



الجمهورية العربية السورية

جامعة تشرين

كلية الهندسة المعلوماتية

السنة الثالثة

اسم المشروع :

تصميم دائرة خريطة كارنوف وكوين ماكسلوسكي

إعداد الطلاب

طارق سامر الجراح

علي ناصر أحمد

مؤيد جهاد علي

اسم المشرف:

د. محمد ياسين صبيح

للعام الدراسي 2017 - 2018

المحتويات

2	مقدمة:
2	KARNAUGH MAP (خريطة كارنوف):
4	خطوات الحل باستخدام كارنوف:
5	كارنوف بمتحولين:
7	كارنوف بأربع متحولات:
10	الكود:
26	خاتمة:

جدول الاشكال:

5	الشكل 1
6	الشكل 2
6	الشكل 3
7	الشكل 4
8	الشكل 5
9	الشكل 6

مقدمة:

نعلم أنه عند حل معادلات بوليانية ب n متغير نقوم بتبسيط هذه المعادلات إلى أبسط شكل ممكن.

مر معنا سابق عدة قوانين تستخدم في الجبر البوليني لتبسيط المعادلات ك قوانين الإبدال وقوانين الدمج في عملية الانفصال والاتصال وقاعدة المتمم للمتمم وقواعد الامتصاص الأولى والثانية ونظريتا دي-مورغان اللتان تعتبران من أهم النظريات في علم الجبر البوليني لكن يوجد طرق أخرى لتبسيط هذه المعادلات (البوابات) لشكل أبسط وبطريقة أبسط ف أحيانا يمكن تبسيط معادلة معقدة باستخدام الجبر البوليني بعشرة بوابات وتكون خطوات التبسيط معقدة جدا وبنفس الوقت يوجد طرق أسهل لفعل ذلك وبأقل من عدد البوابات السابق.

سندرس في بحثنا طريقتين جديدين لتبسيط المعادلات والبوابات المنطقية بشكل أسهل هما karnaugh map التي هي طريقة فعالة لتبسيط الاقترانات (المعادلات) لحد 6 متغيرات.

Karnaugh map (خريطة كارنوف):

خريطة Karnaugh (KM أو K-map) هي طريقة لتبسيط تعبير الجبر البوليني. عند استخدام خريطة كارنوف يتم نقل النتائج المنطقية المطلوبة من جدول الحقيقة إلى شبكة ثنائية الأبعاد

استخدامات خريطة كارنوف: في العديد من الدوائر الرقمية والمشاكل العملية نحتاج إلى إيجاد تعبير بأقل المتغيرات. يمكننا تقليل التعبيرات المنطقية من 3 أو 4 متغيرات بسهولة جدا باستخدام K-map دون

استخدام أي نظريات الجبر البولي. يمكن أن تأخذ K-map شكلين من مجموع المضاريب (SOP) ومضاريب المجموع (POS) وفقاً للحاجة إلى المشكلة.

سوف نستخدم في دراستنا ل خرائط كارنوف بيئة العمل eclipse 2018.



[illegible]

خطوات الحل باستخدام كارنوف:

1. في البداية نقوم برسم ال Gray code المعبر عن المعادلة

2. نحدد K-map وفقاً لعدد المتغيرات.

3. نقوم بملء خريطة كارنوف بالأصفار والواحدات وذلك وفقا لجدول كارنوف.

4. نقوم برسم مجموعات مستطيلة تحتوي على الواحدات فقط ولكن عدد الواحد يكون مرفوع للقوة 2 مثل

2,4,8 ... (باستثناء 1) وحاول تغطية أكبر عدد ممكن من العناصر في مجموعة واحدة.

5. نقوم باختزال كل بت مع متممه في كل سطر وفي كل عمود في المجموعة الواحدة وبعد ذلك ينتج

لدينا أبسط شكل للمعادلة بحسب عدد المجموعات المتبقية.

کارنوف بمتحوّلین:

ليكن لدينا المعادلة التالية:

$$F = xy' + xy + x'y + x'y'$$

الان لندخل المعادلة السابقة في البرنامج ونرى ماذا سيكون الخرج.

[illegible]

الشكل 1

[illegible]

6/26

B \ A	0	1
0	1	1
1	1	1

Y = 1

الشكل 4

وجدنا أن الخرج هو $Y=1$ وهذا صحيح أي يعني أنه تم تطبيق خوارزمية كارنوف ع المعادلة السابقة والان لنجرب على حالة أكبر ولتكن أربع متحولات.

كارنوف بأربع متحولات:

الان لنقوم بتطبيق خريطة كارنوف على أربع متحولات:

ليكن لدينا المعادلة التالية:

$$F = x'y'z'w' + x'y'z'w + x'y'zw' + x'y'zw + x'yz'w' + x'yz'w + x'yzw'$$

الآن لندخل المعادلة السابقة في البرنامج ونرى ماذا سيكون الخرج.

أولاً نختار عدد المتحولات والذي هو أربعة ثم نقوم بملء قيم كل حد من الحدود.

A	B	C	D	Y
0	0	0	0	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
1	0	0	0	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
0	1	0	0	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
0	0	1	0	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
0	0	0	1	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
1	0	1	0	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
1	0	0	1	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
1	1	0	0	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
0	0	1	1	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
0	1	0	1	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
0	1	1	0	<input checked="" type="radio"/> 1 <input type="radio"/> 0 <input type="radio"/> X
1	1	1	0	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
1	0	1	1	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
1	1	0	1	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
0	1	1	1	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X
1	1	1	1	<input type="radio"/> 1 <input checked="" type="radio"/> 0 <input type="radio"/> X

الشكل 5

الان كل ما علينا فعله هو ضغط زر ال Solve ولنرى الخرج.

[illegible]

$$Y = A'D' + A'C' + A'B'$$

الشكل 6

وهو خرج المعادلة بعد التبسيط.

الكود:

أثناء الحل وكتابة الواجهة نقوم باستيراد الحزم التي نحتاجها من خلال وضع مؤشر الماوس ع الغرض
/المتحول/.. الذي نريد استيراده ثم نضغط import وقد احتجنا الحزم التالية من أجل بناء الواجهة

```
1 import javafx.application.Application;
2 import javafx.collections.FXCollections;
3 import javafx.collections.ObservableList;
4 import javafx.geometry.Insets;
5 import javafx.geometry.Pos;
6 import javafx.scene.Scene;
7 import javafx.scene.control.Alert;
8 import javafx.scene.control.ChoiceBox;
9 import javafx.scene.control.TableColumn;
10 import javafx.scene.control.TableView;
11 import javafx.scene.control.TextField;
12 import javafx.scene.control.Tooltip;
13 import javafx.scene.control.Alert.AlertType;
14 import javafx.scene.control.ButtonType;
15 import javafx.scene.control.cell.PropertyValueFactory;
16 import javafx.scene.image.Image;
17 import javafx.scene.layout.GridPane;
18 import javafx.scene.layout.HBox;
19 import javafx.scene.layout.StackPane;
20 import javafx.scene.media.AudioClip;
21 import javafx.scene.media.Media;
22 import javafx.scene.media.MediaPlayer;
23 import javafx.scene.media.MediaPlayer.Status;
24 import javafx.scene.media.MediaView;
25 import javafx.stage.Stage;
26
```

الان لنبدأ بالصف momo:

```
public class momo extends Application {
    |
    StackPane root = new StackPane();
    GridPane root1 = new GridPane();
    TableView<Graytable> table = new TableView<>();
    ObservableList<Graytable> data = FXCollections.observableArrayList();
    TableView<k_map> table1 = new TableView<>();
    TableColumn< Graytable, Integer > lastColumn;
    ObservableList<k_map> data1 = FXCollections.observableArrayList();
    Stage stage;
    Scene scene;
    int x;
    StackPane root2 = new StackPane();
    TextField txt = new TextField();

    Scene scene1 = new Scene(root2);

    public void kmap(MediaPlayer Music)
    {
        GridPane root3 = new GridPane();
        root3.setVgap(30);

        //Media

        MediaPlayer video1 = new MediaPlayer( new Media(getClass().getResource("back2.mp4").toExternalForm()) );
        video1.setCycleCount( MediaPlayer.INDEFINITE );

        MediaView video3 = new MediaView( video1);
        video3.setPreserveRatio( false );
        video3.fitWidthProperty().bind( root3.widthProperty() );
        video3.fitHeightProperty().bind( root3.heightProperty() );
        video1.setAutoplay(true);

        ButtonPane MusicBtn1 = new ButtonPane("Music", new Image(getClass().getResource("1.png").toExternalForm()),
            new Image(getClass().getResource("2.png").toExternalForm()),
            new AudioClip(getClass().getResource("ButtonOnMouse.wav").toExternalForm()),
            new AudioClip(getClass().getResource("Button.wav").toExternalForm()));

        MusicBtn1.setOnAction(e -> {
            //...
        });
    }
}
```

استخدمنا التابع kmap من أجل بناء ال Scene الثاني عند ضغط زر الحل طبعا ويجب اختيار عدد المتحولات والا سوف يظهر لدينا إشارة خطر كما سوف نرى لاحقا واستخدمنا عدة صفوف من اجل عرض الفيديو والموسيقا في خلفية الواجهة وطبعا الازرار وكل ما هو موجود في الواجهة سيتم وضعة في Stack pane

```

MusicBtn1.setOnAction(e -> {
    video1.getOnPlaying();
    Music.play();
    if (video1.getStatus() == Status.PLAYING)
        Music.pause();
    else {
        Music.play();
    }
});
//Back

ButtonPane BackBtn = new ButtonPane("Back", new Image(getClass().getResource("solve2.png").toExternalForm()),
    new Image(getClass().getResource("solve1.png").toExternalForm()),
    new AudioClip(getClass().getResource("ButtonOnMouse.wav").toExternalForm()),
    new AudioClip(getClass().getResource("Button.wav").toExternalForm()));
BackBtn.setTextSize(14);

BackBtn.setOnAction(e -> {
    stage.setScene(scene);
});

HBox controlBox1 = new HBox(BackBtn, MusicBtn1);
controlBox1.setStyle("-fx-background-color: rgba(2, 0, 32, 0.5); -fx-background-radius: 25");
controlBox1.setPadding(new Insets(2, 10, 2, 10));
controlBox1.setSpacing(20);
controlBox1.setAlignment(Pos.CENTER);

//setColumn for K-map

root3.getChildren().addAll(controlBox1);
root3.setPadding(new Insets(20));
root2.getChildren().add( root3);
root3.add(table1, 0, 1);
Sovle.txt = txt;
root3.add(txt, 0, 2);
}

```

استخدمنا ال HBox لوضع الازرار الخاصة بالرجوع للواجهة الأول (قبل الحل) ولزر الموسيقى أيضا.

```

@Override
public void start( Stage stage ) {
    this.stage = stage;
    table1.setItems( data1 );
    root1.setVgap( 30 );

    //alert when we didn't choice the number of var

    Alert alert = new Alert( AlertType.ERROR, "Please, Choose the number of variables.", ButtonType.OK );
    // media

    MediaPlayer video = new MediaPlayer( new Media(getClass().getResource("back2.mp4").toExternalForm()) );
    video.setCycleCount( MediaPlayer.INDEFINITE );
    MediaPlayer Music = new MediaPlayer( new Media(getClass().getResource("music2.mp3").toExternalForm()) );
    Music.setCycleCount( MediaPlayer.INDEFINITE );
    MediaView video2 = new MediaView( video );
    video2.setPreserveRatio( false );
    video2.fitWidthProperty().bind( root.widthProperty() );
    video2.fitHeightProperty().bind( root.heightProperty() );
    video.setAutoPlay(true);
    Music.setAutoPlay(true);

    // musicButton

    ButtonPane MusicBtn = new ButtonPane("Music", new Image(getClass().getResource("1.png").toExternalForm()),
        new Image(getClass().getResource("2.png").toExternalForm()),
        new AudioClip(getClass().getResource("ButtonOnMouse.wav").toExternalForm()),
        new AudioClip(getClass().getResource("Button.wav").toExternalForm()));
    MusicBtn.setOnAction(e -> {
        video.play();
        Music.play();
        if (video.getStatus() == Status.PLAYING) {

            Music.pause();
        } else {

            Music.play();
        }
    });
    kmap(Music);
}

