\_IR.add(IR\_6);

IR\_5.setText("0");

JPanel\_IR.add(IR\_5);

IR\_4.setText("0");

JPanel\_IR.add(IR\_4);

IR\_3.setText("0");

JPanel\_IR.add(IR\_3);

IR\_2.setText("0");

JPanel\_IR.add(IR\_2);

IR\_1.setText("0");

JPanel\_IR.add(IR\_1);

IR\_0.setText("0");

JPanel\_IR.add(IR\_0);

Load\_IR.setText("LD");

Load\_IR.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_IRActionPerformed(evt);

}

});

JPanel\_IR.add(Load\_IR);

JLabel\_R0.setText("R0");

JPanel\_R0.add(JLabel\_R0);

R0\_15.setText("0");

JPanel\_R0.add(R0\_15);

R0\_14.setText("0");

JPanel\_R0.add(R0\_14);

R0\_13.setText("0");

JPanel\_R0.add(R0\_13);

R0\_12.setText("0");

JPanel\_R0.add(R0\_12);

R0\_11.setText("0");

JPanel\_R0.add(R0\_11);

R0\_10.setText("0");

JPanel\_R0.add(R0\_10);

R0\_9.setText("0");

JPanel\_R0.add(R0\_9);

R0\_8.setText("0");

JPanel\_R0.add(R0\_8);

R0\_7.setText("0");

JPanel\_R0.add(R0\_7);

R0\_6.setText("0");

JPanel\_R0.add(R0\_6);

R0\_5.setText("0");

JPanel\_R0.add(R0\_5);

R0\_4.setText("0");

JPanel\_R0.add(R0\_4);

R0\_3.setText("0");

JPanel\_R0.add(R0\_3);

R0\_2.setText("0");

JPanel\_R0.add(R0\_2);

R0\_1.setText("0");

JPanel\_R0.add(R0\_1);

R0\_0.setText("0");

JPanel\_R0.add(R0\_0);

Load\_R0.setText("LD");

Load\_R0.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_R0ActionPerformed(evt);

}

});

JPanel\_R0.add(Load\_R0);

JLabel\_R1.setText("R1");

JPanel\_R1.add(JLabel\_R1);

R1\_15.setText("0");

JPanel\_R1.add(R1\_15);

R1\_14.setText("0");

JPanel\_R1.add(R1\_14);

R1\_13.setText("0");

JPanel\_R1.add(R1\_13);

R1\_12.setText("0");

JPanel\_R1.add(R1\_12);

R1\_11.setText("0");

JPanel\_R1.add(R1\_11);

R1\_10.setText("0");

JPanel\_R1.add(R1\_10);

R1\_9.setText("0");

JPanel\_R1.add(R1\_9);

R1\_8.setText("0");

JPanel\_R1.add(R1\_8);

R1\_7.setText("0");

JPanel\_R1.add(R1\_7);

R1\_6.setText("0");

JPanel\_R1.add(R1\_6);

R1\_5.setText("0");

JPanel\_R1.add(R1\_5);

R1\_4.setText("0");

JPanel\_R1.add(R1\_4);

R1\_3.setText("0");

JPanel\_R1.add(R1\_3);

R1\_2.setText("0");

JPanel\_R1.add(R1\_2);

R1\_1.setText("0");

JPanel\_R1.add(R1\_1);

R1\_0.setText("0");

JPanel\_R1.add(R1\_0);

Load\_R1.setText("LD");

Load\_R1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_R1ActionPerformed(evt);

}

});

JPanel\_R1.add(Load\_R1);

JLabel\_R2.setText("R2");

JPanel\_R2.add(JLabel\_R2);

R2\_15.setText("0");

JPanel\_R2.add(R2\_15);

R2\_14.setText("0");

JPanel\_R2.add(R2\_14);

R2\_13.setText("0");

JPanel\_R2.add(R2\_13);

R2\_12.setText("0");

JPanel\_R2.add(R2\_12);

R2\_11.setText("0");

JPanel\_R2.add(R2\_11);

R2\_10.setText("0");

JPanel\_R2.add(R2\_10);

R2\_9.setText("0");

JPanel\_R2.add(R2\_9);

R2\_8.setText("0");

JPanel\_R2.add(R2\_8);

R2\_7.setText("0");

JPanel\_R2.add(R2\_7);

R2\_6.setText("0");

JPanel\_R2.add(R2\_6);

R2\_5.setText("0");

JPanel\_R2.add(R2\_5);

R2\_4.setText("0");

