

# Examen - Programación 1

## Instituto de Computación

### Diciembre 2024

#### 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.
- Escribir las respuestas de un solo lado de la hoja. Entregar solamente las hojas de solución escritas a lápiz.
- En cada hoja entregada se debe incluir nombre, cédula y qué número de hoja es. En la primera hoja se debe incluir además la cantidad total de hojas entregadas.

#### Ejercicio 1 (44 puntos)

Dadas las siguientes declaraciones:

```
const M = ...; { M > 0 }
type
  Lista      = ^TipoCelda;
  TipoCelda = record
    dato : integer;
    sig  : Lista
  end;

  ArrTope = record
    elems : array [1..M] of integer;
    tope  : 0 .. M
  end;
```

#### a) (20 puntos)

Escriba el procedimiento:

```
procedure eliminarPrimerParLst (var lst : Lista);
```

que elimina el primer elemento de la lista `lst` que sea par. Asuma que la lista tiene al menos un elemento.

Ejemplo:

Si `lst` al momento de invocar al procedimiento es `5 → 1 → 4 → 7 → 6 → 3`,  
luego de la invocación será `5 → 1 → 7 → 6 → 3`.

#### b) (24 puntos)

Escriba el procedimiento:

```
procedure eliminarParesArr (var arr : ArrTope);
```

que elimina **todos** los elementos del arreglo con `tope arr` que sean pares.

**Sugerencia:** Para poder realizar todo en una sola recorrida utilice dos índices: uno para recorrer cada elemento y otro para implementar la eliminación.

## Ejercicio 2 (16 puntos)

Considere el siguiente programa

```
program alcance;
var a, b: integer;

procedure p(a: integer; var b: integer);

    function f(b: integer): integer;
    begin
        ...
    end;

begin
    a := a * 3;
    b := f(b);
    a := a * b;
end;

begin
    read(a, b);
    p(b, a);
    writeln('El triple del primer valor leído es: ', a)
end.
```

en el que no se incluye la implementación de la función `f`. Implemente la función `f`, con una única línea de código, para que el programa imprima el triple del primer valor leído.

Ejemplo: Si la entrada es `2 5`, el programa deberá imprimir `El triple del primer valor leído es: 6`

## Ejercicio 3 (40 puntos)

El ajedrez es un juego que se juega sobre un tablero cuadrulado de  $N \times N$  casillas (en realidad siempre  $N$  es 8, pero la estructura es más general) y que juegan dos jugadores, desplazando piezas sobre él. Las piezas de uno de los jugadores son blancas y la del otro negras. Al principio del juego cada jugador tiene dieciséis piezas: un rey, una dama, dos alfiles, dos caballos, dos torres y ocho peones. Cada jugador puede capturar piezas del otro, retirándolas así del juego.

Dada la siguiente estructura para representar al juego:

```
const N = ...; { N > 0 }
type
  TipoPieza = (Rey, Dama, Alfil, Caballo, Torre, Peon);
  Color     = (Blanco, Negro);

  Pieza     = record
                tipo : TipoPieza;
                color : Color
            end;

  Casilla = record
                case ocupada : boolean of
                    true  : ( p : Pieza) ;
                    false : ()
            end;

  Tablero  = array [1..N,1..N] of Casilla;
  SetPiezas = array [TipoPieza] of integer;
```

### a) (15 puntos)

Implemente una función que dado un `Tablero` y un `Color`, retorne la cantidad de piezas de ese color que quedan en el tablero

```
function cantidadColor(t : Tablero; c : Color) : integer;
```

### b) (25 puntos)

Implemente una función que determina si un tablero tiene una configuración de piezas válida, es decir que de cada color tiene como máximo: un rey, una dama, dos alfiles, dos caballos, dos torres y ocho peones.

**Sugerencia:** Utilice un `SetPiezas` auxiliar para cada color, para llevar el conteo de piezas de ese color en el tablero.

```
function tableroValido(t : Tablero) : boolean;
```