```

نفس الخطوات السابقة تقريبا لكن من أجل الواجهة الأولى مع ملاحظة أننا قمنا باستدعاء التابع Kmap هنا.

```

// SolveButton

ButtonPane SolveBtn = new ButtonPane("Solve", new Image(getClass().getResource("solve2.png").toExternalForm()),
    new Image(getClass().getResource("solve1.png").toExternalForm()),
    new AudioClip(getClass().getResource("ButtonOnMouse.wav").toExternalForm()),
    new AudioClip(getClass().getResource("Button.wav").toExternalForm()));
SolveBtn.setTextSize(14);

// chooseButton the variable number

ChoiceBox<Integer> choice = new ChoiceBox<>();
choice.getItems().addAll(2, 3, 4);
choice.setTooltip(new Tooltip("number of variables"));
SolveBtn.setOnAction(e -> {
    if ( choice.getValue() == null ) {
        alert.show();
        return;
    }

    fillrows( choice.getValue() );
    stage.setScene(scenel);
    fillmap();
    Sovle.solve( choice.getValue() );
});

// when we choice the variable number

choice.setOnAction(e -> {
    int nofv = choice.getValue();
    data.clear();
    table.refresh();
    table.getColumns().clear();
    char c = 'A';
    for (int i = 0; i < nofv; ++i) {

        // add the Column name like A , B ,C ,D

        TableColumn<Graytable, Integer> col = new TableColumn<>("" + c);
        col.setCellValueFactory(new PropertyValueFactory<>("x" + (i + 1)));
        table.getColumns().add(col);
    }
});

```

هنا لدينا زر ال Solve وزر الخيارات الخاص بعدد المتحولات ونقوم أيضا بتعبئة قيم ال Gray table الخاص بالمعادلة المدخلة.

```

        col.setMaxWidth(25);
        col.setMinWidth(25);
        ++c;
    }
    String[] k = { "B\\A", "0", "1", "00", "10", "11", "01" };
    table1.getColumns().clear();
    table1.getItems().clear();
    table1.setFixedCellSize(25);
    //table.getColumns().get(1).getCellData( 2 );
    int startindex=(nofv == 2 ? 0 : 2);
    if (nofv==2) {
        k[2]="1";
    }
    else if (nofv==3 )
        k[2]="C\\AB";
    else if (nofv==4) {
        k[2]="CD\\AB";
    }
    int numcolumn =(nofv == 2 ? 3:5);
    for (int i=0;i<numcolumn;i++)
    {
        TableColumn<k_map, String> col = new TableColumn<>(k[startindex + i ]);
        col.setCellValueFactory(new PropertyValueFactory<>("a" + (i + 1)));
        table1.getColumns().add(col);
        col.setSortable(false);
        col.setReorderable(false);
    }

    // add the Last Column (the input for each column)

    TableColumn<Graytable, Integer> tableColumn = new TableColumn<>("Y");
    tableColumn.setCellFactory(s -> new RadioButtonTableCell<>());
    tableColumn.setCellValueFactory(new PropertyValueFactory<>("x5"));
    tableColumn.setOnEditCommit(s -> {
        ((Graytable) s.getTableView().getItems().get(s.getTablePosition().getRow())).setXS(s.getNewValue());
    });

    table.getColumns().add(tableColumn);
    tableColumn.setSortable(false);
    tableColumn.setReorderable(false);
    testColumn = tableColumn;

```

متابعة وضع القيم في الجدول Gray Table


```

// add rows data for each Column

String[] a = { "", "", "", "", "00", "10", "11", "01", "000", "100", "010", "001", "110", "011", "101",
"111", "0000", "1000", "0100", "0010", "0001", "1010", "1001", "1100", "0011", "0101", "0110",
"1110", "1011", "1101", "0111", "1111" };

int len = 1 << nofv;
for (int i = len; i < len * 2; i++) {
    data.add(new Graytable(a[i], nofv));
}

// the Table size

table.setMaxWidth(25 * table.getColumns().size() + tableColumn.getWidth() + 100);
table.setMinWidth(25 * table.getColumns().size() + tableColumn.getWidth() + 100);

});

// instead of constructor

table.setItems(data);

// Allow the user to edit the table and it's contents

table.setEditable(true);

// add all of musicBtn and solveBtn and choiceBox to a container (to a HBox)

HBox controlBox = new HBox(MusicBtn, SolveBtn, choice);
controlBox.setStyle("-fx-background-color:rgba(2 , 0 ,32 , 0.5); -fx-background-radius:25");
controlBox.setPadding(new Insets(2, 10, 2, 10));
controlBox.setSpacing(20);
controlBox.setAlignment(Pos.CENTER);
root1.getChildren().addAll(controlBox);
root1.add(table, 0, 1);
root1.setPadding(new Insets(20));
root.getChildren().addAll(video2, root1);

scene = new Scene(root);

stage.setScene(scene);
stage.setTitle("K-map");

```

نلاحظ هنا انه تم استخدام مصفوفة سلاسل وقمنا بالتعبئة من خلالها.

```

}

private void fillrow( int n ) {
    data1.clear();
    switch( n ) {
        case 2 : data1.add( new k_map( "0", lastColumn.getCellData( 0 ), lastColumn.getCellData( 1 ), 0, 0 ) );
                data1.add( new k_map( "1", lastColumn.getCellData( 3 ), lastColumn.getCellData( 2 ), 0, 0 ) ); break;

        case 3 : data1.add( new k_map( "0", lastColumn.getCellData( 0 ), lastColumn.getCellData( 1 ), lastColumn.getCellData( 4 ), lastColumn.getCellData( 2 ) ) );
                data1.add( new k_map( "1", lastColumn.getCellData( 3 ), lastColumn.getCellData( 6 ), lastColumn.getCellData( 7 ), lastColumn.getCellData( 5 ) ) ); break;

        case 4 : data1.add( new k_map( "00", lastColumn.getCellData( 0 ), lastColumn.getCellData( 1 ), lastColumn.getCellData( 7 ), lastColumn.getCellData( 2 ) ) );
                data1.add( new k_map( "10", lastColumn.getCellData( 3 ), lastColumn.getCellData( 5 ), lastColumn.getCellData( 11 ), lastColumn.getCellData( 10 ) ) );
                data1.add( new k_map( "11", lastColumn.getCellData( 8 ), lastColumn.getCellData( 12 ), lastColumn.getCellData( 15 ), lastColumn.getCellData( 14 ) ) );
                data1.add( new k_map( "01", lastColumn.getCellData( 4 ), lastColumn.getCellData( 6 ), lastColumn.getCellData( 13 ), lastColumn.getCellData( 9 ) ) ); break;
    }
}

private void fillmap() {
    Solve.Rows = data1.size() - 4 ? 4 : 2;
    Solve.Columns = table1.getColumns().size() - 1;
    int i = -1;
    for ( TableColumn< k_map, ? > col : table1.getColumns() ) {
        if ( i >= -1 ) {
            ++i;
            continue;
        }
        for ( int j = 0; j < Solve.Rows; ++j )
            Solve.map[ j ][ i ] = (String) col.getCellData( j );
        ++i;
    }
}

public static void main(String[] args) {
    Application.launch(args);
}
}

```

هنا لدينا تعبئة القيم حسب عدد المتحولات المختار في كل مرة وتعبئة قيم الخريطة أيضا وفي النهاية في هذا الصف لدينا التابع الرئيسي main ونقوم بإطلاق الواجهة السابقة داخله.

الان الصف Solve :

يحتوي على حل الخريطة وهو اهم خطوة قد قمنا بها.

يحتوي ثلاث توابع رئيسية وهي k_map2 أي لمعالجة المتحولين والكود تم بناءه هنا على أساس انهما فقط متحولين فقمنا ب مناقشات كل الحالات بما ان عددها قليل مقارنة مع 3 و 4 متحولات.

```

private static void Kmap2() {
    txt.setText("Y = ");
    check = true;
    for (int i = 0; i < Rows; ++i) {
        for (int j = 0; j < Colmun; ++j) {
            if (map[i][j].equals( "0" ) || (map[i][j].equals( "X" ) )) {
                check = false;
                break;
            }
        }
        if (!check)
            break;
    }
    if ( check ) {
        txt.appendText("1");
        return;
    }

    // checking all combinations of groups of 2

    //HORIZONTAL

    if(map[0][0] .equals( "1" )&& map[0][1].equals( "1" ))
    {
        if ( (map[1][0].equals( "0" ) || map[1][0].equals( "X" ) ) && (map[1][1].equals( "0" ) || map[1][1].equals( "X" ) ) )
        {
            Y="B";
        }
    }
    else if(map[1][0] .equals( "1" )&& map[1][1].equals( "1" ))
    {
        if ( (map[0][0].equals( "0" ) || map[0][0].equals( "X" ) ) && (map[0][1].equals( "0" ) || map[0][1].equals( "X" ) ) )
        {
            Y="B";
        }
    }
}

```

الان لنرى التابع K_map3 ك لمحة كما شاهدنا سابقا تابع المتحولين حيث يقوم k_map3 بمناقشة جميع الحالات الممكنة ل 3 متحولات مدخلة أولا ناقشنا الحالة الكبرى أي الخريطة ممثلة وثم المجموعات بأربع واحدات ومن ثم المجموعات بواحدين وفي النهاية ناقشنا حالة وجود واحد لوحده قد يمكننا ضمه لمجموعة ما أو فقط يبقى وحده كما يلي:

