

SISTEMAS DE TIEMPO REAL

EJERCICIO 2. ESTRUCTURAS DE CONTROL EN ADA

JUAN MIGUEL HERRADA ACOSTA
JEFFERSON MAX TOMALA VILLARREAL

EJERCICIO 1. INSTRUCCIONES Y ESTRUCTURAS DE CONTROL

Escribir un programa en ADA con las siguientes características:

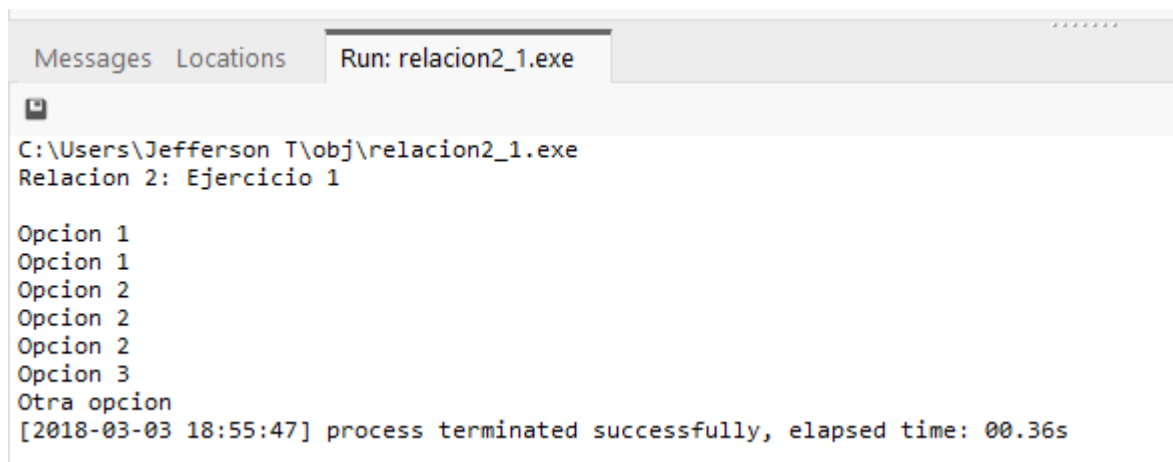
1. Inicializar una variable tipo cadena que contenga los caracteres "ABCDEFGH".
2. Realizar un bucle para extraer la información de la cadena anterior carácter a carácter e imprimir por pantalla en función del carácter leído:
 - "A" o "B": mostrar "Opción 1".
 - "C", "D" o "E": mostrar "Opción 2".
 - "F": mostrar "Opción 3".
 - En otro caso: mostrar "Otra opción".

Nombre del fichero: relacion2_1.adb

```
with Ada.Text_IO;
use Ada.Text_IO;
procedure relacion2_1 is
  --Creamos la cadena y declaramos la variable char.
  --cadena : constant String(1..7) := "ABCDEFGH";
  cadena : String := "ABCDEFGH";
  char : Character;

begin
  Put_Line("Relacion 2: Ejercicio 1");
  New_Line(1);
  -- Recorremos la cadena con un bucle
  for i in cadena'Range loop
    char:=cadena(i);
    --Recorremos la cadena carácter a carácter y comprobamos a través de
    --una estructura de selección por casos que tipo de carácter es y en
    ---consecuencia se imprimirá por pantalla un mensaje
    case char is
      when 'A' | 'B' => Put_Line("Opcion 1");
      when 'C' | 'D' | 'E' => Put_Line("Opcion 2");
      when 'F' => Put_Line("Opcion 3");
      when others => Put_Line("Otra opcion");
    end case;
  end loop;

end relacion2_1;
```



```
Messages Locations Run: relacion2_1.exe
C:\Users\Jefferson T\obj\relacion2_1.exe
Relacion 2: Ejercicio 1

Opcion 1
Opcion 1
Opcion 2
Opcion 2
Opcion 2
Opcion 3
Otra opcion
[2018-03-03 18:55:47] process terminated successfully, elapsed time: 00.36s
```

Declaramos una cadena que contiene “ABCDEFGG”, luego se crea un bucle para recorrer la cadena y para cada carácter se comprueba el tipo de opción y luego se muestra por pantalla.

