



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO



PRACTICA 2 Uso de Java para modo Grafico

Opción círculos líneas y rectángulos

COMPILADORES

Grupo: 3CM17

ALUMNO: MORA GUZMAN JOSE ANTONIO

FECHA ENTREGA: Sabado 23 OCTUBRE 2021

Descripción:

Use la carpeta grafibasi para dibujar círculos, líneas y rectángulos (usar polimorfismo)

circulo NUMERO NUMERO NUMERO
radio x y

linea NUMERO NUMERO NUMERO NUMERO
x1 y1 x2 y2

rectangulo NUMERO NUMERO NUMERO NUMERO
x y ancho alto

Hacer un demo (nombre clave: tele , bocho y changarro)

dibujar un coche use 2 círculos y 2 rectángulos

dibujar una casa con techo de 2 aguas y una puerta use 2 rectángulos y 2 líneas inclinadas 45 y 225 grados

dibujar una persona use un círculo (cabeza), y 4 líneas (1 línea horizontal para los brazos y una vertical para el tronco y 2 líneas inclinadas 45 y 225 grados para las piernas)

guardar las cadenas en un archivo llamado dibujo.txt

Código:

Forma.y

```
inst:  NUMBER {
    ((Algo)$$.obj).inst = maq.code("constpush");
    maq.code(((Algo)$1.obj).simb);
}
| RECTANGULO NUMBER NUMBER NUMBER NUMBER {
    //Push del primer simbolo gramatical      ( X )
    maq.code("constpush");
    maq.code(((Algo)$2.obj).simb);
    //Push del segundo simbolo gramatical    ( Y )
    maq.code("constpush");
    maq.code(((Algo)$3.obj).simb);
    //Push del tercer simbolo gramatical     ( ancho )
    maq.code("constpush");
    maq.code(((Algo)$4.obj).simb);
    //Push del cuarto simbolo gramatical     ( alto )
    maq.code("constpush");
    maq.code(((Algo)$5.obj).simb);
    maq.code("rectangulo");
}
| LINE NUMBER NUMBER NUMBER NUMBER {
    //Push del primer simbolo gramatical      ( X1 )
    maq.code("constpush");
    maq.code(((Algo)$2.obj).simb);
    //Push del segundo simbolo gramatical    ( Y1 )
    maq.code("constpush");
    maq.code(((Algo)$3.obj).simb);
    //Push del tercer simbolo gramatical     ( X2 )
    maq.code("constpush");
    maq.code(((Algo)$4.obj).simb);
    //Push del cuarto simbolo gramatical     ( Y2 )
    maq.code("constpush");
    maq.code(((Algo)$5.obj).simb);
    maq.code("line");
}
```

```

}
| CIRCULO NUMBER NUMBER NUMBER {
    //Push del primer simbolo gramatical ( radio )
    maq.code("constpush");
    maq.code(((Algo)$2.obj).simb);
    //Push del segundo simbolo gramatical ( X )
    maq.code("constpush");
    maq.code(((Algo)$3.obj).simb);
    //Push del tercer simbolo gramatical ( Y )
    maq.code("constpush");
    maq.code(((Algo)$4.obj).simb);
    maq.code("circulo");
}
| COLOR NUMBER {
    maq.code("constpush");
    maq.code(((Algo)$2.obj).simb);
    maq.code("color");
}
;

```

YYLEX()

```

int yylex(){
    String s;
    int tok;
    Double d;
    Simbolo simbo;
    if (!st.hasMoreTokens())
        if (!newline) {
            newline=true;
            return ' ';
        }
        else
            return 0;
    s = st.nextToken();
    try {
        d = Double.valueOf(s);
        yylval = new ParserVal(
            new Algo(tabla.install("", NUMBER, d.doubleValue()),0) );
        tok = NUMBER;
    } catch (Exception e){
        if(Character.isLetter(s.charAt(0))){
            if((simbo=tabla.lookup(s))==null)
                yylval = new ParserVal(new Algo(simbo, 0));
            tok= simbo.tipo;
        } else {
            tok = s.charAt(0);
        }
    }
    return tok;
}