الان لنرى التابع K_map4 ك لمحة كما شاهدنا سابقا تابع الK_map3 حيث يقوم k_map4 بمناقشة جميع الحالات الممكنة ل 4 متحولات مدخلة أولا ناقشنا الحالة الكبرى أي الخريطة ممثلة 16 عنصر و ثم المجموعات 8 ثم بأربع وحدات ومن ثم المجموعات بواحد وفي النهاية ناقشنا حالة وجود واحد لوحده قد يمكننا ضمه لمجموعة ما أو فقط يبقى وحده كما يلي:

```

190
191 private static void Enap4() {
192     test.setText(" ");
193     check = true;
194     for (int i = 0; i < 4096; ++i) {
195         for (int j = 0; j < 4096; ++j) {
196             if (map[i][j].equals("0") || map[i][j].equals("X") > 0) {
197                 check = false;
198                 break;
199             }
200             if (i%4096)
201                 break;
202         }
203         if (check) {
204             test.appendText("1");
205             return;
206         }
207     }
208     // checking groups of 8
209     //NR(2074)
210
211     for(int i=0;i<4096)
212     {
213         if ( map[0][i].equals("1")&& map[1][i+156].equals("1")&& map[2][i+312].equals("1")&& map[3][i].equals("1")&& map[4][i+468].equals("1")&& map[5][i+624].equals("1")&& map[6][i+780].equals("1")&& map[7][i+936].equals("1")&& map[8][i+1092].equals("1")&& map[9][i+1248].equals("1")&& map[10][i+1404].equals("1")&& map[11][i+1560].equals("1")&& map[12][i+1716].equals("1")&& map[13][i+1872].equals("1")&& map[14][i+2028].equals("1")&& map[15][i+2184].equals("1")&& map[16][i+2340].equals("1")&& map[17][i+2496].equals("1")&& map[18][i+2652].equals("1")&& map[19][i+2808].equals("1")&& map[20][i+2964].equals("1")&& map[21][i+3120].equals("1")&& map[22][i+3276].equals("1")&& map[23][i+3432].equals("1")&& map[24][i+3588].equals("1")&& map[25][i+3744].equals("1")&& map[26][i+3900].equals("1")&& map[27][i+4056].equals("1")&& map[28][i+4212].equals("1")&& map[29][i+4368].equals("1")&& map[30][i+4524].equals("1")&& map[31][i+4680].equals("1")&& map[32][i+4836].equals("1")&& map[33][i+4992].equals("1")&& map[34][i+5148].equals("1")&& map[35][i+5304].equals("1")&& map[36][i+5460].equals("1")&& map[37][i+5616].equals("1")&& map[38][i+5772].equals("1")&& map[39][i+5928].equals("1")&& map[40][i+6084].equals("1")&& map[41][i+6240].equals("1")&& map[42][i+6396].equals("1")&& map[43][i+6552].equals("1")&& map[44][i+6708].equals("1")&& map[45][i+6864].equals("1")&& map[46][i+7020].equals("1")&& map[47][i+7176].equals("1")&& map[48][i+7332].equals("1")&& map[49][i+7488].equals("1")&& map[50][i+7644].equals("1")&& map[51][i+7800].equals("1")&& map[52][i+7956].equals("1")&& map[53][i+8112].equals("1")&& map[54][i+8268].equals("1")&& map[55][i+8424].equals("1")&& map[56][i+8580].equals("1")&& map[57][i+8736].equals("1")&& map[58][i+8892].equals("1")&& map[59][i+9048].equals("1")&& map[60][i+9204].equals("1")&& map[61][i+9360].equals("1")&& map[62][i+9516].equals("1")&& map[63][i+9672].equals("1")&& map[64][i+9828].equals("1")&& map[65][i+9984].equals("1")&& map[66][i+10140].equals("1")&& map[67][i+10296].equals("1")&& map[68][i+10452].equals("1")&& map[69][i+10608].equals("1")&& map[70][i+10764].equals("1")&& map[71][i+10920].equals("1")&& map[72][i+11076].equals("1")&& map[73][i+11232].equals("1")&& map[74][i+11388].equals("1")&& map[75][i+11544].equals("1")&& map[76][i+11700].equals("1")&& map[77][i+11856].equals("1")&& map[78][i+12012].equals("1")&& map[79][i+12168].equals("1")&& map[80][i+12324].equals("1")&& map[81][i+12480].equals("1")&& map[82][i+12636].equals("1")&& map[83][i+12792].equals("1")&& map[84][i+12948].equals("1")&& map[85][i+13104].equals("1")&& map[86][i+13260].equals("1")&& map[87][i+13416].equals("1")&& map[88][i+13572].equals("1")&& map[89][i+13728].equals("1")&& map[90][i+13884].equals("1")&& map[91][i+14040].equals("1")&& map[92][i+14196].equals("1")&& map[93][i+14352].equals("1")&& map[94][i+14508].equals("1")&& map[95][i+14664].equals("1")&& map[96][i+14820].equals("1")&& map[97][i+14976].equals("1")&& map[98][i+15132].equals("1")&& map[99][i+15288].equals("1")&& map[100][i+15444].equals("1")&& map[101][i+15600].equals("1")&& map[102][i+15756].equals("1")&& map[103][i+15912].equals("1")&& map[104][i+16068].equals("1")&& map[105][i+16224].equals("1")&& map[106][i+16380].equals("1")&& map[107][i+16536].equals("1")&& map[108][i+16692].equals("1")&& map[109][i+16848].equals("1")&& map[110][i+17004].equals("1")&& map[111][i+17160].equals("1")&& map[112][i+17316].equals("1")&& map[113][i+17472].equals("1")&& map[114][i+17628].equals("1")&& map[115][i+17784].equals("1")&& map[116][i+17940].equals("1")&& map[117][i+18096].equals("1")&& map[118][i+18252].equals("1")&& map[119][i+18408].equals("1")&& map[120][i+18564].equals("1")&& map[121][i+18720].equals("1")&& map[122][i+18876].equals("1")&& map[123][i+19032].equals("1")&& map[124][i+19188].equals("1")&& map[125][i+19344].equals("1")&& map[126][i+19500].equals("1")&& map[127][i+19656].equals("1")&& map[128][i+19812].equals("1")&& map[129][i+19968].equals("1")&& map[130][i+20124].equals("1")&& map[131][i+20280].equals("1")&& map[132][i+20436].equals("1")&& map[133][i+20592].equals("1")&& map[134][i+20748].equals("1")&& map[135][i+20904].equals("1")&& map[136][i+21060].equals("1")&& map[137][i+21216].equals("1")&& map[138][i+21372].equals("1")&& map[139][i+21528].equals("1")&& map[140][i+21684].equals("1")&& map[141][i+21840].equals("1")&& map[142][i+21996].equals("1")&& map[143][i+22152].equals("1")&& map[144][i+22308].equals("1")&& map[145][i+22464].equals("1")&& map[146][i+22620].equals("1")&& map[147][i+22776].equals("1")&& map[148][i+22932].equals("1")&& map[149][i+23088].equals("1")&& map[150][i+23244].equals("1")&& map[151][i+23400].equals("1")&& map[152][i+23556].equals("1")&& map[153][i+23712].equals("1")&& map[154][i+23868].equals("1")&& map[155][i+24024].equals("1")&& map[156][i+24180].equals("1")&& map[157][i+24336].equals("1")&& map[158][i+24492].equals("1")&& map[159][i+24648].equals("1")&& map[160][i+24804].equals("1")&& map[161][i+24960].equals("1")&& map[162][i+25116].equals("1")&& map[163][i+25272].equals("1")&& map[164][i+25428].equals("1")&& map[165][i+25584].equals("1")&& map[166][i+25740].equals("1")&& map[167][i+25896].equals("1")&& map[168][i+26052].equals("1")&& map[169][i+26208].equals("1")&& map[170][i+26364].equals("1")&& map[171][i+26520].equals("1")&& map[172][i+26676].equals("1")&& map[173][i+26832].equals("1")&& map[174][i+26988].equals
```

```

// checking groups of 4
//SQR5
for(int i=0;i<4;i++){
    for(int j=0;j<4;j++){
        if( mp[i][j].equals("1") && mp[i+1][j].equals("1") && mp[i][j+1].equals("1") && mp[i+1][j+1].equals("1") && !vis[i][j] && !vis[i+1][j] && !vis[i][j+1] && !vis[i+1][j+1] )
        {
            chosenRow.add( i ); chosenCol.add( j );
            chosenRow.add( i+1 ); chosenCol.add( j );
            chosenRow.add( i ); chosenCol.add( j+1 );
            chosenRow.add( i+1 ); chosenCol.add( j+1 );
            System.out.println("sum");
            if( S2==0 )
                F += "A";
            else
                F += "B";
            if( i==0 || j==0 )
                F += "C";
            if( S2!=0 )
                F += "D";
            else
                F += "E";
            if( i==0 || j==0 )
                F += "F";
            F += "G";
        }
    }
}
//SQR6
for(int i=0;i<4;i++){
    if( mp[i][0].equals("1") && mp[i][1].equals("1") && mp[i][2].equals("1") && mp[i][3].equals("1") && !vis[i][0] && !vis[i][1] && !vis[i][2] && !vis[i][3] )
    {
        chosenRow.add( i ); chosenCol.add( 0 );
        chosenRow.add( i ); chosenCol.add( 1 );
        chosenRow.add( i ); chosenCol.add( 2 );
        chosenRow.add( i ); chosenCol.add( 3 );
        System.out.println("sum");
        F += "H";
    }
}

```

```

// checking groups of 4
//SQR7
for(int i=0;i<4;i++){
    for(int j=0;j<4;j++){
        if( mp[i][j].equals("2") && !vis[i][j] )
        {
            if( mp[i][j+1].equals("1") )
            {
                vis[i][j] = true;
                if( S2==0 )
                    F += "A";
                else
                    F += "B";
                if( i==0 || j==0 )
                    F += "C";
                if( i==0 || j==1 )
                    F += "D";
                if( i==0 || j==2 )
                    F += "E";
                if( i==0 || j==3 )
                    F += "F";
            }
            else if( mp[i+1][j].equals("1") )
            {
                vis[i][j] = true;
                if( i==0 || j==0 )
                    F += "G";
                if( S2!=0 )
                    F += "H";
                else
                    F += "I";
                if( i==0 || j==1 )
                    F += "J";
                if( i==0 || j==2 )
                    F += "K";
                if( i==0 || j==3 )
                    F += "L";
            }
            else if( mp[i+1][j+1].equals("1") )
            {
                vis[i][j] = true;
                if( i==0 || j==0 )
                    F += "M";
                if( S2!=0 )
                    F += "N";
                else
                    F += "O";
                if( i==0 || j==1 )
                    F += "P";
                if( i==1 || j==0 )
                    F += "Q";
                if( i==1 || j==1 )
                    F += "R";
            }
        }
    }
}

