
Práctico 4: Variables

Objetivos:

Al finalizar este práctico se espera que los alumnos:

- comprendan el concepto de variable y la operación de asignación,
- reconozcan los distintos tipos predefinidos de Pascal,
- utilicen variables y distingan el alcance de las mismas,
- Entiendan la diferencia entre parametros y variables de locales a un procedimiento.

1) Realice un procedimiento para Intercambiar los valores de dos variables enteras. ¿Cuáles serían las diferencias si las variables fueran de tipo caracter?

2) Realice un procedimiento para sumar los N primeros números naturales. (*Ejemplo si $N=5$ se debe responder con la suma de $1+2+3+4+5$*)

3) Realice un procedimiento para colocar en una variable la posición que tiene un elemento dado en la fila ORIGINAL. Si el elemento no está en la fila la posición sera -1

4) El siguiente programa intenta ser una solución a este problema: *Se tiene una fila CHEQUES, donde cada elemento representa el importe de un cheque, y otra SOCIOS, donde cada elemento representa el número de carnet de un socio. Determinar cuánto recibe cada socio si se distribuye equitativamente el total del dinero de los cheques.*

a) cumple el objetivo? b) Si tiene errores, marcarlos y corregirlos

```
Program DistribucionSocios;
{$INCLUDE/IntroProg/Estructu}
procedure CantElementos(FilaAnalizada:fila; total:integer);
begin
    while not filavacia(FilaAnalizada) do
    begin
        agregar(Aux,extraer(FilaAnalizada));
        total:= total+ 1;
    end;
end;
Cheques, Socios:fila;
Personas, Importe:integer;
PlataSocio:real;
begin
    Readfila(Cheques);
    Readfila(Socios);
    CantElementos(Socios,Personas);
    CantElementos(Cheques,Importe);
    PlataSocio:=Importe/Personas;
    Writeln('La plata que recibirá cada socio sera:' PlataSocio);
end.
```

5) Realice un procedimiento para: dada la variable entera Posición, eliminar de la pila Secuencia el elemento que se encuentre en dicha posición(si la posición es inválida no hace nada).

6) Utilizando el procedimiento del ejercicio anterior resuelva: Dadas dos pilas (ORIGINAL y

POSICIONES), eliminar de ORIGINAL todos los elementos que POSICIONES indica, usando los procedimientos anteriores. Note que al eliminar un elemento los restantes disminuyen en uno su ubicación. Contemple este caso para que los corrimientos no afecten su algoritmo.

7) En base a una fila NUMEROS (que contiene al menos un elemento), generar una pila RESULTADO donde cada elemento será la suma de los valores de NUMEROS ubicados en una posición menor o igual al mismo.

El primer elemento de NUMEROS corresponde a la posición 1. Por ejemplo, dada la fila NUMEROS: <último> 3 4 5 1 2 <primero>

RESULTADO será: <base> 15 12 8 3 2 <tope>

8) Considerar que una pila DADA está formada por Secuencias, que son números no nulos, separados entre sí por un cero. Codificar el procedimiento ExtraerSecuencia que recibe en el parámetro DADA la pila de la que extrae, y retorna en la pila NUEVA la primer secuencia de elementos no nulos de DADA. Usando el procedimiento ExtraerSecuencia, codificar el procedimiento CantidadSecuencias que recibe como parámetro la pila DADA, y que devuelve en el parámetro entero Cantidad el número de secuencias de la pila DADA.

9) Dada la fila DATOS y dos variables POSICION y VALOR debe implementar un procedimiento que actualice la fila DATOS de manera que al número existente en la POSICION lo incremente con VALOR. (este procedimiento no debe fallar si la posición no es válida)

Ejemplo: $\rightarrow 4\ 6\ 3\ 7\ 5 \rightarrow$ Posición=2, Valor=9, resultado $\rightarrow 4\ 6\ 3\ 16\ 5 \rightarrow$

Luego utilizando ese procedimiento realice un programa principal que solicite la fila DATOS y las pilas POSICIONES y VALORES al usuario e incremente en DATOS todos los números existentes en POSICIONES por sus VALORES respectivos. Las tres estructuras pueden tener cero o más elementos y no tienen que coincidir en la cantidad de los mismos. Finalmente debe mostrar el resultado de la fila DATOS. Ejemplo:

DATOS = $\rightarrow 4\ 6\ 3\ 7\ 5 \rightarrow$, POSICIONES = | 3 5 2 \leftrightarrow , VALORES = | 10 20 30 \leftrightarrow

(Resultado: DATOS = $\rightarrow 24\ 6\ 13\ 37\ 5 \rightarrow$)

Aclaraciones: El proceso termina cuando POSICIONES O VALORES no tienen más datos. Si la posición no es válida no debe fallar y debe pasar a la siguiente (pasando también al siguiente valor, es decir no tomando en cuenta el valor de la posición inválida).

Por ejemplo: DATOS = $\rightarrow 4\ 6\ 3\ 7\ 5 \rightarrow$, POSICIONES = | 5 2 \leftrightarrow , VALORES = | 10 20 30 \leftrightarrow

(Resultado: DATOS = $\rightarrow 24\ 6\ 3\ 37\ 5 \rightarrow$)