

# LENGUAJES Y AUTÓMATAS II

## ME XENIA PADILLA MADRID



ACTIVIDAD DE LABORATORIO	11
NOMBRE DE LA ACTIVIDAD	VARIACIONES DEL IF
OBJETIVO DE APRENDIZAJE	
INSTRUCCIONES	
NOMBRE ESTUDIANTE	Itsai Zempoaltecatl
GRUPO	6SA
FECHA	26/marzo/2022

TRADUCCION:

Gramática:

```
body:
    if_sentence (elif_sentence | else_sentence)? #ifElse
    ;
elif_sentence: ELSE if_sentence (elif_sentence | else_sentence)? #sentenciaElif;
if_sentence: IF P0 condition PC KO body* KC #sentenciaIf;
else_sentence: ELSE KO body* KC #sentenciaElse;
```

Formas de if: if único, if-else acompañado de un else, if{if} if dentro de otro if (if anidado) y if-else if acompañado de un else que a su vez acompaña con if.

En base a estas variaciones cree un lexema que donde verifique si viene acompañada o no de sus variantes, para el if anidado tenemos a if en body que lo visita a si mismo si existe uno.

La variante de if-else-if puede estar acompañado de las mismas variantes que acompañan a if, es por eso que igual puede o no tenerlas

\*La condición sigue igual que en ejercicio de if-else

Visitor

```
@Override
public String visitIfElse(LenguajeParser.IfElseContext ctx) {
    String lineIf = visit(ctx.if_sentence());
    String lineElse="";
    String lineElif="";
    if(ctx.else_sentence()!=null){
        lineElse=visit(ctx.else_sentence());
    }else if(ctx.elif_sentence()!=null){
        lineElif=visit(ctx.elif_sentence());
    }
    return lineIf+ lineElse +lineElif ;
}
```

La primera visita es a IfElse, primero visita a if\_sentence y revisa si alguna de las opciones que lo pueden acompañar no estén nulos, si alguna opción contiene información lo visita, en caso contrario no realiza ninguna acción y devolvería solo el if visitado

# LENGUAJES Y AUTÓMATAS II

## ME XENIA PADILLA MADRID



```
@Override
public String visitSentenciaElif(LenguajeParser.SentenciaElifContext ctx) {
    String lineElIf = "el"+visit(ctx.if_sentence());
    String lineElse="";
    String lineElif="";
    if(ctx.else_sentence()!=null){
        lineElse=visit(ctx.else_sentence());
    }else if(ctx.elif_sentence()!=null){
        lineElif=visit(ctx.elif_sentence());
    }
    return lineElIf+ lineElse +lineElif ;
}
```

La visita Elif (else-if) tiene una logica similar a la visita IfElse ya que else-if puede estar acompañado de else u otro else-if, es por eso que se valida que alguna de estas opciones sea diferente de nula y la visita.

### Resultado

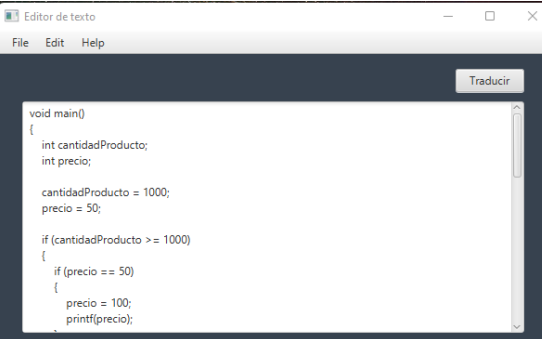
```
void main()
{
    int cantidadProducto;
    int precio;

    cantidadProducto = 1000;
    precio = 50;

    if (cantidadProducto >= 1000)
    {
        if (precio == 50)
        {
            precio = 100;
            printf(precio);
        }
        else
        {
            cantidadProducto = 2000;
            printf(cantidadProducto);
        }
    }
    else if (cantidadProducto <= 2000)
    {
        cantidadProducto = 0;
        precio = 0;
        printf(cantidadProducto);
        printf(precio);
    }
    else
    {
        cantidadProducto = 5000;
        precio = 5000;
        printf(cantidadProducto);
        printf(precio);
    }


    printf(cantidadProducto);
    printf(precio);

    if ((cantidadProducto == precio) && (precio >= 0))
    {
        printf(0);
    }
}
```