```

```

111 //VERTICAL
112 for(int i=0;i<4;i++){
113     if(map[0][i].equals("1") && map[1][i].equals("1") && map[2][i].equals("1") && map[3][i].equals("1")){
114         chooseRow.add( i ); chooseCol.add( i );
115         chooseRow.add( i ); chooseCol.add( i );
116         chooseRow.add( i ); chooseCol.add( i );
117         chooseRow.add( i ); chooseCol.add( i );
118         System.out.println("win");
119         V += "A";
120         if(i==0 || i==3)
121             V += "1";
122         if(i==1 || i==2)
123             V += "2";
124         if(i==0 || i==3)
125             V += "4";
126     }
127 }
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
141 }
142 }
143 }
144 }
145 }
146 }
147 }
148 }
149 }
150 }
151 }
152 }
153 }
154 }
155 }
156 }
157 }
158 }
159 }
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
201 }
202 }
203 }
204 }
205 }
206 }
207 }
208 }
209 }
210 }
211 }
212 }
213 }
214 }
215 }
216 }
217 }
218 }
219 }
220 }
221 }
222 }
223 }
224 }
225 }
226 }
227 }
228 }
229 }
230 }
231 }
232 }
233 }
234 }
235 }
236 }
237 }
238 }
239 }
240 }
241 }
242 }
243 }
244 }
245 }
246 }
247 }
248 }
249 }
250 }
251 }
252 }
253 }
254 }
255 }
256 }
257 }
258 }
259 }
260 }
261 }
262 }
263 }
264 }
265 }
266 }
267 }
268 }
269 }
270 }
271 }
272 }
273 }
274 }
275 }
276 }
277 }
278 }
279 }
280 }
281 }
282 }
283 }
284 }
285 }
286 }
287 }
288 }
289 }
290 }
291 }
292 }
293 }
294 }
295 }
296 }
297 }
298 }
299 }
300 }
301 }
302 }
303 }
304 }
305 }
306 }
307 }
308 }
309 }
310 }
311 }
312 }
313 }
314 }
315 }
316 }
317 }
318 }
319 }
320 }
321 }
322 }
323 }
324 }
325 }
326 }
327 }
328 }
329 }
330 }
331 }
332 }
333 }
334 }
335 }
336 }
337 }
338 }
339 }
340 }
341 }
342 }
343 }
344 }
345 }
346 }
347 }
348 }
349 }
350 }
351 }
352 }
353 }
354 }
355 }
356 }
357 }
358 }
359 }
360 }
361 }
362 }
363 }
364 }
365 }
366 }
367 }
368 }
369 }
370 }
371 }
372 }
373 }
374 }
375 }
376 }
377 }
378 }
379 }
380 }
381 }
382 }
383 }
384 }
385 }
386 }
387 }
388 }
389 }
390 }
391 }
392 }
393 }
394 }
395 }
396 }
397 }
398 }
399 }
400 }
401 }
402 }
403 }
404 }
405 }
406 }
407 }
408 }
409 }
410 }
411 }
412 }
413 }
414 }
415 }
416 }
417 }
418 }
419 }
420 }
421 }
422 }
423 }
424 }
425 }
426 }
427 }
428 }
429 }
430 }
431 }
432 }
433 }
434 }
435 }
436 }
437 }
438 }
439 }
440 }
441 }
442 }
443 }
444 }
445 }
446 }
447 }
448 }
449 }
450 }
451 }
452 }
453 }
454 }
455 }
456 }
457 }
458 }
459 }
460 }
461 }
462 }
463 }
464 }
465 }
466 }
467 }
468 }
469 }
470 }
471 }
472 }
473 }
474 }
475 }
476 }
477 }
478 }
479 }
480 }
481 }
482 }
483 }
484 }
485 }
486 }
487 }
488 }
489 }
490 }
491 }
492 }
493 }
494 }
495 }
496 }
497 }
498 }
499 }
500 }
501 }
502 }
503 }
504 }
505 }
506 }
507 }
508 }
509 }
510 }
511 }
512 }
513 }
514 }
515 }
516 }
517 }
518 }
519 }
520 }
521 }
522 }
523 }
524 }
525 }
526 }
527 }
528 }
529 }
530 }
531 }
532 }
533 }
534 }
535 }
536 }
537 }
538 }
539 }
540 }
541 }
542 }
543 }
544 }
545 }
546 }
547 }
548 }
549 }
550 }
551 }
552 }
553 }
554 }
555 }
556 }
557 }
558 }
559 }
560 }
561 }
562 }
563 }
564 }
565 }
566 }
567 }
568 }
569 }
570 }
571 }
572 }
573 }
574 }
575 }
576 }
577 }
578 }
579 }
580 }
581 }
582 }
583 }
584 }
585 }
586 }
587 }
588 }
589 }
590 }
591 }
592 }
593 }
594 }
595 }
596 }
597 }
598 }
599 }
600 }
601 }
602 }
603 }
604 }
605 }
606 }
607 }
608 }
609 }
610 }
611 }
612 }
613 }
614 }
615 }
616 }
617 }
618 }
619 }
620 }
621 }
622 }
623 }
624 }
625 }
626 }
627 }
628 }
629 }
630 }
631 }
632 }
633 }
634 }
635 }
636 }
637 }
638 }
639 }
640 }
641 }
642 }
643 }
644 }
645 }
646 }
647 }
648 }
649 }
650 }
651 }
652 }
653 }
654 }
655 }
656 }
657 }
658 }
659 }
660 }
661 }
662 }
663 }
664 }
665 }
666 }
667 }
668 }
669 }
670 }
671 }
672 }
673 }
674 }
675 }
676 }
677 }
678 }
679 }
680 }
681 }
682 }
683 }
684 }
685 }
686 }
687 }
688 }
689 }
690 }
691 }
692 }
693 }
694 }
695 }
696 }
697 }
698 }
699 }
700 }
701 }
702 }
703 }
704 }
705 }
706 }
707 }
708 }
709 }
710 }
711 }
712 }
713 }
714 }
715 }
716 }
717 }
718 }
719 }
720 }
721 }
722 }
723 }
724 }
725 }
726 }
727 }
728 }
729 }
730 }
731 }
732 }
733 }
734 }
735 }
736 }
737 }
738 }
739 }
740 }
741 }
742 }
743 }
744 }
745 }
746 }
747 }
748 }
749 }
750 }
751 }
752 }
753 }
754 }
755 }
756 }
757 }
758 }
759 }
760 }
761 }
762 }
763 }
764 }
765 }
766 }
767 }
768 }
769 }
770 }
771 }
772 }
773 }
774 }
775 }
776 }
777 }
778 }
779 }
780 }
781 }
782 }
783 }
784 }
785 }
786 }
787 }
788 }
789 }
790 }
791 }
792 }
793 }
794 }
795 }
796 }
797 }
798 }
799 }
800 }
801 }
802 }
803 }
804 }
805 }
806 }
807 }
808 }
809 }
810 }
811 }
812 }
813 }
814 }
815 }
816 }
817 }
818 }
819 }
820 }
821 }
822 }
823 }
824 }
825 }
826 }
827 }
828 }
829 }
830 }
831 }
832 }
833 }
834 }
835 }
836 }
837 }
838 }
839 }
840 }
841 }
842 }
843 }
844 }
845 }
846 }
847 }
848 }
849 }
850 }
851 }
852 }
853 }
854 }
855 }
856 }
857 }
858 }
859 }
860 }
861 }
862 }
863 }
864 }
865 }
866 }
867 }
868 }
869 }
870 }
871 }
872 }
873 }
874 }
875 }
876 }
877 }
878 }
879 }
880 }
881 }
882 }
883 }
884 }
885 }
886 }
887 }
888 }
889 }
890 }
891 }
892 }
893 }
894 }
895 }
896 }
897 }
898 }
899 }
900 }
901 }
902 }
903 }
904 }
905 }
906 }
907 }
908 }
909 }
910 }
911 }
912 }
913 }
914 }
915 }
916 }
917 }
918 }
919 }
920 }
921 }
922 }
923 }
924 }
925 }
926 }
927 }
928 }
929 }
930 }
931 }
932 }
933 }
934 }
935 }
936 }
937 }
938 }
939 }
940 }
941 }
942 }
943 }
944 }
945 }
946 }
947 }
948 }
949 }
950 }
951 }
952 }
953 }
954 }
955 }
956 }
957 }
958 }
959 }
960 }
961 }
962 }
963 }
964 }
965 }
966 }
967 }
968 }
969 }
970 }
971 }
972 }
973 }
974 }
975 }
976 }
977 }
978 }
979 }
980 }
981 }
982 }
983 }
984 }
985 }
986 }
987 }
988 }
989 }
990 }
991 }
992 }
993 }
994 }
995 }
996 }
997 }
998 }
999 }
1000 }

