

**Universidad Tecnológica Nacional**

**Carrera: Tecnicatura Universitaria en Programación**

**Tipo de evaluación: Primer evaluación parcial**

**Cátedra: Programación II**

**Profesor: Figueredo, Fernando Ariel**

<b>Contenidos a evaluar</b>
-----------------------------

Unidad 1

<b>Criterios de Evaluación</b>
--------------------------------

- Aplicar los conocimientos de las herramientas y usar los procedimientos adecuados.
- Claridad Conceptual y uso del vocabulario técnico requerido en la cátedra
- Producción escrita clara y adecuada.
- Uso correcto de las estructuras y tipos de datos.

<b>Consignas</b>
------------------

1. Calcule la complejidad de los siguientes subprogramas. **(20 pts)**

**FUNCION APUESTA(v : VECTOR(ENTERO), n: ENTERO) : ENTERO ES**

```
    fin : boolean;
    fin:= false;
    apuesta, gana : real;
    n: entero;
    gana := 100;
    apuesta := gana / 2;
    LEER(n)
    MIENTRAS (n >= 0) and (fin=false) HACER
        gana := gana + vector[n];
        n := n - 1;
        apuesta:=apuesta/2;
        leer(fin);
    FINMIENTRAS
    DEVOLVER gana;
FINFUNCION
```

2. Dado el siguiente algoritmo recursivo: **(20 pts)**

```

entero expR (entero base, entero exp)
{
    entero res;
    if (exp == 0) {
        res:= 1;
    }
    else
        res=base * expR (base, exp - 1);
}
return res;
}

```

- a) Diga que retornan las siguientes llamadas:
  - i. `expR (2,3)`
  - ii. `expR (3,4)`
- b) cual creo ud es el cálculo que resuelve la función

3. Dado el siguiente algoritmo recursivo: **(60 pts)**

- a) cargue dos matrices (A y B) de 3X3 con valores enteros comprendidos entre 1 y 10.
- b) Genere un matriz resultado (C) también de 3x3, donde el contenido de cada una de sus celdas será igual a la suma de las celdas homónimas en A y B en el caso de que el valor de la celda en A sea menor que el de la celda en B; y en caso de que el valor de la celda en A sea mayor o igual al valor de la celda en B se deberá restar.
- c) Muestre en pantalla el contenido de la matriz resultante.

Ejemplo:

A			B			C		
1	2	3	2	3	3	3	5	0
4	5	6	1	5	6	3	0	0
7	8	9	8	7	4	15	1	5