JPanel\_R2.add(R2\_4);

R2\_3.setText("0");

JPanel\_R2.add(R2\_3);

R2\_2.setText("0");

JPanel\_R2.add(R2\_2);

R2\_1.setText("0");

JPanel\_R2.add(R2\_1);

R2\_0.setText("0");

JPanel\_R2.add(R2\_0);

Load\_R2.setText("LD");

Load\_R2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_R2ActionPerformed(evt);

}

});

JPanel\_R2.add(Load\_R2);

JLabel\_R3.setText("R3");

JPanel\_R3.add(JLabel\_R3);

R3\_15.setText("0");

JPanel\_R3.add(R3\_15);

R3\_14.setText("0");

JPanel\_R3.add(R3\_14);

R3\_13.setText("0");

JPanel\_R3.add(R3\_13);

R3\_12.setText("0");

JPanel\_R3.add(R3\_12);

R3\_11.setText("0");

JPanel\_R3.add(R3\_11);

R3\_10.setText("0");

JPanel\_R3.add(R3\_10);

R3\_9.setText("0");

JPanel\_R3.add(R3\_9);

R3\_8.setText("0");

JPanel\_R3.add(R3\_8);

R3\_7.setText("0");

JPanel\_R3.add(R3\_7);

R3\_6.setText("0");

JPanel\_R3.add(R3\_6);

R3\_5.setText("0");

JPanel\_R3.add(R3\_5);

R3\_4.setText("0");

JPanel\_R3.add(R3\_4);

R3\_3.setText("0");

JPanel\_R3.add(R3\_3);

R3\_2.setText("0");

JPanel\_R3.add(R3\_2);

R3\_1.setText("0");

JPanel\_R3.add(R3\_1);

R3\_0.setText("0");

JPanel\_R3.add(R3\_0);

Load\_R3.setText("LD");

Load\_R3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_R3ActionPerformed(evt);

}

});

JPanel\_R3.add(Load\_R3);

JLabel\_PC.setText("PC");

JPanel\_PC.add(JLabel\_PC);

PC\_11.setText("0");

JPanel\_PC.add(PC\_11);

PC\_10.setText("0");

JPanel\_PC.add(PC\_10);

PC\_9.setText("0");

JPanel\_PC.add(PC\_9);

PC\_8.setText("0");

JPanel\_PC.add(PC\_8);

PC\_7.setText("0");

JPanel\_PC.add(PC\_7);

PC\_6.setText("0");

JPanel\_PC.add(PC\_6);

PC\_5.setText("0");

JPanel\_PC.add(PC\_5);

PC\_4.setText("0");

JPanel\_PC.add(PC\_4);

PC\_3.setText("0");

JPanel\_PC.add(PC\_3);

PC\_2.setText("0");

JPanel\_PC.add(PC\_2);

PC\_1.setText("0");

JPanel\_PC.add(PC\_1);

PC\_0.setText("0");

JPanel\_PC.add(PC\_0);

Load\_PC.setText("LD");

Load\_PC.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_PCActionPerformed(evt);

}

});

JPanel\_PC.add(Load\_PC);

JLabel\_MAR.setText("MAR");

JPanel\_MAR.add(JLabel\_MAR);

MAR\_11.setText("0");

JPanel\_MAR.add(MAR\_11);

MAR\_10.setText("0");

JPanel\_MAR.add(MAR\_10);

MAR\_9.setText("0");

JPanel\_MAR.add(MAR\_9);

MAR\_8.setText("0");

JPanel\_MAR.add(MAR\_8);

MAR\_7.setText("0");

JPanel\_MAR.add(MAR\_7);

MAR\_6.setText("0");

JPanel\_MAR.add(MAR\_6);

MAR\_5.setText("0");

JPanel\_MAR.add(MAR\_5);

MAR\_4.setText("0");

JPanel\_MAR.add(MAR\_4);

MAR\_3.setText("0");

JPanel\_MAR.add(MAR\_3);

MAR\_2.setText("0");

JPanel\_MAR.add(MAR\_2);

MAR\_1.setText("0");

JPanel\_MAR.add(MAR\_1);

MAR\_0.setText("0");

JPanel\_MAR.add(MAR\_0);

Load\_MAR.setText("LD");

Load\_MAR.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_MARActionPerformed(evt);

}

});

JPanel\_MAR.add(Load\_MAR);

JLabel\_MBR.setText("MBR");

JPanel\_MBR.add(JLabel\_MBR);

MBR\_15.setText("0");

JPanel\_MBR.add(MBR\_15);

