

1. Obtener el tiempo de ejecución $t(n)$, $O(t(n))$, $\Omega(t(n))$ y $\Theta(t(n))$ para el siguiente algoritmo de multiplicación de matrices.

```

for i=1 to n
  for j=1 to n
    suma = 0
    for k=1 to n
      suma = suma + a[i,k] * b[k,j]
    endfor
    c[i,j] = suma
  endfor
endfor

```

$$\begin{aligned}
 t(n) &= \sum_{i=1}^n \left(\sum_{j=1}^n \left(1 + \sum_{k=1}^n (1) + 1 \right) \right) \\
 &= \sum_{i=1}^n \left(\sum_{j=1}^n (2 + n) \right) \\
 &= \sum_{i=1}^n \left(\sum_{j=1}^n (2) + \sum_{j=1}^n (n) \right) \\
 &= \sum_{i=1}^n (2n + n^2) \\
 &= \sum_{i=1}^n (2n) + \sum_{i=1}^n (n^2) = 2n^2 + n^3
 \end{aligned}$$

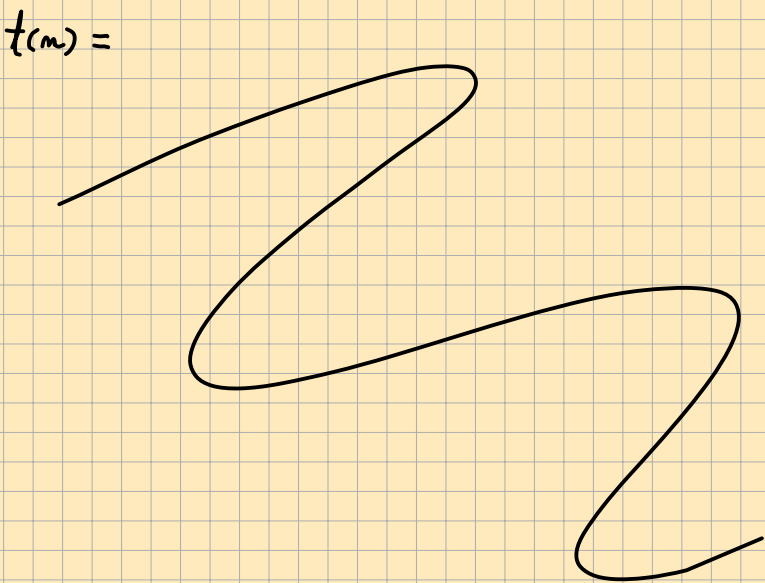
$t(n) = n^3$
 $O(n) = n^3$
 $\Omega(n) = n^3$
 $\Theta(n) = n^3$

2. Obtener O y Ω , y Θ del tiempo promedio para el algoritmo de búsqueda binaria.

```

- i = 1
- j = n
repeat
  m = (i + j) div 2
  if a[m] > x
    j = m - 1
  else
    i = m + 1
  end-if
until i > j or a[m] = x

```



14. Dado el siguiente algoritmo :

```

var
  i, j: integer;
begin
  for i:=0 to n-1 do
    for j:= i+1 to n do
      A[i,j] := 2*A[j,i] + A[i,j];
    end;
  end;
end;

```

$$\begin{aligned}
 t(n) &= \sum_{i=0}^{n-1} \left(\sum_{j=i+1}^n (1) \right) \\
 &= \sum_{i=0}^{n-1} (1 \cdot (n - (i+1) + 1)) \\
 &= \sum_{i=0}^{n-1} (n - i) = \sum_{i=0}^{n-1} (n) - \sum_{i=0}^{n-1} (i) = n^2 - \frac{n^2 - n}{2} \\
 &= \frac{2n^2}{2} - \frac{n^2}{2} + \frac{n}{2} = \frac{n^2 + n}{2}
 \end{aligned}$$

1 EXAMEN_SAMU Calcula el tiempo mejor, peor y promedio.

```

a) for (int i=0; i<N; i++){
  if (datos[i] % 3 == 0){
    x++;
  }
}

```

$$\begin{aligned}
 t(n) &= \sum_{i=1}^n (1 + p_a \cdot t_a + p_g \cdot t_g) \\
 &= \sum_{i=1}^n (1 + \frac{1}{3} \cdot 1 + \frac{2}{3} \cdot 0) \\
 &= \sum_{i=1}^n (1 + \frac{1}{3}) =
 \end{aligned}$$

1) (3 puntos) Dado el código:

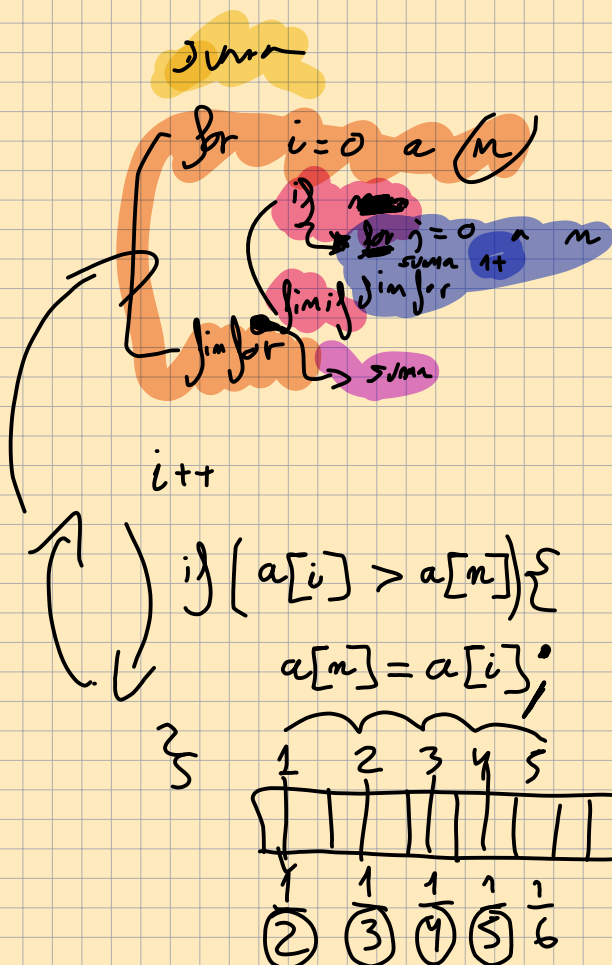
```

i = 1; aux=0;
while i<=n
  index = M[1,i];
  k=i+1;
  while k<=n and index!=M[k,i]
    if index < M[k,i], index = M[k,i]; else index = M[k,i]; endif
    j=1;
    while j<= n
      if (M[k,j]>aux), aux++; endif
      j=j*2;
    endwhile
    k++;
  endwhile
  i++;
endwhile

```

$$t(n) = 1 + \sum_{i=1}^n (2 + r$$

donde M :array[1..n + 1, 1..n + 1] de enteros, estudiar los órdenes O , Ω y Θ de $t(n)$.



$$\begin{aligned}
 t(n) &= 1 + \sum_{i=0}^n (t_{i,j} + 1) \\
 t_p(n) &= 1 + \sum_{i=0}^n \left(\left(\frac{t_a}{2} \cdot p_a + \frac{t_g}{2} \cdot p_g \right) + 1 \right) \\
 &= 1 + \sum_{i=0}^n \left(\sum_{j=0}^n (1) \cdot \frac{1}{i+1} + 1 \right) \\
 &= 1 + \sum_{i=0}^n \left(\frac{n}{i+1} + 1 \right) =
 \end{aligned}$$