Entrada

```
#initial()->
use cantidadProducto;
use precio;
cantidadProducto=1000;
precio=50;
if(cantidadProducto>=1000)->
if(precio==50)->
precio=100;
systalk(precio);
<-else->
cantidadProducto=2000;
systalk(cantidadProducto);
<-
<-elif(cantidadProducto<=2000)->
cantidadProducto=0;
precio=0;
systalk(cantidadProducto);
systalk(precio);
<-else->
cantidadProducto=5000;
precio=5000;
systalk(cantidadProducto);
systalk(precio);
<-
systalk(cantidadProducto);
systalk(precio);
if((cantidadProducto==precio)&&(precio>=0))->
systalk(0);
<-
<-
```



Salida

# LENGUAJES Y AUTÓMATAS II

## ME XENIA PADILLA MADRID

### EJECUCION

#### Gramática:

```
instructions:
    if_sentence (elif_sentence | else_sentence)? #ifElse
    elif_sentence: elif_frag_condition (elif_sentence | else_sentence)? #sentenciaElif;
    elif_frag_condition: ELIF PO condition PC KO body KC #condicionElif;
    if_sentence: IF PO condition PC KO body KC #sentenciaIf;
    else_sentence: ELSE KO body KC #sentenciaElse;
```

Ya que en la traducción sustituyo else-if(...) por elif(...), agregue otro lexema llamado elif\_frag\_condition que se encarga de reconocer al elif y evaluar su condición, esto porque en el visitor esa línea devuelve un valor booleano, y usando la misma lógica de ifElse, creo el lexema de elif\_sentence ya que puede o no venir acompañada de un else, elif o contener un if en su body.

```
@Override
public Object visitIfElse(OpmezParser.IfElseContext ctx) {
    try{
        boolean result= (boolean) visit(ctx.if_sentence());
        if(!result){
            if(ctx.else_sentence()!=null){
                visit(ctx.else_sentence());
            }else if(ctx.elif_sentence()!=null){
                visit(ctx.elif_sentence());
            }
        }
    }catch (Exception e){
    }
    return null;
}
```

Modifique la visita a ifElse para que igual verifique si elif\_sentence existe, si sí, la visito.

```
@Override
public Object visitSentenciaElif(OpmezParser.SentenciaElifContext ctx) {
    try{
        boolean result= (boolean) visit(ctx.elif_frag_condition());
        if(!result){
            if(ctx.else_sentence()!=null){
                visit(ctx.else_sentence());
            }else if(ctx.elif_sentence()!=null){
                visit(ctx.elif_sentence());
            }
        }
    }catch (Exception e){
    }
    return null;
}
```

Como sentenciaElif comparte el mismo formato que ifElse, al visitarlo se realizan las mismas comparaciones, con la diferencia que este obtiene su resultado desde elif\_frag\_condition

# LENGUAJES Y AUTÓMATAS II

## ME XENIA PADILLA MADRID

```
@Override
public Object visitCondicionElif(OpmezParser.CondicionElifContext ctx) {
    try{
        boolean result = (boolean)visit(ctx.condition());
        if(result){
            visit(ctx.body());
        }
        return result;
    }catch(Exception e){
        ps.println("Algo fallo en: elif(\"+ctx.condition().getText()+\")");
        return null;
    }
}
```

condicionElif(elif\_frag\_condition) es similar a condicionIf, igual evalua a su condition y en base al resultado de esta visita el cuerpo, se devuelve el mismo valor obtenido de la condition para usarlo en la visita anterior vista.

Resultados:

```
#initial()->
use cantidadProducto;
use precio;
cantidadProducto=1000;
precio=50;
if(cantidadProducto>=1000)->
if(precio==50)->
precio=100;
systalk(precio);
<-else->
cantidadProducto=2000;
systalk(cantidadProducto);
<-
<-elif(cantidadProducto<=2000)->
cantidadProducto=0;
precio=0;
systalk(cantidadProducto);
systalk(precio);
<-else->
cantidadProducto=5000;
precio=5000;
systalk(cantidadProducto);
systalk(precio);
<-
systalk(cantidadProducto);
systalk(precio);
If((cantidadProducto==precio)&&(precio>0))->
systalk(0);
<-
<-
```

Entrada

Executar

100  
1000  
100

Salida