MBR\_14.setText("0");

JPanel\_MBR.add(MBR\_14);

MBR\_13.setText("0");

JPanel\_MBR.add(MBR\_13);

MBR\_12.setText("0");

JPanel\_MBR.add(MBR\_12);

MBR\_11.setText("0");

JPanel\_MBR.add(MBR\_11);

MBR\_10.setText("0");

JPanel\_MBR.add(MBR\_10);

MBR\_9.setText("0");

JPanel\_MBR.add(MBR\_9);

MBR\_8.setText("0");

JPanel\_MBR.add(MBR\_8);

MBR\_7.setText("0");

JPanel\_MBR.add(MBR\_7);

MBR\_6.setText("0");

JPanel\_MBR.add(MBR\_6);

MBR\_5.setText("0");

JPanel\_MBR.add(MBR\_5);

MBR\_4.setText("0");

JPanel\_MBR.add(MBR\_4);

MBR\_3.setText("0");

JPanel\_MBR.add(MBR\_3);

MBR\_2.setText("0");

JPanel\_MBR.add(MBR\_2);

MBR\_1.setText("0");

JPanel\_MBR.add(MBR\_1);

MBR\_0.setText("0");

JPanel\_MBR.add(MBR\_0);

Load\_MBR.setText("LD");

Load\_MBR.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_MBRActionPerformed(evt);

}

});

JPanel\_MBR.add(Load\_MBR);

OP\_15.setText("15");

OP\_15.setDebugGraphicsOptions(DebugGraphics.NONE\_OPTION);

OP\_15.setIconTextGap(2);

JPanel\_OP.add(OP\_15);

OP\_14.setText("14");

OP\_14.setIconTextGap(2);

JPanel\_OP.add(OP\_14);

OP\_13.setText("13");

OP\_13.setIconTextGap(2);

JPanel\_OP.add(OP\_13);

OP\_12.setText("12");

OP\_12.setIconTextGap(2);

JPanel\_OP.add(OP\_12);

OP\_11.setText("11");

OP\_11.setIconTextGap(2);

JPanel\_OP.add(OP\_11);

OP\_10.setText("10");

OP\_10.setIconTextGap(2);

JPanel\_OP.add(OP\_10);

GPR\_9.setText(" 9 ");

GPR\_9.setIconTextGap(2);

JPanel\_GPR.add(GPR\_9);

GPR\_8.setText(" 8 ");

GPR\_8.setIconTextGap(2);

JPanel\_GPR.add(GPR\_8);

IXR\_7.setText(" 7 ");

IXR\_7.setIconTextGap(2);

JPanel\_IXR.add(IXR\_7);

IXR\_6.setText(" 6 ");

IXR\_6.setIconTextGap(2);

JPanel\_IXR.add(IXR\_6);

I\_5.setText(" 5 ");

I\_5.setIconTextGap(2);

JPanel\_I.add(I\_5);

ADDRESS\_4.setText(" 4 ");

ADDRESS\_4.setIconTextGap(2);

JPanel\_ADDRESS.add(ADDRESS\_4);

ADDRESS\_3.setText(" 3 ");

ADDRESS\_3.setIconTextGap(2);

JPanel\_ADDRESS.add(ADDRESS\_3);

ADDRESS\_2.setText(" 2 ");

ADDRESS\_2.setIconTextGap(2);

JPanel\_ADDRESS.add(ADDRESS\_2);

ADDRESS\_1.setText(" 1 ");

ADDRESS\_1.setIconTextGap(2);

JPanel\_ADDRESS.add(ADDRESS\_1);

ADDRESS\_0.setText(" 0 ");

ADDRESS\_0.setIconTextGap(2);

JPanel\_ADDRESS.add(ADDRESS\_0);

JPanel\_OP\_Main.setLayout(new GridLayout(2, 1));

JPanel\_OP\_Main.add(JPanel\_OP);

JPanel\_OP\_Main.add(new JLabel("Operation"));

JPanel\_GPR\_Main.setLayout(new GridLayout(2, 1));

JPanel\_GPR\_Main.add(JPanel\_GPR);

JPanel\_GPR\_Main.add(new JLabel("GPR"));

JPanel\_IXR\_Main.setLayout(new GridLayout(2, 1));

JPanel\_IXR\_Main.add(JPanel\_IXR);

JPanel\_IXR\_Main.add(new JLabel("IXR"));

JPanel\_I\_Main.setLayout(new GridLayout(2, 1));

JPanel\_I\_Main.add(JPanel\_I);