```

وقمنا باستدعاء تابع ال Sovle في الصف momo,الان لنرى صفى الخريطة وجدول كارنوف.

```

1 import java.beans.property.SimpleStringProperty;
2
3 public class k_map {
4     SimpleStringProperty a1,a2,a3,a4,a5;
5
6     public k_map( String a1, int a2, int a3, int a4, int a5 ) {
7         this.a1 = new SimpleStringProperty( a1 );
8         this.a2 = new SimpleStringProperty( a2 );
9         this.a3 = new SimpleStringProperty( a3 );
10        this.a4 = new SimpleStringProperty( a4 );
11        this.a5 = new SimpleStringProperty( a5 );
12    }
13
14    private String getValue( int n ) {
15        return n == -1 ? "A" : "" + n;
16    }
17
18    public void setA1(String a1)
19    {
20        this.a1 = new SimpleStringProperty(a1);
21    }
22    public String getA1()
23    {
24        return a1.getValue();
25    }
26
27    public void setA2(String a2)
28    {
29        this.a2 = new SimpleStringProperty(a2);
30    }
31    public String getA2()
32    {
33        return a2.getValue();
34    }
35
36    public void setA3(String a3)
37    {
38        this.a3 = new SimpleStringProperty(a3);
39    }
40    public String getA3()
41    {
42        return a3.getValue();
43    }
44
45    public void setA4(String a4)
46    {
47        this.a4 = new SimpleStringProperty(a4);
48    }
49    public String getA4()
50    {
51        return a4.getValue();
52    }
53
54    public void setA5(String a5)
55    {
56        this.a5 = new SimpleStringProperty(a5);
57    }
58    public String getA5()
59    {
60        return a5.getValue();
61    }
62
63    public void setA1(int a1)
64    {
65        this.a1 = new SimpleStringProperty(a1);
66    }
67    public String getA1()
68    {
69        return a1.getValue();
70    }
71
72    public void setA2(int a2)
73    {
74        this.a2 = new SimpleStringProperty(a2);
75    }
76    public String getA2()
77    {
78        return a2.getValue();
79    }
80
81    public void setA3(int a3)
82    {
83        this.a3 = new SimpleStringProperty(a3);
84    }
85    public String getA3()
86    {
87        return a3.getValue();
88    }
89
90    public void setA4(int a4)
91    {
92        this.a4 = new SimpleStringProperty(a4);
93    }
94    public String getA4()
95    {
96        return a4.getValue();
97    }
98
99    public void setA5(int a5)
100   {
101       this.a5 = new SimpleStringProperty(a5);
102   }
103   public String getA5()
104   {
105       return a5.getValue();
106   }
107
108   public void setA1(String a1, int a2, int a3, int a4, int a5)
109   {
110       this.a1 = new SimpleStringProperty(a1);
111       this.a2 = new SimpleStringProperty(a2);
112       this.a3 = new SimpleStringProperty(a3);
113       this.a4 = new SimpleStringProperty(a4);
114       this.a5 = new SimpleStringProperty(a5);
115   }
116
117   public String getA1()
118   {
119       return a1.getValue();
120   }
121
122   public String getA2()
123   {
124       return a2.getValue();
125   }
126
127   public String getA3()
128   {
129       return a3.getValue();
130   }
131
132   public String getA4()
133   {
134       return a4.getValue();
135   }
136
137   public String getA5()
138   {
139       return a5.getValue();
140   }
141
142   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6)
143   {
144       this.a1 = new SimpleStringProperty(a1);
145       this.a2 = new SimpleStringProperty(a2);
146       this.a3 = new SimpleStringProperty(a3);
147       this.a4 = new SimpleStringProperty(a4);
148       this.a5 = new SimpleStringProperty(a5);
149       this.a6 = new SimpleStringProperty(a6);
150   }
151
152   public String getA1()
153   {
154       return a1.getValue();
155   }
156
157   public String getA2()
158   {
159       return a2.getValue();
160   }
161
162   public String getA3()
163   {
164       return a3.getValue();
165   }
166
167   public String getA4()
168   {
169       return a4.getValue();
170   }
171
172   public String getA5()
173   {
174       return a5.getValue();
175   }
176
177   public String getA6()
178   {
179       return a6.getValue();
180   }
181
182   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7)
183   {
184       this.a1 = new SimpleStringProperty(a1);
185       this.a2 = new SimpleStringProperty(a2);
186       this.a3 = new SimpleStringProperty(a3);
187       this.a4 = new SimpleStringProperty(a4);
188       this.a5 = new SimpleStringProperty(a5);
189       this.a6 = new SimpleStringProperty(a6);
190       this.a7 = new SimpleStringProperty(a7);
191   }
192
193   public String getA1()
194   {
195       return a1.getValue();
196   }
197
198   public String getA2()
199   {
200       return a2.getValue();
201   }
202
203   public String getA3()
204   {
205       return a3.getValue();
206   }
207
208   public String getA4()
209   {
210       return a4.getValue();
211   }
212
213   public String getA5()
214   {
215       return a5.getValue();
216   }
217
218   public String getA6()
219   {
220       return a6.getValue();
221   }
222
223   public String getA7()
224   {
225       return a7.getValue();
226   }
227
228   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8)
229   {
230       this.a1 = new SimpleStringProperty(a1);
231       this.a2 = new SimpleStringProperty(a2);
232       this.a3 = new SimpleStringProperty(a3);
233       this.a4 = new SimpleStringProperty(a4);
234       this.a5 = new SimpleStringProperty(a5);
235       this.a6 = new SimpleStringProperty(a6);
236       this.a7 = new SimpleStringProperty(a7);
237       this.a8 = new SimpleStringProperty(a8);
238   }
239
240   public String getA1()
241   {
242       return a1.getValue();
243   }
244
245   public String getA2()
246   {
247       return a2.getValue();
248   }
249
250   public String getA3()
251   {
252       return a3.getValue();
253   }
254
255   public String getA4()
256   {
257       return a4.getValue();
258   }
259
260   public String getA5()
261   {
262       return a5.getValue();
263   }
264
265   public String getA6()
266   {
267       return a6.getValue();
268   }
269
270   public String getA7()
271   {
272       return a7.getValue();
273   }
274
275   public String getA8()
276   {
277       return a8.getValue();
278   }
279
280   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9)
281   {
282       this.a1 = new SimpleStringProperty(a1);
283       this.a2 = new SimpleStringProperty(a2);
284       this.a3 = new SimpleStringProperty(a3);
285       this.a4 = new SimpleStringProperty(a4);
286       this.a5 = new SimpleStringProperty(a5);
287       this.a6 = new SimpleStringProperty(a6);
288       this.a7 = new SimpleStringProperty(a7);
289       this.a8 = new SimpleStringProperty(a8);
290       this.a9 = new SimpleStringProperty(a9);
291   }
292
293   public String getA1()
294   {
295       return a1.getValue();
296   }
297
298   public String getA2()
299   {
300       return a2.getValue();
301   }
302
303   public String getA3()
304   {
305       return a3.getValue();
306   }
307
308   public String getA4()
309   {
310       return a4.getValue();
311   }
312
313   public String getA5()
314   {
315       return a5.getValue();
316   }
317
318   public String getA6()
319   {
320       return a6.getValue();
321   }
322
323   public String getA7()
324   {
325       return a7.getValue();
326   }
327
328   public String getA8()
329   {
330       return a8.getValue();
331   }
332
333   public String getA9()
334   {
335       return a9.getValue();
336   }
337
338   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10)
339   {
340       this.a1 = new SimpleStringProperty(a1);
341       this.a2 = new SimpleStringProperty(a2);
342       this.a3 = new SimpleStringProperty(a3);
343       this.a4 = new SimpleStringProperty(a4);
344       this.a5 = new SimpleStringProperty(a5);
345       this.a6 = new SimpleStringProperty(a6);
346       this.a7 = new SimpleStringProperty(a7);
347       this.a8 = new SimpleStringProperty(a8);
348       this.a9 = new SimpleStringProperty(a9);
349       this.a10 = new SimpleStringProperty(a10);
350   }
351
352   public String getA1()
353   {
354       return a1.getValue();
355   }
356
357   public String getA2()
358   {
359       return a2.getValue();
360   }
361
362   public String getA3()
363   {
364       return a3.getValue();
365   }
366
367   public String getA4()
368   {
369       return a4.getValue();
370   }
371
372   public String getA5()
373   {
374       return a5.getValue();
375   }
376
377   public String getA6()
378   {
379       return a6.getValue();
380   }
381
382   public String getA7()
383   {
384       return a7.getValue();
385   }
386
387   public String getA8()
388   {
389       return a8.getValue();
390   }
391
392   public String getA9()
393   {
394       return a9.getValue();
395   }
396
397   public String getA10()
398   {
399       return a10.getValue();
400   }
401
402   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10, String a11)
403   {
404       this.a1 = new SimpleStringProperty(a1);
405       this.a2 = new SimpleStringProperty(a2);
406       this.a3 = new SimpleStringProperty(a3);
407       this.a4 = new SimpleStringProperty(a4);
408       this.a5 = new SimpleStringProperty(a5);
409       this.a6 = new SimpleStringProperty(a6);
410       this.a7 = new SimpleStringProperty(a7);
411       this.a8 = new SimpleStringProperty(a8);
412       this.a9 = new SimpleStringProperty(a9);
413       this.a10 = new SimpleStringProperty(a10);
414       this.a11 = new SimpleStringProperty(a11);
415   }
416
417   public String getA1()
418   {
419       return a1.getValue();
420   }
421
422   public String getA2()
423   {
424       return a2.getValue();
425   }
426
427   public String getA3()
428   {
429       return a3.getValue();
430   }
431
432   public String getA4()
433   {
434       return a4.getValue();
435   }
436
437   public String getA5()
438   {
439       return a5.getValue();
440   }
441
442   public String getA6()
443   {
444       return a6.getValue();
445   }
446
447   public String getA7()
448   {
449       return a7.getValue();
450   }
451
452   public String getA8()
453   {
454       return a8.getValue();
455   }
456
457   public String getA9()
458   {
459       return a9.getValue();
460   }
461
462   public String getA10()
463   {
464       return a10.getValue();
465   }
466
467   public String getA11()
468   {
469       return a11.getValue();
470   }
471
472   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10, String a11, String a12)
473   {
474       this.a1 = new SimpleStringProperty(a1);
475       this.a2 = new SimpleStringProperty(a2);
476       this.a3 = new SimpleStringProperty(a3);
477       this.a4 = new SimpleStringProperty(a4);
478       this.a5 = new SimpleStringProperty(a5);
479       this.a6 = new SimpleStringProperty(a6);
480       this.a7 = new SimpleStringProperty(a7);
481       this.a8 = new SimpleStringProperty(a8);
482       this.a9 = new SimpleStringProperty(a9);
483       this.a10 = new SimpleStringProperty(a10);
484       this.a11 = new SimpleStringProperty(a11);
485       this.a12 = new SimpleStringProperty(a12);
486   }
487
488   public String getA1()
489   {
490       return a1.getValue();
491   }
492
493   public String getA2()
494   {
495       return a2.getValue();
496   }
497
498   public String getA3()
499   {
500       return a3.getValue();
501   }
502
503   public String getA4()
504   {
505       return a4.getValue();
506   }
507
508   public String getA5()
509   {
510       return a5.getValue();
511   }
512
513   public String getA6()
514   {
515       return a6.getValue();
516   }
517
518   public String getA7()
519   {
520       return a7.getValue();
521   }
522
523   public String getA8()
524   {
525       return a8.getValue();
526   }
527
528   public String getA9()
529   {
530       return a9.getValue();
531   }
532
533   public String getA10()
534   {
535       return a10.getValue();
536   }
537
538   public String getA11()
539   {
540       return a11.getValue();
541   }
542
543   public String getA12()
544   {
545       return a12.getValue();
546   }
547
548   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10, String a11, String a12, String a13)
549   {
550       this.a1 = new SimpleStringProperty(a1);
551       this.a2 = new SimpleStringProperty(a2);
552       this.a3 = new SimpleStringProperty(a3);
553       this.a4 = new SimpleStringProperty(a4);
554       this.a5 = new SimpleStringProperty(a5);
555       this.a6 = new SimpleStringProperty(a6);
556       this.a7 = new SimpleStringProperty(a7);
557       this.a8 = new SimpleStringProperty(a8);
558       this.a9 = new SimpleStringProperty(a9);
559       this.a10 = new SimpleStringProperty(a10);
560       this.a11 = new SimpleStringProperty(a11);
561       this.a12 = new SimpleStringProperty(a12);
562       this.a13 = new SimpleStringProperty(a13);
563   }
564
565   public String getA1()
566   {
567       return a1.getValue();
568   }
569
570   public String getA2()
571   {
572       return a2.getValue();
573   }
574
575   public String getA3()
576   {
577       return a3.getValue();
578   }
579
580   public String getA4()
581   {
582       return a4.getValue();
583   }
584
585   public String getA5()
586   {
587       return a5.getValue();
588   }
589
590   public String getA6()
591   {
592       return a6.getValue();
593   }
594
595   public String getA7()
596   {
597       return a7.getValue();
598   }
599
600   public String getA8()
601   {
602       return a8.getValue();
603   }
604
605   public String getA9()
606   {
607       return a9.getValue();
608   }
609
610   public String getA10()
611   {
612       return a10.getValue();
613   }
614
615   public String getA11()
616   {
617       return a11.getValue();
618   }
619
620   public String getA12()
621   {
622       return a12.getValue();
623   }
624
625   public String getA13()
626   {
627       return a13.getValue();
628   }
629
630   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10, String a11, String a12, String a13, String a14)
631   {
632       this.a1 = new SimpleStringProperty(a1);
633       this.a2 = new SimpleStringProperty(a2);
634       this.a3 = new SimpleStringProperty(a3);
635       this.a4 = new SimpleStringProperty(a4);
636       this.a5 = new SimpleStringProperty(a5);
637       this.a6 = new SimpleStringProperty(a6);
638       this.a7 = new SimpleStringProperty(a7);
639       this.a8 = new SimpleStringProperty(a8);
640       this.a9 = new SimpleStringProperty(a9);
641       this.a10 = new SimpleStringProperty(a10);
642       this.a11 = new SimpleStringProperty(a11);
643       this.a12 = new SimpleStringProperty(a12);
644       this.a13 = new SimpleStringProperty(a13);
645       this.a14 = new SimpleStringProperty(a14);
646   }
647
648   public String getA1()
649   {
650       return a1.getValue();
651   }
652
653   public String getA2()
654   {
655       return a2.getValue();
656   }
657
658   public String getA3()
659   {
660       return a3.getValue();
661   }
662
663   public String getA4()
664   {
665       return a4.getValue();
666   }
667
668   public String getA5()
669   {
670       return a5.getValue();
671   }
672
673   public String getA6()
674   {
675       return a6.getValue();
676   }
677
678   public String getA7()
679   {
680       return a7.getValue();
681   }
682
683   public String getA8()
684   {
685       return a8.getValue();
686   }
687
688   public String getA9()
689   {
690       return a9.getValue();
691   }
692
693   public String getA10()
694   {
695       return a10.getValue();
696   }
697
698   public String getA11()
699   {
700       return a11.getValue();
701   }
702
703   public String getA12()
704   {
705       return a12.getValue();
706   }
707
708   public String getA13()
709   {
710       return a13.getValue();
711   }
712
713   public String getA14()
714   {
715       return a14.getValue();
716   }
717
718   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10, String a11, String a12, String a13, String a14, String a15)
719   {
720       this.a1 = new SimpleStringProperty(a1);
721       this.a2 = new SimpleStringProperty(a2);
722       this.a3 = new SimpleStringProperty(a3);
723       this.a4 = new SimpleStringProperty(a4);
724       this.a5 = new SimpleStringProperty(a5);
725       this.a6 = new SimpleStringProperty(a6);
726       this.a7 = new SimpleStringProperty(a7);
727       this.a8 = new SimpleStringProperty(a8);
728       this.a9 = new SimpleStringProperty(a9);
729       this.a10 = new SimpleStringProperty(a10);
730       this.a11 = new SimpleStringProperty(a11);
731       this.a12 = new SimpleStringProperty(a12);
732       this.a13 = new SimpleStringProperty(a13);
733       this.a14 = new SimpleStringProperty(a14);
734       this.a15 = new SimpleStringProperty(a15);
735   }
736
737   public String getA1()
738   {
739       return a1.getValue();
740   }
741
742   public String getA2()
743   {
744       return a2.getValue();
745   }
746
747   public String getA3()
748   {
749       return a3.getValue();
750   }
751
752   public String getA4()
753   {
754       return a4.getValue();
755   }
756
757   public String getA5()
758   {
759       return a5.getValue();
760   }
761
762   public String getA6()
763   {
764       return a6.getValue();
765   }
766
767   public String getA7()
768   {
769       return a7.getValue();
770   }
771
772   public String getA8()
773   {
774       return a8.getValue();
775   }
776
777   public String getA9()
778   {
779       return a9.getValue();
780   }
781
782   public String getA10()
783   {
784       return a10.getValue();
785   }
786
787   public String getA11()
788   {
789       return a11.getValue();
790   }
791
792   public String getA12()
793   {
794       return a12.getValue();
795   }
796
797   public String getA13()
798   {
799       return a13.getValue();
800   }
801
799   public String getA14()
800   {
801       return a14.getValue();
802   }
803
804   public String getA15()
805   {
806       return a15.getValue();
807   }
808
809   public void setA1(String a1, int a2, int a3, int a4, int a5, String a6, String a7, String a8, String a9, String a10, String a11, String a12, String a13, String a14
```



```
1 package org.springframework.samples.karyonof;
2
3 import org.springframework.stereotype.Component;
4 import org.springframework.util.StringUtils;
5
6 @Component
7 public class Karyonof {
8     private String a1;
9     private String a2;
10    private String a3;
11    private String a4;
12
13    public String getA1() {
14        return a1;
15    }
16
17    public void setA1(String a1) {
18        this.a1 = a1;
19    }
20
21    public String getA2() {
22        return a2;
23    }
24
25    public void setA2(String a2) {
26        this.a2 = a2;
27    }
28
29    public String getA3() {
30        return a3;
31    }
32
33    public void setA3(String a3) {
34        this.a3 = a3;
35    }
36
37    public String getA4() {
38        return a4;
39    }
40
41    public void setA4(String a4) {
42        this.a4 = a4;
43    }
44
45    public String getA1A2() {
46        return a1 + a2;
47    }
48
49    public void setA1A2(String a1A2) {
50        this.a1A2 = a1A2;
51    }
52
53    public String getA3A4() {
54        return a3 + a4;
55    }
56
57    public void setA3A4(String a3A4) {
58        this.a3A4 = a3A4;
59    }
60
61    public String getA1A2A3A4() {
62        return a1 + a2 + a3 + a4;
63    }
64
65    public void setA1A2A3A4(String a1A2A3A4) {
66        this.a1A2A3A4 = a1A2A3A4;
67    }
68
69    public String getA1A2A3() {
70        return a1 + a2 + a3;
71    }
72
73    public void setA1A2A3(String a1A2A3) {
74        this.a1A2A3 = a1A2A3;
75    }
76
77    public String getA1A2A4() {
78        return a1 + a2 + a4;
79    }
80
81    public void setA1A2A4(String a1A2A4) {
82        this.a1A2A4 = a1A2A4;
83    }
84
85    public String getA1A3A4() {
86        return a1 + a3 + a4;
87    }
88
89    public void setA1A3A4(String a1A3A4) {
90        this.a1A3A4 = a1A3A4;
91    }
92
93    public String getA2A3A4() {
94        return a2 + a3 + a4;
95    }
96
97    public void setA2A3A4(String a2A3A4) {
98        this.a2A3A4 = a2A3A4;
99    }
100
101    public String getA1A2A3A4A() {
102        return a1 + a2 + a3 + a4 + "A";
103    }
104
105    public void setA1A2A3A4A(String a1A2A3A4A) {
106        this.a1A2A3A4A = a1A2A3A4A;
107    }
108
109    public String getA1A2A3A4B() {
110        return a1 + a2 + a3 + a4 + "B";
111    }
112
113    public void setA1A2A3A4B(String a1A2A3A4B) {
114        this.a1A2A3A4B = a1A2A3A4B;
115    }
116
117    public String getA1A2A3A4C() {
118        return a1 + a2 + a3 + a4 + "C";
119    }
120
121    public void setA1A2A3A4C(String a1A2A3A4C) {
122        this.a1A2A3A4C = a1A2A3A4C;
123    }
124
125    public String getA1A2A3A4D() {
126        return a1 + a2 + a3 + a4 + "D";
127    }
128
129    public void setA1A2A3A4D(String a1A2A3A4D) {
130        this.a1A2A3A4D = a1A2A3A4D;
131    }
132
133    public String getA1A2A3A4E() {
134        return a1 + a2 + a3 + a4 + "E";
135    }
136
137    public void setA1A2A3A4E(String a1A2A3A4E) {
138        this.a1A2A3A4E = a1A2A3A4E;
139    }
140
141    public String getA1A2A3A4F() {
142        return a1 + a2 + a3 + a4 + "F";
143    }
144
145    public void setA1A2A3A4F(String a1A2A3A4F) {
