

MÔN : CÔNG NGHỆ JAVA

Bài thực hành 5.2 : Xây dựng chương trình giải pt bậc 2 dùng AWT

I. Mục tiêu :

- Giúp SV thấy rõ lợi ích của tính thừa kế trong việc viết 1 ứng dụng cụ thể.

II. Nội dung :

- Dùng tính thừa kế để tổ chức lại mã nguồn chương trình giải phương trình bậc 2 ở chế độ đồ họa dùng các đối tượng AWT theo các Layout khác nhau.

III. Chuẩn đầu ra :

- Sinh viên nắm vững lợi ích của tính thừa kế trong việc viết 1 ứng dụng cụ thể.

IV. Qui trình :

1. Nhân bản thư mục chứa Project phần mềm được tạo ra từ bài thực hành 5.1 thành thư mục AWGTGPTB2OTM.
2. Chạy NetBean 7.3.1, nếu cửa sổ Project có hiển thị các Project cũ hãy đóng chúng lại.
3. Chọn menu File.Open Project rồi mở lại Project trong thư mục AWGTGPTB2OTM.
4. Xem lại 4 class miêu tả 4 form dùng 4 Layout khác nhau ta sẽ thấy 4 class này chứa phần chung rất lớn, sự khác biệt chỉ là 1 phần trong hàm khởi tạo form theo Layout riêng. Ta có thể dùng tính thừa kế để viết phần chung của 4 class trong 1 class riêng có tên là FrGPTB2 (class cha) rồi định nghĩa từng class miêu tả Layout cụ thể như là class con của class FrGPTB2.
5. dôi chuột về mục awtgptb2 trong cây Project, ấn kép chuột vào nó để hiển thị menu lệnh, chọn option New.Java Class để hiển thị cửa sổ "New Java Class". Nhập tên "**FrGPTB2**" vào textbox "Class name" rồi click button Finish để máy tạo thực sự class mới theo yêu cầu. Cửa sổ soạn mã nguồn của class được hiển thị, hãy soạn nội dung của class như sau :

```
package awtgptb2;
//khai báo các package chứa các tài nguyên cần dùng
import java.awt.*;
import java.awt.event.*;
public class FrGPTB2 extends Frame {
    //định nghĩa các biến tham khảo đến các đối tượng giao diện
    private Label labelA;
    private TextField txtA;
    private Label labelB;
    private TextField txtB;
    private Label labelC;
    private TextField txtC;
    private Button btnStart;
    private Label lblKetqua;
    private Label lblX1;
    private Label lblX2;
    //xóa nội dung các Textbox và Label kết quả về null
    public void init() {
        txtA.setText(null);
        txtB.setText(null);
        txtC.setText(null);
    }
}
```

```

        lblKetqua.setText(null);
        lblX1.setText(null);
        lblX2.setText(null);
    }
    //định nghĩa các biến cần dùng
    double a, b, c;
    double delta;
    double x1, x2;

    //Hàm nhập 3 tham số a, b, c của phương trình
    private void NhapABC() {
        //mã hóa dữ liệu chuỗi thành giá trị số Double
        a = Double.valueOf(txtA.getText());
        b = Double.valueOf(txtB.getText());
        c = Double.valueOf(txtC.getText());
    }

    //Hàm giải phương trình bậc 2
    private void GiaiPT() {
        //tính biệt số delta
        delta = b * b - 4 * a * c;
        //kiểm tra biệt số delata và quyết định xử lý
        if (delta >= 0) { //trường hợp có 2 nghiệm thực
            x1 = (-b + Math.sqrt(delta)) / 2 / a;
            x2 = (-b - Math.sqrt(delta)) / 2 / a;
        } else { //trường hợp vô nghiệm
            lblKetqua.setText("Phương trình vô nghiệm !");
            lblX1.setText("");
            lblX2.setText("");
        }
    }

    //Hàm xuất kết quả của phương trình bậc 2
    private void XuatKetqua() {
        //kiểm tra biệt số delata và quyết định xử lý
        if (delta >= 0) { //trường hợp có 2 nghiệm thực
            lblKetqua.setText("Phương trình có 2 nghiệm :");
            lblX1.setText("x1 = " + x1);
            lblX2.setText("x2 = " + x2);
        } else { //trường hợp vô nghiệm
            lblKetqua.setText("Phương trình vô nghiệm !");
            lblX1.setText("");
            lblX2.setText("");
        }
        this.setSize(this.getWidth(),this.getHeight()+1);
    }

    //hàm giải phương trình bậc 2 theo 3 tham số nhập
    void giaiPTB2() {
        //1. Nhập 3 tham số a, b, c của phương trình
        NhapABC();
        //2. Giải phương trình
    }