```

Maquina.java

```
void execute(int p){
    String inst;
    System.out.println("progsiz=" + prog.size());
    for(pc=0; pc < prog.size(); pc=pc+1){
        System.out.println("pc=" + pc + " inst " + prog.elementAt(pc));
    }
    for(pc=p; !(inst=(String)prog.elementAt(pc)).equals("STOP") && !returning;){
        //for(pc=p; pc < prog.size();){
        try {
            //System.out.println("111 pc= " + pc);
            inst=(String)prog.elementAt(pc);
            pc=pc+1;
            System.out.println("222 pc= " + pc + " instr " + inst);
            c=this.getClass();
            //System.out.println("clase " + c.getName());
            metodo=c.getDeclaredMethod(inst, null);
            metodo.invoke(this, null);
        }
        catch(NoSuchMethodException e){
            System.out.println("No metodo " + e);
        }
        catch(InvocationTargetException e){
            System.out.println(e);
        }
        catch(IllegalAccessException e){
            System.out.println(e);
        }
    }
}
```

```
void constpush(){
    Simbolo s;
    Double d;
    s=(Simbolo)prog.elementAt(pc);
    pc=pc+1;
    pila.push(new Double(s.val));
}

void color(){
    Color colors[]={Color.red,Color.green,Color.blue};
    double d1;
    d1=((Double)pila.pop()).doubleValue();
    if(g!=null){
        g.setColor(colors[(int)d1]);
    }
}
```

```

void line(){
    double X1, Y1, X2, Y2;
    //Obtenemos el primer valor, haciendo pop de la pila
    X1 = ((Double)pila.pop()).doubleValue();
    //Obtenemos el segundo valor, haciendo pop de la pila
    Y1 = ((Double)pila.pop()).doubleValue();
    //Obtenemos el tercer valor, haciendo pop de la pila
    X2 = ((Double)pila.pop()).doubleValue();
    //Obtenemos el cuarto valor, haciendo pop de la pila
    Y2 = ((Double)pila.pop()).doubleValue();

    //Los gráficos no deben ser nulos para poder dibujar
    if(g!=null){
        //Creamos un objeto Linea con los datos obtenidos de la pila
        ( new Linea((int)X1, (int)Y1, (int)X2, (int)Y2) ).dibuja(g);
    }
}

void circulo(){
    double radio, X, Y;
    //Obtenemos el valor del radio haciendo pop de la pila
    radio = ((Double)pila.pop()).doubleValue();
    //Obtenemos el valor de la posición X haciendo pop de la pila
    X = ((Double)pila.pop()).doubleValue();
    //Obtenemos el valor de la posición Y haciendo pop de la pila
    Y = ((Double)pila.pop()).doubleValue();
    //Para poder dibujar la variable g no debe ser nula
    if(g!=null){
        //Creamos un nuevo objeto circulo
        ( new Circulo((int)radio, (int)X, (int)Y) ).dibuja(g);
    }
}

```

```

void rectangulo(){
    double X, Y, ancho, alto;
    //Obtenemos el valor de la posición en X haciendo pop de la pila
    X = ((Double)pila.pop()).doubleValue();
    //Obtenemos el valor de la posición en Y haciendo pop de la pila
    Y = ((Double)pila.pop()).doubleValue();
    //Obtenemos el valor de la anchura del rectangulo haciendo pop de la pila
    ancho = ((Double)pila.pop()).doubleValue();
    //Obtenemos el valor de la altura dle rectangulo haciendo pop de la pila
    alto = ((Double)pila.pop()).doubleValue();
    if(g!=null){
        ( new Rectangulo((int)X, (int)Y, (int)ancho, (int)alto) ).dibuja(g);
    }
}

```

Pruebas del programa

Se debe compilar con:

- **byaccj -J forma.y**
- **javac Parser.java**
- **java Parser**

y una vez ejecutado meti los siguientes datos para las figuras:

//CASA

color 0 ;

rectangulo 100 100 100 100 ;

color 0 ;

rectangulo 50 25 150 139 ;

color 2 ;

line 100 100 50 150 ;

line 50 150 100 200 ;