146        this.a1A2A3A4F = a1A2A3A4F;
147    }
148
149    public String getA1A2A3A4G() {
150        return a1 + a2 + a3 + a4 + "G";
151    }
152
153    public void setA1A2A3A4G(String a1A2A3A4G) {
154        this.a1A2A3A4G = a1A2A3A4G;
155    }
156
157    public String getA1A2A3A4H() {
158        return a1 + a2 + a3 + a4 + "H";
159    }
160
161    public void setA1A2A3A4H(String a1A2A3A4H) {
162        this.a1A2A3A4H = a1A2A3A4H;
163    }
164
165    public String getA1A2A3A4I() {
166        return a1 + a2 + a3 + a4 + "I";
167    }
168
169    public void setA1A2A3A4I(String a1A2A3A4I) {
170        this.a1A2A3A4I = a1A2A3A4I;
171    }
172
173    public String getA1A2A3A4J() {
174        return a1 + a2 + a3 + a4 + "J";
175    }
176
177    public void setA1A2A3A4J(String a1A2A3A4J) {
178        this.a1A2A3A4J = a1A2A3A4J;
179    }
180
181    public String getA1A2A3A4K() {
182        return a1 + a2 + a3 + a4 + "K";
183    }
184
185    public void setA1A2A3A4K(String a1A2A3A4K) {
186        this.a1A2A3A4K = a1A2A3A4K;
187    }
188
189    public String getA1A2A3A4L() {
190        return a1 + a2 + a3 + a4 + "L";
191    }
192
193    public void setA1A2A3A4L(String a1A2A3A4L) {
194        this.a1A2A3A4L = a1A2A3A4L;
195    }
196
197    public String getA1A2A3A4M() {
198        return a1 + a2 + a3 + a4 + "M";
199    }
200
201    public void setA1A2A3A4M(String a1A2A3A4M) {
202        this.a1A2A3A4M = a1A2A3A4M;
203    }
204
205    public String getA1A2A3A4N() {
206        return a1 + a2 + a3 + a4 + "N";
207    }
208
209    public void setA1A2A3A4N(String a1A2A3A4N) {
210        this.a1A2A3A4N = a1A2A3A4N;
211    }
212
213    public String getA1A2A3A4O() {
214        return a1 + a2 + a3 + a4 + "O";
215    }
216
217    public void setA1A2A3A4O(String a1A2A3A4O) {
218        this.a1A2A3A4O = a1A2A3A4O;
219    }
220
221    public String getA1A2A3A4P() {
222        return a1 + a2 + a3 + a4 + "P";
223    }
224
225    public void setA1A2A3A4P(String a1A2A3A4P) {
226        this.a1A2A3A4P = a1A2A3A4P;
227    }
228
229    public String getA1A2A3A4Q() {
230        return a1 + a2 + a3 + a4 + "Q";
231    }
232
233    public void setA1A2A3A4Q(String a1A2A3A4Q) {
234        this.a1A2A3A4Q = a1A2A3A4Q;
235    }
236
237    public String getA1A2A3A4R() {
238        return a1 + a2 + a3 + a4 + "R";
239    }
240
241    public void setA1A2A3A4R(String a1A2A3A4R) {
242        this.a1A2A3A4R = a1A2A3A4R;
243    }
244
245    public String getA1A2A3A4S() {
246        return a1 + a2 + a3 + a4 + "S";
247    }
248
249    public void setA1A2A3A4S(String a1A2A3A4S) {
250        this.a1A2A3A4S = a1A2A3A4S;
251    }
252
253    public String getA1A2A3A4T() {
254        return a1 + a2 + a3 + a4 + "T";
255    }
256
257    public void setA1A2A3A4T(String a1A2A3A4T) {
258        this.a1A2A3A4T = a1A2A3A4T;
259    }
260
261    public String getA1A2A3A4U() {
262        return a1 + a2 + a3 + a4 + "U";
263    }
264
265    public void setA1A2A3A4U(String a1A2A3A4U) {
266        this.a1A2A3A4U = a1A2A3A4U;
267    }
268
269    public String getA1A2A3A4V() {
270        return a1 + a2 + a3 + a4 + "V";
271    }
272
273    public void setA1A2A3A4V(String a1A2A3A4V) {
274        this.a1A2A3A4V = a1A2A3A4V;
275    }
276
277    public String getA1A2A3A4W() {
278        return a1 + a2 + a3 + a4 + "W";
279    }
280
281    public void setA1A2A3A4W(String a1A2A3A4W) {
282        this.a1A2A3A4W = a1A2A3A4W;
283    }
284
285    public String getA1A2A3A4X() {
286        return a1 + a2 + a3 + a4 + "X";
287    }
288
289    public void setA1A2A3A4X(String a1A2A3A4X) {
290        this.a1A2A3A4X = a1A2A3A4X;
291    }
292
293    public String getA1A2A3A4Y() {
294        return a1 + a2 + a3 + a4 + "Y";
295    }
296
297    public void setA1A2A3A4Y(String a1A2A3A4Y) {
298        this.a1A2A3A4Y = a1A2A3A4Y;
299    }
300
301    public String getA1A2A3A4Z() {
302        return a1 + a2 + a3 + a4 + "Z";
303    }
304
305    public void setA1A2A3A4Z(String a1A2A3A4Z) {
306        this.a1A2A3A4Z = a1A2A3A4Z;
307    }
308
309    public String getA1A2A3A4AA() {
310        return a1 + a2 + a3 + a4 + "AA";
311    }
312
313    public void setA1A2A3A4AA(String a1A2A3A4AA) {
314        this.a1A2A3A4AA = a1A2A3A4AA;
315    }
316
317    public String getA1A2A3A4AB() {
318        return a1 + a2 + a3 + a4 + "AB";
319    }
320
321    public void setA1A2A3A4AB(String a1A2A3A4AB) {
322        this.a1A2A3A4AB = a1A2A3A4AB;
323    }
324
325    public String getA1A2A3A4AC() {
326        return a1 + a2 + a3 + a4 + "AC";
327    }
328
329    public void setA1A2A3A4AC(String a1A2A3A4AC) {
330        this.a1A2A3A4AC = a1A2A3A4AC;
331    }
332
333    public String getA1A2A3A4AD() {
334        return a1 + a2 + a3 + a4 + "AD";
335    }
336
337    public void setA1A2A3A4AD(String a1A2A3A4AD) {
338        this.a1A2A3A4AD = a1A2A3A4AD;
339    }
340
341    public String getA1A2A3A4AE() {
342        return a1 + a2 + a3 + a4 + "AE";
343    }
344
345    public void setA1A2A3A4AE(String a1A2A3A4AE) {
346        this.a1A2A3A4AE = a1A2A3A4AE;
347    }
348
349    public String getA1A2A3A4AF() {
350        return a1 + a2 + a3 + a4 + "AF";
351    }
352
353    public void setA1A2A3A4AF(String a1A2A3A4AF) {
354        this.a1A2A3A4AF = a1A2A3A4AF;
355    }
356
357    public String getA1A2A3A4AG() {
358        return a1 + a2 + a3 + a4 + "AG";
359    }
360
361    public void setA1A2A3A4AG(String a1A2A3A4AG) {
362        this.a1A2A3A4AG = a1A2A3A4AG;
363    }
364
365    public String getA1A2A3A4AH() {
366        return a1 + a2 + a3 + a4 + "AH";
367    }
368
369    public void setA1A2A3A4AH(String a1A2A3A4AH) {
370        this.a1A2A3A4AH = a1A2A3A4AH;
371    }
372
373    public String getA1A2A3A4AI() {
374        return a1 + a2 + a3 + a4 + "AI";
375    }
376
377    public void setA1A2A3A4AI(String a1A2A3A4AI) {
378        this.a1A2A3A4AI = a1A2A3A4AI;
379    }
380
381    public String getA1A2A3A4AJ() {
382        return a1 + a2 + a3 + a4 + "AJ";
383    }
384
385    public void setA1A2A3A4AJ(String a1A2A3A4AJ) {
386        this.a1A2A3A4AJ = a1A2A3A4AJ;
387    }
388
389    public String getA1A2A3A4AK() {
390        return a1 + a2 + a3 + a4 + "AK";
391    }
392
393    public void setA1A2A3A4AK(String a1A2A3A4AK) {
394        this.a1A2A3A4AK = a1A2A3A4AK;
395    }
396
397    public String getA1A2A3A4AL() {
398        return a1 + a2 + a3 + a4 + "AL";
399    }
400
401    public void setA1A2A3A4AL(String a1A2A3A4AL) {
402        this.a1A2A3A4AL = a1A2A3A4AL;
403    }
404
405    public String getA1A2A3A4AM() {
406        return a1 + a2 + a3 + a4 + "AM";
407    }
408
409    public void setA1A2A3A4AM(String a1A2A3A4AM) {
410        this.a1A2A3A4AM = a1A2A3A4AM;
411    }
412
413    public String getA1A2A3A4AN() {
414        return a1 + a2 + a3 + a4 + "AN";
415    }
416
417    public void setA1A2A3A4AN(String a1A2A3A4AN) {
418        this.a1A2A3A4AN = a1A2A3A4AN;
419    }
420
421    public String getA1A2A3A4AO() {
422        return a1 + a2 + a3 + a4 + "AO";
423    }
424
425    public void setA1A2A3A4AO(String a1A2A3A4AO) {
426        this.a1A2A3A4AO = a1A2A3A4AO;
427    }
428
429    public String getA1A2A3A4AP() {
430        return a1 + a2 + a3 + a4 + "AP";
431    }
432
433    public void setA1A2A3A4AP(String a1A2A3A4AP) {
434        this.a1A2A3A4AP = a1A2A3A4AP;
435    }
436
437    public String getA1A2A3A4AQ() {
438        return a1 + a2 + a3 + a4 + "AQ";
439    }
440
441    public void setA1A2A3A4AQ(String a1A2A3A4AQ) {
442        this.a1A2A3A4AQ = a1A2A3A4AQ;
443    }
444
445    public String getA1A2A3A4AR() {
446        return a1 + a2 + a3 + a4 + "AR";
447    }
448
449    public void setA1A2A3A4AR(String a1A2A3A4AR) {
450        this.a1A2A3A4AR = a1A2A3A4AR;
451    }
452
453    public String getA1A2A3A4AS() {
454        return a1 + a2 + a3 + a4 + "AS";
455    }
456
457    public void setA1A2A3A4AS(String a1A2A3A4AS) {
458        this.a1A2A3A4AS = a1A2A3A4AS;
459    }
460
461    public String getA1A2A3A4AT() {
462        return a1 + a2 + a3 + a4 + "AT";
463    }
464
465    public void setA1A2A3A4AT(String a1A2A3A4AT) {
466        this.a1A2A3A4AT = a1A2A3A4AT;
467    }
468
469    public String getA1A2A3A4AU() {
470        return a1 + a2 + a3 + a4 + "AU";
471    }
472
473    public void setA1A2A3A4AU(String a1A2A3A4AU) {
474        this.a1A2A3A4AU = a1A2A3A4AU;
475    }
476
477    public String getA1A2A3A4AV() {
478        return a1 + a2 + a3 + a4 + "AV";
479    }
480
481    public void setA1A2A3A4AV(String a1A2A3A4AV) {
482        this.a1A2A3A4AV = a1A2A3A4AV;
483    }
484
485    public String getA1A2A3A4AW() {
486        return a1 + a2 + a3 + a4 + "AW";
487    }
488
489    public void setA1A2A3A4AW(String a1A2A3A4AW) {
490        this.a1A2A3A4AW = a1A2A3A4AW;
491    }
492
493    public String getA1A2A3A4AX() {
494        return a1 + a2 + a3 + a4 + "AX";
495    }
496
497    public void setA1A2A3A4AX(String a1A2A3A4AX) {
498        this.a1A2A3A4AX = a1A2A3A4AX;
499    }
500
501    public String getA1A2A3A4AY() {
502        return a1 + a2 + a3 + a4 + "AY";
503    }
504
505    public void setA1A2A3A4AY(String a1A2A3A4AY) {
506        this.a1A2A3A4AY = a1A2A3A4AY;
507    }
508
509    public String getA1A2A3A4AZ() {
510        return a1 + a2 + a3 + a4 + "AZ";
511    }
512
513    public void setA1A2A3A4AZ(String a1A2A3A4AZ) {
514        this.a1A2A3A4AZ = a1A2A3A4AZ;
515    }
516
517    public String getA1A2A3A4BA() {
518        return a1 + a2 + a3 + a4 + "BA";
519    }
520
521    public void setA1A2A3A4BA(String a1A2A3A4BA) {
522        this.a1A2A3A4BA = a1A2A3A4BA;
523    }
524
525    public String getA1A2A3A4BB() {
526        return a1 + a2 + a3 + a4 + "BB";
527    }
528
529    public void setA1A2A3A4BB(String a1A2A3A4BB) {
530        this.a1A2A3A4BB = a1A2A3A4BB;
531    }
532
533    public String getA1A2A3A4BC() {
534        return a1 + a2 + a3 + a4 + "BC";
535    }
536
537    public void setA1A2A3A4BC(String a1A2A3A4BC) {
538        this.a1A2A3A4BC = a1A2A3A4BC;
539    }
540
541    public String getA1A2A3A4BD() {
542        return a1 + a2 + a3 + a4 + "BD";
543    }
544
545    public void setA1A2A3A4BD(String a1A2A3A4BD) {
546        this.a1A2A3A4BD = a1A2A3A4BD;
547    }
548
549    public String getA1A2A3A4BE() {
550        return a1 + a2 + a3 + a4 + "BE";
551    }
552
553    public void setA1A2A3A4BE(String a1A2A3A4BE) {
554        this.a1A2A3A4BE = a1A2A3A4BE;
555    }
556
557    public String getA1A2A3A4BF() {
558        return a1 + a2 + a3 + a4 + "BF";
559    }
560
561    public void setA1A2A3A4BF(String a1A2A3A4BF) {
562        this.a1A2A3A4BF = a1A2A3A4BF;
563    }
564
565    public String getA1A2A3A4BG() {
566        return a1 + a2 + a3 + a4 + "BG";
567    }
568
569    public void setA1A2A3A4BG(String a1A2A3A4BG) {
570        this.a1A2A3A4BG = a1A2A3A4BG;
571    }
572
573    public String getA1A2A3A4BH() {
574        return a1 + a2 + a3 + a4 + "BH";
575    }
576
577    public void setA1A2A3A4BH(String a1A2A3A4BH) {
578        this.a1A2A3A4BH = a1A2A3A4BH;
579    }
580
581    public String getA1A2A3A4BI() {
582        return a1 + a2 + a3 + a4 + "BI";
583    }
584
585    public void setA1A2A3A4BI(String a1A2A3A4BI) {
586        this.a1A2A3A4BI = a1A2A3A4BI;
587    }
588
589    public String getA1A2A3A4BJ() {
590        return a1 + a2 + a3 + a4 + "BJ";
591    }
592
593    public void setA1A2A3A4BJ(String a1A2A3A4BJ) {
594        this.a1A2A3A4BJ = a1A2A3A4BJ;
595    }
596
597    public String getA1A2A3A4BK() {
598        return a1 + a2 + a3 + a4 + "BK";
599    }
600
601    public void setA1A2A3A4BK(String a1A2A3A4BK) {
602        this.a1A2A3A4BK = a1A2A3A4BK;
603    }
604
605    public String getA1A2A3A4BL() {
606        return a1 + a2 + a3 + a4 + "BL";
607    }
608
609    public void setA1A2A3A4BL(String a1A2A3A4BL) {
610        this.a1A2A3A4BL = a1A2A3A4BL;
611    }
612
613    public String getA1A2A3A4BM() {
614        return a1 + a2 + a3 + a4 + "BM";
615    }
616
617    public void setA1A2A3A4BM(String a1A2A3A4BM) {
618        this.a1A2A3A4BM = a1A2A3A4BM;
619    }
620
621    public String getA1A2A3A4BN() {
622        return a1 + a2 + a3 + a4 + "BN";
623    }
624
625    public void setA1A2A3A4BN(String a1A2A3A4BN) {
626        this.a1A2A3A4BN = a1A2A3A4BN;
627    }
628
629    public String getA1A2A3A4BO() {
630        return a1 + a2 + a3 + a4 + "BO";
631    }
632
633    public void setA1A2A3A4BO(String a1A2A3A4BO) {
634        this.a1A2A3A4BO = a1A2A3A4BO;
635    }
636
637    public String getA1A2A3A4BP() {
638        return a1 + a2 + a3 + a4 + "BP";
639    }
640
641    public void setA1A2A3A4BP(String a1A2A3A4BP) {
642        this.a1A2A3A4BP = a1A2A3A4BP;
643    }
644
645    public String getA1A2A3A4BQ() {
646        return a1 + a2 + a3 + a4 + "BQ";
647    }
648
649    public void setA1A2A3A4BQ(String a1A2A3A4BQ) {
650        this.a1A2A3A4BQ = a1A2A3A4BQ;
651    }
652
653    public String getA1A2A3A4BR() {
654        return a1 + a2 + a3 + a4 + "BR";
655    }
656
657    public void setA1A2A3A4BR(String a1A2A3A4BR) {
658        this.a1A2A3A4BR = a1A2A3A4BR;
659    }
660
661    public String getA1A2A3A4BS() {
662        return a1 + a2 + a3 + a4 + "BS";
663    }
664
665    public void setA1A2A3A4BS(String a1A2A3A4BS) {
666        this.a1A2A3A4BS = a1A2A3A4BS;
667    }
668
669    public String getA1A2A3A4BT() {
670        return a1 + a2 + a3 + a4 + "BT";
671    }
672
673    public void setA1A2A3A4BT(String a1A2A3A4BT) {
674        this.a1A2A3A4BT = a1A2A3A4BT;
675    }
676
677    public String getA1A2A3A4BU() {
678        return a1 + a2 + a3 + a4 + "BU";
679    }
680
681    public void setA1A2A3A4BU(String a1A2A3A4BU) {
682        this.a1A2A3A4BU = a1A2A3A4BU;
683    }
684
685    public String getA1A2A3A4BV() {
686        return a1 + a2 + a3 + a4 + "BV";
687    }
688
689    public void setA1A2A3A4BV(String a1A2A3A4BV) {
690        this.a1A2A3A4BV = a1A2A3A4BV;
691    }
692
693    public String getA1A2A3A4BW() {
694        return a1 + a2 + a3 + a4 + "BW";
695    }
696
697    public void setA1A2A3A4BW(String a1A2A3A4BW) {
698        this.a1A2A3A4BW = a1A2A3A4BW;
699    }
700
701    public String getA1A2A3A4BX() {
702        return a1 + a2 + a3 + a4 + "BX";
703    }
704
705    public void setA1A2A3A4BX(String a1A2A3A4BX) {
706        this.a1A2A3A4BX = a1A2A3A4BX;
707    }
708
709    public String getA1A2A3A4BY() {
710        return a1 + a2 + a3 + a4 + "BY";
711    }
712
713    public void setA1A2A3A4BY(String a1A2A3A4BY) {
714        this.a1A2A3A4BY = a1A2A3A4BY;
715    }
716
717    public String getA1A2A3A4BZ() {
718        return a1 + a2 + a3 + a4 + "BZ";
719    }
720
721    public void setA1A2A3A4BZ(String a1A2A3A4BZ) {
722        this.a1A2A3A4BZ = a1A2A3A4BZ;
723    }
724
725    public String getA1A2A3A4CA() {
726        return a1 + a2 + a3 + a4 + "CA";
727    }
728
729    public void setA1A2A3A4CA(String a1A2A3A4CA) {
730        this.a1A2A3A4CA = a1A2A3A4CA;
731    }
732
733    public String getA1A2A3A4CB() {
734        return a1 + a2 + a3 + a4 + "CB";
735    }
736
737    public void setA1A2A3A4CB(String a1A2A3A4CB) {
738        this.a1A2A3A4CB = a1A2A3A4CB;
739    }
740
741    public String getA1A2A3A4CC() {
742        return a1 + a2 + a3 + a4 + "CC";
743    }
744
745    public void setA1A2A3A4CC(String a1A2A3A4CC) {
746        this.a1A2A3A4CC = a1A2A3A4CC;
747    }
748
749    public String getA1A2A3A4CD() {
750        return a1 + a2 + a3 + a4 + "CD";
751    }
752
753    public void setA1A2A3A4CD(String a1A2A3A4CD) {
754        this.a1A2A3A4CD = a1A2A3A4CD;
755    }
756
757    public String getA1A2A3A4CE() {
758        return a1 + a2 + a3 + a4 + "CE";
759    }
760
761    public void setA1A2A3A4CE(String a1A2A3A4CE) {
762        this.a1A2A3A4CE = a1A2A3A4CE;
763    }
764
765    public String getA1A2A3A4CF() {
766        return a1 + a2 + a3 + a4 + "CF";
767    }
768
769    public void setA1A2A3A4CF(String a1A2A3A4CF) {
770        this.a1A2A3A4CF = a1A2A3A4CF;
771    }
772
773    public String getA1A2A3A4CG() {
774        return a1 + a2 + a3 + a4 + "CG";
775    }
776
777    public void setA1A2A3A4CG(String a1A2A3A4CG) {
778        this.a1A2A3A4CG = a1A2A3A4CG;
779    }
780
781    public String getA1A2A3A4CH() {
782        return a1 + a2 + a3 + a4 + "CH";
783    }
784
785    public void setA1A2A3A4CH(String a1A2A3A4CH) {
786        this.a1A2A3A4CH = a1A2A3A4CH;
787    }
788
789    public String getA1A2A3A4CI() {
790        return a1 + a2 + a3 + a4 + "CI";
791    }
792
793    public void setA1A2A3A4CI(String a1A2A3A4CI) {
794        this.a1A2A3A4CI = a1A2A3A4CI;
795    }
796
797    public String getA1A2A3A4CJ() {
798        return a1 + a2 + a3 + a4 + "CJ";
799    }
800
801    public void setA1A2A3A4CJ(String a1A2A3A4CJ) {
802        this.a1A2A3A4CJ = a1A2A3A4CJ;
803    }
804
805    public String getA1A2A3A4CK() {
806        return a1 + a2 + a3 + a4 + "CK";
807    }
808
809    public void setA1A2A3A4CK(String a1A2A3A4CK) {
810        this.a1A2A3A4CK = a1A2A3A4CK;
811    }
812
813    public String getA1A2A3A4CL() {
814        return a1 + a2 + a3 + a4 + "CL";
815    }
816
817    public void setA1A2A3A4CL(String a1A2A3A4CL) {
818        this.a1A2A3A4CL = a1A2A3A4CL;
819    }
820
821    public String getA1A2A3A4CM() {
822        return a1 + a2 + a3 + a4 + "CM";
823    }
824
825    public void setA1A2A3A4CM(String a1A2A3A4CM) {
826        this.a1A2A3A4CM = a1A2A3A4CM;
827    }
828
829    public String getA1A2A3A4CN() {
830        return a1 + a2 + a3 + a4 + "CN";
831    }
832
833    public void setA1A2A3A4CN(String a1A2A3A4CN) {
834        this.a1A2A3A4CN = a1A2A3A4CN;
835    }
836
837    public String getA1A2A3A4CO() {
838        return a1 + a2 + a3 + a4 + "CO";
839    }
840
841    public void setA1A2A3A4CO(String a1A2A3A4CO) {
842        this.a1A2A3A4CO = a1A2A3A4CO;
843    }
844
845    public String getA1A2A3A4CP() {
846        return a1 + a2 + a3 + a4 + "CP";
847    }
848
849    public void setA1A2A3A4CP(String a1A2A3A4CP) {
850        this.a1A2A3A4CP = a1A2A3A4CP;
851    }
852
853    public String getA1A2A3A4CQ() {
854        return a1 + a2 + a3 + a4 + "CQ";
855    }
856
857    public void setA1A2A3A4CQ(String a1A2A3A4CQ) {
858        this.a1A2A3A4CQ = a1A2A3A4CQ;
859    }
860
861    public String getA1A2A3A4CR() {
862        return a1 + a2 + a3 + a4 + "CR";
863    }
864
865    public void setA1A2A3A4CR(String a1A2A3A4CR) {
866        this.a1A2A3A4CR = a1A2A3A4CR;
867    }
868
869    public String getA1A2A3A4CS() {
870        return a1 + a2 + a3 + a4 + "CS";
871    }
872
873    public void setA1A2A3A4CS(String a1A2A3A4CS) {
874        this.a1A2A3A4CS = a1A2A3A4CS;
875    }
876
877    public String getA1A2A3A4CT() {
878        return a1 + a2 + a3 + a4 + "CT";
879    }
880
881    public void setA1A2A3A4CT(String a1A2A3A4CT) {
882        this.a1A2A3A4CT = a1A2A3A4CT;
883    }
884
885    public String getA1A2A3A4CU() {
886        return a1 + a2 + a3 + a4 + "CU";
887    }
888
889    public void setA1A2A3A4CU(String a1A2A3A4CU) {
890        this.a1A2A3A4CU = a1A2A3A4CU;
891    }
892
893    public String getA1A2A3A4CV() {
894        return a1 + a2 + a3 + a
```



```

1 import java.beans.property.SimpleIntegerProperty;
2
3 public class Graytable
4 {
5     SimpleIntegerProperty x1,x2,x3,x4,x5;
6
7     public void setX1(int x)
8     {
9         this.x1 = new SimpleIntegerProperty(x);
10    }
11
12    public int getX1()
13    {
14        return x1.getValue();
15    }
16
17    public void setX2(Integer x)
18    {
19        this.x2 = new SimpleIntegerProperty(x);
20    }
21
22    public Integer getX2()
23    {
24        return x2.getValue();
25    }
26
27    public void setX3(Integer x)
28    {
29        this.x3 = new SimpleIntegerProperty(x);
30    }
31
32    public Integer getX3()
33    {
34        return x3.getValue();
35    }
36
37    public void setX4(Integer x)
38    {
39        this.x4 = new SimpleIntegerProperty(x);
40    }
41
42    public Integer getX4()
43    {
44        return x4.getValue();
45    }
46
47    public void setX5(Integer x)
48    {
49        this.x5 = new SimpleIntegerProperty(x);
50    }
51
52    public Integer getX5()
53    {
54        return x5.getValue();
55    }
56
57    public Graytable( String s , int n )
58    {
59        if (n==2)
60        {
61            x1 = new SimpleIntegerProperty( s.charAt( 0 ) - 48 );
62            x2 = new SimpleIntegerProperty( s.charAt( 1 ) - 48 );
63            x3 = new SimpleIntegerProperty(0);
64            x4 = new SimpleIntegerProperty(0);
65            x5 = new SimpleIntegerProperty(0);
66        }
67        else if(n==3)
68        {
69            x1 = new SimpleIntegerProperty( s.charAt( 0 ) - 48 );
70            x2 = new SimpleIntegerProperty( s.charAt( 1 ) - 48 );
71            x3 = new SimpleIntegerProperty( s.charAt( 2 ) - 48 );
72            x4 = new SimpleIntegerProperty(0);
73            x5 = new SimpleIntegerProperty(0);
74        }
75        else if(n==4)
76        {
77            x1 = new SimpleIntegerProperty( s.charAt( 0 ) - 48 );
78            x2 = new SimpleIntegerProperty( s.charAt( 1 ) - 48 );
79            x3 = new SimpleIntegerProperty( s.charAt( 2 ) - 48 );
80            x4 = new SimpleIntegerProperty( s.charAt( 3 ) - 48 );
81            x5 = new SimpleIntegerProperty(0);
82        }
83    }
84 }