JPanel\_I\_Main.add(new JLabel("I"));

JPanel\_ADDRESS\_Main.setLayout(new GridLayout(2, 1));

JPanel\_ADDRESS\_Main.add(JPanel\_ADDRESS);

JPanel\_ADDRESS\_Main.add(new JLabel("Address"));

JLabel\_X1.setText("X1");

JPanel\_X1.add(JLabel\_X1);

X1\_15.setText("0");

JPanel\_X1.add(X1\_15);

X1\_14.setText("0");

JPanel\_X1.add(X1\_14);

X1\_13.setText("0");

JPanel\_X1.add(X1\_13);

X1\_12.setText("0");

JPanel\_X1.add(X1\_12);

X1\_11.setText("0");

JPanel\_X1.add(X1\_11);

X1\_10.setText("0");

JPanel\_X1.add(X1\_10);

X1\_9.setText("0");

JPanel\_X1.add(X1\_9);

X1\_8.setText("0");

JPanel\_X1.add(X1\_8);

X1\_7.setText("0");

JPanel\_X1.add(X1\_7);

X1\_6.setText("0");

JPanel\_X1.add(X1\_6);

X1\_5.setText("0");

JPanel\_X1.add(X1\_5);

X1\_4.setText("0");

JPanel\_X1.add(X1\_4);

X1\_3.setText("0");

JPanel\_X1.add(X1\_3);

X1\_2.setText("0");

JPanel\_X1.add(X1\_2);

X1\_1.setText("0");

JPanel\_X1.add(X1\_1);

X1\_0.setText("0");

JPanel\_X1.add(X1\_0);

Load\_X1.setText("LD");

Load\_X1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_X1ActionPerformed(evt);

}

});

JPanel\_X1.add(Load\_X1);

JLabel\_X2.setText("X2");

JPanel\_X2.add(JLabel\_X2);

X2\_15.setText("0");

JPanel\_X2.add(X2\_15);

X2\_14.setText("0");

JPanel\_X2.add(X2\_14);

X2\_13.setText("0");

JPanel\_X2.add(X2\_13);

X2\_12.setText("0");

JPanel\_X2.add(X2\_12);

X2\_11.setText("0");

JPanel\_X2.add(X2\_11);

X2\_10.setText("0");

JPanel\_X2.add(X2\_10);

X2\_9.setText("0");

JPanel\_X2.add(X2\_9);

X2\_8.setText("0");

JPanel\_X2.add(X2\_8);

X2\_7.setText("0");

JPanel\_X2.add(X2\_7);

X2\_6.setText("0");

JPanel\_X2.add(X2\_6);

X2\_5.setText("0");

JPanel\_X2.add(X2\_5);

X2\_4.setText("0");

JPanel\_X2.add(X2\_4);

X2\_3.setText("0");

JPanel\_X2.add(X2\_3);

X2\_2.setText("0");

JPanel\_X2.add(X2\_2);

X2\_1.setText("0");

JPanel\_X2.add(X2\_1);

X2\_0.setText("0");

JPanel\_X2.add(X2\_0);

Load\_X2.setText("LD");

Load\_X2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_X2ActionPerformed(evt);

}

});

JPanel\_X2.add(Load\_X2);

JLabel\_X3.setText("X3");

JPanel\_X3.add(JLabel\_X3);

X3\_15.setText("0");

JPanel\_X3.add(X3\_15);

X3\_14.setText("0");

JPanel\_X3.add(X3\_14);

X3\_13.setText("0");

JPanel\_X3.add(X3\_13);

X3\_12.setText("0");

JPanel\_X3.add(X3\_12);

X3\_11.setText("0");

JPanel\_X3.add(X3\_11);

X3\_10.setText("0");

JPanel\_X3.add(X3\_10);

X3\_9.setText("0");

JPanel\_X3.add(X3\_9);

X3\_8.setText("0");

JPanel\_X3.add(X3\_8);

X3\_7.setText("0");

JPanel\_X3.add(X3\_7);

X3\_6.setText("0");

JPanel\_X3.add(X3\_6);

X3\_5.setText("0");

JPanel\_X3.add(X3\_5);

X3\_4.setText("0");

JPanel\_X3.add(X3\_4);

X3\_3.setText("0");

JPanel\_X3.add(X3\_3);

X3\_2.setText("0");

JPanel\_X3.add(X3\_2);

X3\_1.setText("0");

JPanel\_X3.add(X3\_1);

X3\_0.setText("0");

JPanel\_X3.add(X3\_0);

