

Conceptos de Lenguajes de Programación

EXCEPCIONES

Excepciones

Una excepción es una situación anómala que se da en la ejecución de un programa y que se supone que ocurre con poca frecuencia.

Para que un lenguaje trate excepciones debe proveer:

- ✓ Un modo de definirlas
- ✓ Una forma de alcanzarlas, invocarlas
- ✓ Una forma de manejarlas
- ✓ Un criterio de continuación

Modelo de manejo de Excepciones

ADA

TERMINACIÓN

CLU

TERMINACIÓN

PL1

REASUNCIÓN

Excepciones - ADA

- Modelo por TERMINACIÓN
- Variables = identificador -> = alcance
- Hay excepciones predefinidas, como por ejemplo, `constraint_error`, estas NO deben declararse
- ADA tiene propagación dinámica
- Las excepciones son levantadas con la palabra clave `raise[nombre_excepcion]`
- Si el nombre de la excepción no se encuentra entonces se levanta la misma excepción que esa esta manejando, esto solo puede darse dentro de un manejador.
- Se pueden declarar bloques `Declare` donde al final puede haber manejadores.
- Excepciones con el mismo nombre se enmascaran entre si.

Excepciones – ADA – Ejemplo

Prog MAIN

e: exeption

***DECLARACION

...

Begin

...

.. if (condError) then

*** INVOCACION

raise e ** si aca esta raise solo levanta una excepciòn anònima

endif

....

Exception

when e : begin ...

****MANEJADORES

...

... (raise) *** este raise levanta la misma excepcion que esta manejando

End

When others Begin

...

..

End

End //(de main)

Excepciones - CLU

- Modelo de TERMINACIÓN a nivel de sentencia, el proceso que levanta la excepción SIEMPRE termina (aunque tenga manejadores), el proceso que tiene el manejador no termina, la ejecución continua en la línea siguiente a la llamada del proceso que levantó la excepción.
- variables = identificador -> = alcance
- Las excepciones que puede levantar un proceso deben estar indicadas en el encabezado del mismo.
- Los manejadores se encuentran asociados a sentencias y siempre se colocan al lado de la misma o en la línea siguiente.
- Hay excepciones predefinidas, como por ejemplo, `constraint_error...` No deben declararse
- CLU tiene propagación dinámica de UN nivel, es decir el proceso que levanta la excepción termina y busca el manejador en el llamante inmediato. Si la sentencia de invocación al proceso tiene asociado el manejador correspondiente entonces se ejecuta ese manejador y la ejecución continua en la instrucción siguiente a la invocación. Si no hay un manejador asociado a la llamada al proceso, entonces, se busca el manejador en las sentencias que contienen a ese llamado en forma estática. Si así no se encuentra el manejador termina todo en failure.
- Los manejadores en CLU pueden definirse con parámetros.
- Una excepción es levantada con la palabra clave `signal nombre_excepcion`, existe un `resignal` que permite propagar una vez mas en forma dinámica la excepción que se esta manejando, esto se puede hacer solo una vez.

Excepciones – CLU – Ejemplo

Procedure Main

Error1 : exception;

x, y: integer;

Procedure UNO() signals error1; // en el encabezado se deben poner los manejadores que un proceso puede levantar

x:integer

Begin

x:=2;

While y < x Do

If y=0 Then signal error1; //se levanta la excepcion error

end if; exception when error1 -> y:=y+7; x:=x+2; resignal; end; // manejador4

Dos();

y:=y+1;

Wend; exception when error1 -> y:=x+3; x:=x+3; Resignal; .End; //manejador3

End; //UNO

Procedure Dos() signals error1;

m:integer;

Begin

...

if m=0 then signal error1; //se levanta la excepcion error

End;

Begin //MAIN

x:=1; y:=0;

Uno(); exception when error1 -> x:=x+1; y:=y+1; end; // manejador1

...

Dos(): exception when error1 → resignal; end; // manejador2

...

End; //MAIN

Excepciones - PL1

- Modelo de REASUNCIÓN
- Ningún proceso termina cuando se levanta una excepción, simplemente se detecta la excepción, se la atiende y continúa el flujo de ejecución en el punto siguiente a donde se levantó la excepción.
- Se definen con la instrucción ON CONDITION nombre Manejador
- Se levantan con la instrucción SIGNAL CONDITION nombre
- Alcance de una excepción y búsqueda del manejador: a medida que se va ejecutando el proceso y se encuentran ON CONDITION se apila el manejador en una pila de manejadores, cuando se levanta una excepción se busca el nombre de la misma desde el tope de la pila hacia abajo, el primer manejador que se encuentra es el que se ejecuta.
- Dentro de un bloque puede haber n cantidad de manejadores en cualquier lugar.
- Excepciones con el mismo nombre se enmascaran entre si.

Excepciones – PL1 – Ejemplo

Prog Main

PROC UNO

Begin

...

ON CONDITION PEPE begin ... end; ***DECLARACION y MANEJADOR3

..

If (condError) then

 SIGNAL CONDITION PEPE *** INVOCACION

end

....

Begin

...

ON CONDITION PIPO begin ... end; ***DECLARACION y MANEJADOR1

..

ON CONDITION PEPE begin ... end; ***DECLARACION y MANEJADOR2

...

UNO;

...

If (condError) then

 SIGNAL CONDITION PIPO

end