```

```

43 public Integer getX4()
44 {
45     return x4.getValue();
46 }
47
48 public void setX5(Integer x)
49 {
50     this.x5 = new SimpleIntegerProperty(x);
51 }
52
53 public Integer getX5()
54 {
55     return x5.getValue();
56 }
57
58 public Graytable( String s , int n )
59 {
60     if (n==2)
61     {
62         x1 = new SimpleIntegerProperty( s.charAt( 0 ) - 48 );
63         x2 = new SimpleIntegerProperty( s.charAt( 1 ) - 48 );
64         x3 = new SimpleIntegerProperty(0);
65         x4 = new SimpleIntegerProperty(0);
66         x5 = new SimpleIntegerProperty(0);
67     }
68     else if(n==3)
69     {
70         x1 = new SimpleIntegerProperty( s.charAt( 0 ) - 48 );
71         x2 = new SimpleIntegerProperty( s.charAt( 1 ) - 48 );
72         x3 = new SimpleIntegerProperty( s.charAt( 2 ) - 48 );
73         x4 = new SimpleIntegerProperty(0);
74         x5 = new SimpleIntegerProperty(0);
75     }
76     else if(n==4)
77     {
78         x1 = new SimpleIntegerProperty( s.charAt( 0 ) - 48 );
79         x2 = new SimpleIntegerProperty( s.charAt( 1 ) - 48 );
80         x3 = new SimpleIntegerProperty( s.charAt( 2 ) - 48 );
81         x4 = new SimpleIntegerProperty( s.charAt( 3 ) - 48 );
82         x5 = new SimpleIntegerProperty(0);
83     }
84 }