3. OPCIONAL: Extender el código anterior para admitir una cadena de entrada por teclado.

Nombre del fichero: relacion2_1opcional.adb

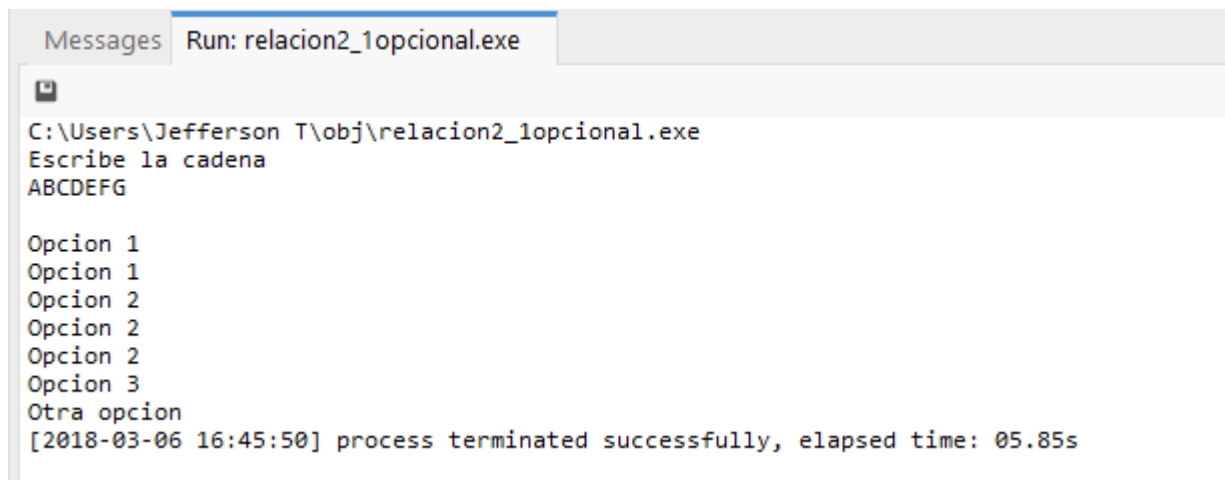
```
with Ada.Text_IO;
use Ada.Text_IO;

procedure relacion2_1opcional is
  --Declaramos las variables
  aux:String(1..20);
  tamaño:Natural;
  caracter:Character;
begin

  Put_Line("Escribe la cadena");
  --solicita una cadena por consola
  Get_Line(aux,tamaño);
  New_Line;
  --
  for i in 1..tamaño loop
    caracter:=aux(i);
    --Recorremos la cadena carácter a carácter y comprobamos a través de
    --una estructura de selección por casos que tipo de carácter es y en
    ---consecuencia se imprimirá por pantalla un mensaje
    case caracter is
      when 'A' | 'B' => Put_Line("Opcion 1");
      when 'C' | 'D' | 'E' => Put_Line("Opcion 2");
      when 'F' => Put_Line("Opcion 3");
      when others => Put_Line("Otra opcion");
    end case;

  end loop;

end relacion2_1opcional;
```



```
Messages Run: relacion2_1opcional.exe
C:\Users\Jefferson T\obj\relacion2_1opcional.exe
Escribe la cadena
ABCDEFGG

Opcion 1
Opcion 1
Opcion 2
Opcion 2
Opcion 2
Opcion 3
Otra opcion
[2018-03-06 16:45:50] process terminated successfully, elapsed time: 05.85s
```

Declaramos una variable aux para guardar la el string que se pase por consola, luego se crea un bucle para recorrer la cadena y para cada carácter se comprueba el tipo de opción y luego se muestra por pantalla.