//Persona

color 0 ;

circulo 60 100 290 ;

color 2 ;

line 160 315 220 315 ;

line 217 315 240 330 ;

line 217 315 240 300 ;

line 180 295 180 340 ;

//carro

color 0 ;

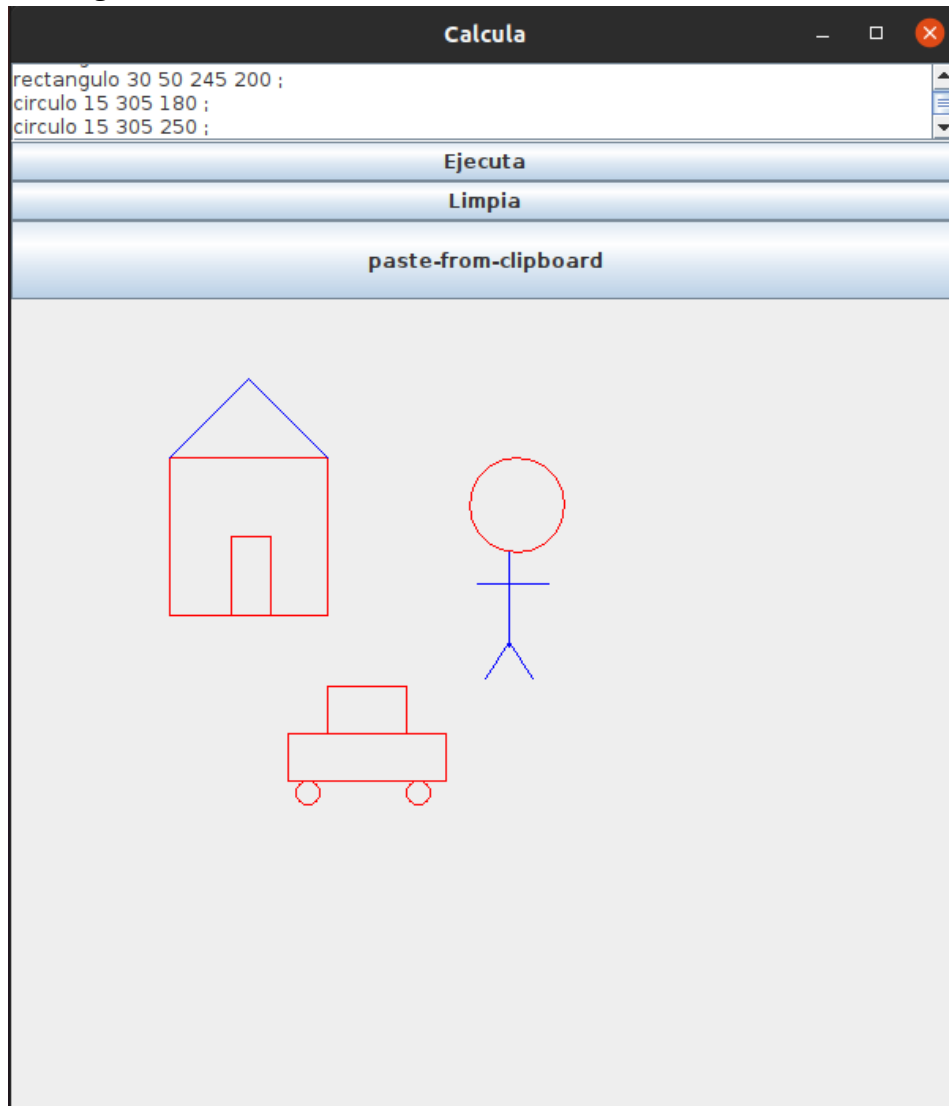
rectangulo 30 100 275 175 ;

rectangulo 30 50 245 200 ;

circulo 15 305 180 ;

circulo 15 305 250 ;

Y el resultado es el siguiente



Conclusion

Esta practica se me hizo complicado en la parte de tener que compilar con java porque no sabia como y despues de investigar un buen rato por distintas fuentes pude hacerlo, el codigo fue relativamente sencillo aunque al momento de hacer los dibujos me tarde poniendo los diseños y coordenadas a modo de prueba y error

Link a prueba:

<https://youtu.be/6l4cxT7wFFE>