```

```

        GiaiPT();
        //3. Hiển thị kết quả
        XuấtKetqua();
    }

    public FrGPTB2() { //constructor của frame
        super("Giải phương trình bậc 2");
        setSize(350, 230); //thiết lập kích thước frame
        //định nghĩa hàm xử lý sự kiện đóng Form
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
    }
}

```

6. Hiện thị cửa sổ soạn mã nguồn của class FlowFrame rồi hiệu chỉnh lại mã nguồn của nó như sau :

```

package awtgptb2;
//khai báo các package chứa các tài nguyên cần dùng
import java.awt.*;
import java.awt.event.*;
public class FlowFrame extends FrGPTB2 {
    public FlowFrame() { //constructor của frame
        super();
        //thiết lập LayoutManager cho Frame
        this.setLayout(new FlowLayout());
        //tạo Label và add nó vào Panel
        labelA = new Label("Nhập a : ");
        this.add(labelA);
        //tạo Textbox và add nó vào Panel
        txtA = new TextField(20);
        this.add(txtA);

        //tạo Label và add nó vào Panel
        labelB = new Label("Nhập b : ");
        this.add(labelB);
        //tạo Textbox và add nó vào Panel
        txtB = new TextField(20);
        this.add(txtB);

        //tạo Label và add nó vào Panel
        labelC = new Label("Nhập c : ");
        this.add(labelC);
        //tạo Textbox và add nó vào Panel
        txtC = new TextField(20);
        this.add(txtC);

        //tạo Button và add nó vào Panel
    }
}

```

```

btnStart = new Button("Bắt đầu giải");
//định nghĩa hàm xử lý click chuột trên Button
btnStart.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent ae) {
        giaiPTB2(); //gọi hàm giải ptb2
    }
});
this.add(btnStart); //add Button vào Panel

//tạo Label kết quả và add nó vào Panel
lblKetqua = new Label("");
this.add(lblKetqua);
//tạo Label x1= và add nó vào Panel
lblX1 = new Label("");
this.add(lblX1);
//tạo Label x2= và add nó vào Panel
lblX2 = new Label("9");
this.add(lblX2);
}
}

```

7. Hiện thị cửa sổ soạn mã nguồn của class GridFrame rồi hiệu chỉnh lại mã nguồn của nó như sau :

```

package awtgptb2;
//khai báo các package chứa các tài nguyên cần dùng
import java.awt.*;
import java.awt.event.*;
public class GridFrame extends FrGPTB2 {

    public GridFrame() { //constructor của frame
        super();
        //thiết lập LayoutManager cho Frame
        this.setLayout(new GridLayout(7,1));

        //tạo Panel chứa label và textbox A
        Panel mPanel = new Panel();
        //khai báo layout cho Panel có 2 cột
        mPanel.setLayout(new GridLayout(1, 2));
        //tạo Label và add nó vào Panel
        labelA = new Label("Nhập a : ");
        mPanel.add(labelA);
        //tạo Textbox và add nó vào Panel
        txtA = new TextField("");
        mPanel.add(txtA);
        //add Panel vào Frame
        this.add(mPanel);

        //tạo Panel chứa label và textbox B
        mPanel = new Panel();
        //khai báo layout cho Panel có 2 cột