Load\_X3.setText("LD");

Load\_X3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Load\_X3ActionPerformed(evt);

}

});

JPanel\_X3.add(Load\_X3);

JLabel\_MFR.setText("MFR");

JPanel\_MFR.add(JLabel\_MFR);

MFR\_3.setText("0");

JPanel\_MFR.add(MFR\_3);

MFR\_2.setText("0");

JPanel\_MFR.add(MFR\_2);

MFR\_1.setText("0");

JPanel\_MFR.add(MFR\_1);

MFR\_0.setText("0");

JPanel\_MFR.add(MFR\_0);

JLabel\_CC.setText("CC");

JPanel\_CC.add(JLabel\_CC);

CC\_3.setText("0");

JPanel\_CC.add(CC\_3);

CC\_2.setText("0");

JPanel\_CC.add(CC\_2);

CC\_1.setText("0");

JPanel\_CC.add(CC\_1);

CC\_0.setText("0");

JPanel\_CC.add(CC\_0);

Store.setText("Store");

Store.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

StoreActionPerformed(evt);

}

});

JPanel\_Func.add(Store);

Load.setText("Load");

Load.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

LoadActionPerformed(evt);

}

});

JPanel\_Func.add(Load);

Clear\_All.setText("Clear All");

Clear\_All.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Clear\_AllActionPerformed(evt);

}

});

JPanel\_Func.add(Clear\_All);

Init.setText("Init");

Init.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

InitActionPerformed(evt);

}

});

JPanel\_Init.add(Init);

Custom\_Init.setText("Custom Init");

Custom\_Init.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

Custom\_InitActionPerformed(evt);

}

});

JPanel\_Init.add(Custom\_Init);

SS.setText("SS");

SS.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

SSActionPerformed(evt);

}

});

JPanel\_OPT.add(SS);

Run.setText("Run");

Run.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

RunActionPerformed(evt);

}

});

JPanel\_OPT.add(Run);

G\_Layout = new GridLayout(2, 2);

G\_Layout.setVgap(10);

G\_Layout.setHgap(10);

JPanel\_Display.setLayout(G\_Layout);

JLabel\_Halt.setText("Halt");

JPanel\_Display.add(JLabel\_Halt);

Halt\_Display.setBackground(Color.black);

JPanel\_Display.add(Halt\_Display);

JLabel\_Run.setText("Run");

JPanel\_Display.add(JLabel\_Run);

JPanel\_Display.add(JLabel\_Run);

JPanel\_Run.setBackground(Color.black);

JPanel\_Display.add(JPanel\_Run);

JPanel\_INSTRUCTION.add(JPanel\_OP\_Main);

JPanel\_INSTRUCTION.add(JPanel\_GPR\_Main);

JPanel\_INSTRUCTION.add(JPanel\_IXR\_Main);

JPanel\_INSTRUCTION.add(JPanel\_I\_Main);

JPanel\_INSTRUCTION.add(JPanel\_ADDRESS\_Main);

JPanelMain.add(JPanel\_R0);

JPanelMain.add(JPanel\_R1);

JPanelMain.add(JPanel\_R2);

JPanelMain.add(JPanel\_R3);

JPanelMain.add(JPanel\_X1);

JPanelMain.add(JPanel\_X2);

JPanelMain.add(JPanel\_X3);

JPanelMain.add(JPanel\_IR);

JPanelMain.add(JPanel\_MBR);

JPanel\_Grid1 = new JPanel();

G\_Layout1 = new GridLayout(3, 1);

JPanel\_Grid1.setLayout(G\_Layout1);

JPanel\_Grid1.add(JPanel\_MAR);

JPanel\_Grid1.add(JPanel\_PC);

JPanel\_Grid1.add(JPanel\_MFR);

JPanel\_Grid1.add(JPanel\_CC);

JPanelMain.add(JPanel\_Grid1);

JPanelMain.add(JPanel\_INSTRUCTION);

JPanel\_Grid2 = new JPanel();

G\_Layout2 = new GridLayout(1, 3);

JPanel\_Grid2.setLayout(G\_Layout2);

JPanel\_Grid2.add(JPanel\_Func);

JPanel\_Grid2.add(JPanel\_Init);

JPanel\_Grid2.add(JPanel\_OPT);

JPanelMain.add(JPanel\_Grid2);

JPanelMain.add(JPanel\_Display);

f.add(JPanelMain);

f.setVisible(true);

}

public static void main(String[] args) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new ComputerSimulatorGroup9();

}

});

}

}