```

قمنا استخدام خمسة متغيرات وهو عدد الاعمدة الاعظمي لدينا في حال $n=4$ حيث n هو عدد المتحولات واستخدمناها في الباني الخاص بجدول كارنوف Graytable وأخيرا قمنا بكتابة كل من تابعي ال get وال set الخاص بكل متغير .

وبهكذا يكون تم بناء خوارزمية كارنوف على أكمل وجه وكان بالإمكان عرض الخرج على شكل بوابات منطقية أما من خلال استخدام الأداة Logisim المستخدمة كثيرا في الجافا 5 وما فوق أو كان فقط بإمكاننا ربط خرج المعادلة بعد التبسيط على أي موقع نت خاص برسم المعادلة المنطقية على شكل بوابات لكن هذا قد يجعل تطبيقنا السابق لا يعمل إلا في حال كان الجهاز متصل على الشبكة وكان أيضا يمكننا باستخدام الصور رسم كل حالة من الحالات التي قد تنتج معنا ونضعها مباشرة عند يكون الخرج مطابق للصورة وذلك من خلال اسمها لكن هذا سيكلف الكثير من الجهد والتعب لذلك لم نستخدم أي من الحالات السابقة.

المراجع المستخدمة:

1. Learn JavaFX 8 - Building User Experience and Interfaces with Java 8 - 1st Edition (2015)
2. Java - The Complete Reference, Ninth Edition By Herbert Schildt
3. Introduction to Java Programming Comprehensive Version, 10th Edition

خاتمة

أخيرا عندما نستخدم خوارزمية كوين ماكلوسكي فنجد أنها طويلة جدا من حيث عدد الخطوات ومن حيث الحل أيضا وقد نخطأ أثناء استخدامها حيث أن عدد المتحولات يكون كبير وعدد الحدود كبير أيضا لذلك يمكن أن نستخدم خرائط كارنوف في التأكد من صحة نتائج طريقة كوين من خلال تطبيق كارنوف على الحدود الناتجة معنا فإذا ظهرت النتيجة ذاتها معنا فهذا يعني أنه تم الحل بشكل صحيح في كلا الطريقتين وإلا يوجد خطأ أثناء تطبيق خوارزمية كوين ماكلوسكي.

وفي النهاية وليس لطالب علم من نهاية .. بعد أن أتممنا هذا المشروع ، نرجو أن نكون قد وفقنا في جهدنا المتواضع في الوصول إلى الهدف المطلوب بالصورة المثلى .