```

```

mPanel.setLayout(new GridLayout(1, 2));
//tạo Label và add nó vào Panel
labelB = new Label("Nhập b : ");
mPanel.add(labelB);
//tạo Textbox và add nó vào Panel
txtB = new TextField("");
mPanel.add(txtB);
//add Panel vào Frame
this.add(mPanel);

//tạo Panel chứa label và textbox C
mPanel = new Panel();
//khai báo layout cho Panel có 2 cột
mPanel.setLayout(new GridLayout(1, 2));
//tạo Label và add nó vào Panel
labelC = new Label("Nhập c : ");
mPanel.add(labelC);
//tạo Textbox và add nó vào Panel
txtC = new TextField("");
mPanel.add(txtC);
//add Panel vào Frame
this.add(mPanel);
//tạo Panel có 3 cột để chứa Button ở giữa
mPanel = new Panel(); mPanel.setLayout(new GridLayout(1, 3));
mPanel.add(new Label(""));
//tạo Button và add nó vào Panel
btnStart = new Button("Bắt đầu giải");
//định nghĩa hàm xử lý click chuột trên Button
btnStart.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent ae) {
        giaiPTB2(); //gọi hàm giải ptb2
    }
});
mPanel.add(btnStart); //add Button vào Panel
mPanel.add(new Label(""));
this.add(mPanel); //add Panel vào Frame

//tạo Label kết quả và add nó vào Panel
lblKetqua = new Label("");
this.add(lblKetqua);
//tạo Label x1= và add nó vào Panel
lblX1 = new Label("");
this.add(lblX1);
//tạo Label x2= và add nó vào Panel
lblX2 = new Label("9");
this.add(lblX2);
}
}

```

8. Hiện thị cửa sổ soạn mã nguồn của class GridBagFrame rồi hiệu chỉnh lại mã nguồn của nó như sau :

```

package awtgptb2;
//khai báo các package chứa các tài nguyên cần dùng
import java.awt.*;
import java.awt.event.*;
public class GridBagFrame extends FrGPTB2 {
    public GridBagFrame() { //constructor của frame
        super();
        //thiết lập LayoutManager cho Frame
        this.setLayout(new GridBagLayout());
        GridBagConstraints c = new GridBagConstraints();

        //tạo Label và add nó vào Panel
        labelA = new Label("Nhập a : ");
        c.gridx = 0; c.gridy = 0; c.gridwidth = 1; c.gridheight = 1; c.weightx = 0.1;
        this.add(labelA,c);
        //tạo Textbox và add nó vào Panel
        txtA = new TextField(20);
        c.gridx = 1; c.gridy = 0; c.gridwidth = 2; c.gridheight = 1; c.weightx = 0.1;
        this.add(txtA,c);

        //tạo Label và add nó vào Panel
        labelB = new Label("Nhập b : ");
        c.gridx = 0; c.gridy = 1; c.gridwidth = 1; c.gridheight = 1; c.weightx = 0.1;
        this.add(labelB,c);
        //tạo Textbox và add nó vào Panel
        txtB = new TextField(20);
        c.gridx = 1; c.gridy = 1; c.gridwidth = 2; c.gridheight = 1; c.weightx = 1;
        this.add(txtB,c);

        //tạo Label và add nó vào Panel
        labelC = new Label("Nhập c : ");
        c.gridx = 0; c.gridy = 2; c.gridwidth = 1; c.gridheight = 1; c.weightx = 0.1;
        this.add(labelC,c);
        //tạo Textbox và add nó vào Panel
        txtC = new TextField(20);
        c.gridx = 1; c.gridy = 2; c.gridwidth = 2; c.gridheight = 1; c.weightx = 1;
        this.add(txtC,c);
        //tạo Button và add nó vào Panel
        btnStart = new Button("Bắt đầu giải");
        //định nghĩa hàm xử lý click chuột trên Button
        btnStart.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent ae) {
                giaiPTB2(); //gọi hàm giải ptb2
            }
        });
        c.gridx = 1; c.gridy = 3; c.gridwidth = 1; c.gridheight = 1; //c.weighty = 0.1;
        this.add(btnStart,c);
        //tạo Label kết quả và add nó vào Panel
        lblKetqua = new Label(String.format("%50s", ""));
        c.gridx = 0; c.gridy = 4; c.gridwidth = 2; c.gridheight = 1; c.weightx = 0.1;

```

```

this.add(lblKetqua,c);
//tạo Label x1= và add nó vào Panel
lblX1 = new Label(String.format("%50s", ""));
c.gridx = 0; c.gridy = 5; c.gridwidth = 2; c.gridheight = 1; c.weightx = 0.1;
this.add(lblX1,c);
//tạo Label x2= và add nó vào Panel
lblX2 = new Label(String.format("%50s", ""));
c.gridx = 0; c.gridy = 6; c.gridwidth = 2; c.gridheight = 1; c.weightx = 0.1;
this.add(lblX2,c);
}
}

