

Parcial - Programación 1
Instituto de Computación - Facultad de Ingeniería
Diciembre 2020

Leer con atención

- Todos los programas o fragmentos de programas deben ser escritos en el lenguaje **Pascal** tal como fue dado en el curso.
- En todos los problemas se evaluará, además de la lógica correcta, la utilización de un buen estilo de programación de acuerdo a los criterios impartidos en el curso.
- Entregue solamente las hojas de solución escritas a lápiz.

Ejercicio 1 (10 pts)

Dado el siguiente programa, escribir cuál será su salida cuando la variable x se carga de la entrada estándar con **el último dígito de su CI** (antes del dígito verificador). Por ejemplo, si su CI es 1.234.567-8, el último dígito es 7:

```
program Ejercicio4;

var x, y, z: integer;

procedure calcular (a : integer; var b : integer);
function sumar (a, b : integer) : integer;
var i : integer;
begin
    for i := 1 to 3 do
        a := a + i;
        sumar := a + b
    end;
begin
    b := sumar ( b, b);
    a := b - a;
    writeln (a, b);
    x := a + 3
end;
begin
    readln(x);
    y := x + 3;
    z := x div 2;
    calcular (x, y);
    writeln(x, y, z)
end.
```

Ejercicio 2 (28 pts)

Se define el siguiente tipo para listas cuyas celdas pueden contener enteros, reales o caracteres:

```
type TCaso = (creal, cint, cchar);
RealIntChar = record case caso : TCaso of
    creal : (vreal : real);
    cint : (vint : integer);
    cchar : (vchar : char)
end;

ListaH = ^TCeldaH;
TCeldaH = record
    info : RealIntChar;
    sig : ListaH
end;
```

Escribir la siguiente función producto, que retorna el producto de todos los números de la lista incluyendo enteros y reales. Si la lista no contiene ningún número, el resultado del producto debe ser 1. Notar que si alguna de las celdas contiene un 0 el resultado del producto será 0.

```
function producto (lista : listaH) : real;
```

Ejemplos:

- lista = (caso=cint, vint=2) → (caso=cchar, vchar='a') → (caso=creal, vreal=3.0), resultado = 6.0
- lista = (caso=cint, vint=2) → (caso=cint, vint=0) → (caso=creal, vreal=3.0), resultado = 0.0
- lista = (caso=cchar, vchar='2') → (caso=cchar, vchar='a'), resultado = 1.0
- lista = vacía, resultado = 1.0

Ejercicio 3 (50 pts)

Dadas las siguientes definiciones para textos y sus estadísticas:

```
const MAXTXT = (* entero mayor que 0*);

type TLetra = 'A' .. 'Z';

TTexto = record
    txt : array [1 .. MAXTXT] of TLetra;
    tope : 0 .. MAXTXT;
end;

TEstadistica = record
    cantidades : array [TLetra] of 0 .. MAXTXT;
    porcentajes : array [TLetra] of real;
    mas, menos : TLetra;
end;
```

Parte a) – 30 pts

Escribir un procedimiento estadisticas, que dado un texto **no vacío** texto retorna en est las estadísticas de ese texto, que se componen por la cantidad y el porcentaje de veces que aparece cada letra, y cuáles son las letras que aparecen más y menos veces (si es más de una letra, la primera de ellas en orden alfabético).

```
procedure estadisticas (texto : TTexto; var est : TEstadistica);
```

Ejemplo:

- texto = "CAECHGDD"
- est = (cantidades = ['A'=1, 'B'=0, 'C'=2, 'D'=2, 'E'=1, 'F'=0, 'G'=1, 'H'=1, el resto todos 0],
porcentajes = ['A'=12.5, 'B'=0.0, 'C'=25.0, 'D'=25.0, 'E'=12.5, 'F'=0.0, 'G'=12.5, 'H'=12.5, el resto todos 0.0],
mas = 'C', menos = 'B')

Parte b) – 20 pts

Escribir una función masDeUnMaximo, que dada una estadística est calculada como se indica en la parte anterior, retorna true si existe más de una letra que se repita el número máximo de veces y false en caso contrario. En otras palabras, verifica si hay más letras que se repitan la misma cantidad de veces que est.mas.

```
function masDeUnMaximo (est : TEstadistica) : boolean;
```

En el ejemplo de la parte anterior retorna true, porque 'D' se repite la misma cantidad de veces que 'C'.