```

9. Hiện thị cửa sổ soạn mã nguồn của class BorderFrame rồi hiệu chỉnh lại mã nguồn của nó như sau :

```

package awtgptb2;
//khai báo các package chứa các tài nguyên cần dùng
import java.awt.*;
import java.awt.event.*;
public class BorderFrame extends FrGPTB2 {
    public BorderFrame() { //constructor của frame
        super();
        //thiết lập LayoutManager cho Frame
        this.setLayout(new BorderLayout());

        //tạo Panel chứa các label và các textbox nhập liệu
        Panel mPanel = new Panel();
        //khai báo layout cho Panel có 3 hàng 2 cột
        mPanel.setLayout(new GridLayout(3, 2));
        //tạo Label và add nó vào Panel
        labelA = new Label("Nhập a : ");
        mPanel.add(labelA);
        //tạo Textbox và add nó vào Panel
        txtA = new TextField("");
        mPanel.add(txtA);

        //tạo Label và add nó vào Panel
        labelB = new Label("Nhập b : ");
        mPanel.add(labelB);
        //tạo Textbox và add nó vào Panel
        txtB = new TextField("");
        mPanel.add(txtB);

        //tạo Label và add nó vào Panel
        labelC = new Label("Nhập c : ");
        mPanel.add(labelC);
        //tạo Textbox và add nó vào Panel
        txtC = new TextField("");
        mPanel.add(txtC);
        //add Panel vào Frame
        this.add(mPanel,BorderLayout.NORTH);
    }
}

```

```

//tạo Panel có 3 cột để chứa Button ở giữa
mPanel = new Panel(); mPanel.setLayout(new GridLayout(1, 3));
mPanel.add(new Label(""));
//tạo Button và add nó vào Panel
btnStart = new Button("Bắt đầu giải");
//định nghĩa hàm xử lý click chuột trên Button
btnStart.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent ae) {
        giaiPTB2(); //gọi hàm giải ptb2
    }
});
mPanel.add(btnStart); //add Button vào Panel
mPanel.add(new Label(""));
mPanel.setSize(30, 25);
this.add(mPanel, BorderLayout.CENTER); //add Panel vào Frame

mPanel = new Panel(); mPanel.setLayout(new GridLayout(3, 1));
//tạo Label kết quả và add nó vào Panel
lblKetqua = new Label("");
mPanel.add(lblKetqua);
//tạo Label x1= và add nó vào Panel
lblX1 = new Label("");
mPanel.add(lblX1);
//tạo Label x2= và add nó vào Panel
lblX2 = new Label("9");
mPanel.add(lblX2);
//add Panel vào Frame
this.add(mPanel, BorderLayout.SOUTH);
}
}

```

10. Hiệu chỉnh lại hàm main() của chương trình như sau để chạy với FlowFrame (chỉ thay đổi tên kiểu của biến mainFrame và tên class FlowFrame của frame cần dùng):

```

public static void main(String[] args) {
    FrGPTB2 mainFrame = new FlowFrame();
    //khởi động nội dung ban đầu cho các đối tượng
    mainFrame.init();
    //hiển thị frame để người dùng sử dụng
    mainFrame.setVisible(true);
}

```

11. Hiệu chỉnh lại hàm main() của chương trình như sau để chạy với GridFrame (chỉ thay đổi tên class GridFrame của frame cần dùng) :

```

public static void main(String[] args) {
    FrGPTB2 mainFrame = new GridFrame();
    //khởi động nội dung ban đầu cho các đối tượng
    mainFrame.init();
    //hiển thị frame để người dùng sử dụng
    mainFrame.setVisible(true);
}

```


12. Hiệu chỉnh lại hàm main() của chương trình như sau để chạy với FlowFrame (chỉ thay đổi tên class GridBagFrame của frame cần dùng) :

```
public static void main(String[] args) {  
    FrGPTB2 mainFrame = new GridBagFrame();  
    //khởi động nội dung ban đầu cho các đối tượng  
    mainFrame.init();  
    //hiển thị frame để người dùng sử dụng  
    mainFrame.setVisible(true);  
}
```

13. Hiệu chỉnh lại hàm main() của chương trình như sau để chạy với FlowFrame (chỉ thay đổi tên class BorderFrame của frame cần dùng) :

```
public static void main(String[] args) {  
    FrGPTB2 mainFrame = new BorderFrame();  
    //khởi động nội dung ban đầu cho các đối tượng  
    mainFrame.init();  
    //hiển thị frame để người dùng sử dụng  
    mainFrame.setVisible(true);  
}
```