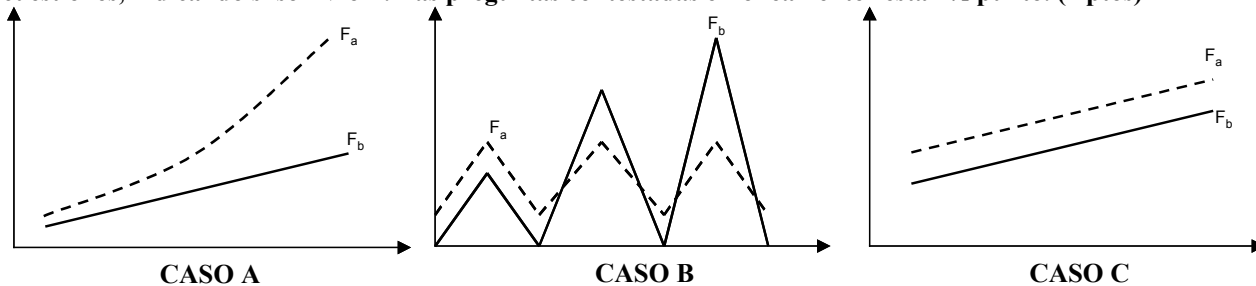


CUESTIONES (1:30 h)

1) Responde a las siguientes cuestiones, indicando si son Verdaderas o Falsas. Las preguntas contestadas erróneamente restan ½ punto. (2 pts)

- Los algoritmos se diferencian de un conjunto de reglas en que las instrucciones de los algoritmos definen claramente las acciones a ejecutar por el procesador. V – F
- Un algoritmo y un programa son la misma cosa V – F
- Un mismo programa puede tener un coste espacial o temporal diferente si se ejecuta sobre ordenadores diferentes V – F
- El coste espacial de un algoritmo se define como el espacio de memoria ocupado por el algoritmo y sus variables antes, durante y después de su ejecución V – F
- El tamaño de un problema nunca puede ser igual al número de instancias del problema V – F
- El número de instancias de un algoritmo depende del tipo de datos que utiliza el algoritmo, por ejemplo si trabaja con variables, con vectores o con matrices, y no de lo que haga el algoritmo V – F
- El método de la instrucción crítica lo podemos utilizar en el cálculo del coste medio, sólo si los costes mejor y peor del algoritmo son iguales V – F
- Un algoritmo tendrá una única instancia si y sólo si su coste en el caso mejor es igual al coste en el caso peor V – F

2) Dadas las siguientes gráficas, que representan el coste de dos funciones a y b. Responde a las siguientes cuestiones, indicando si son V o F. Las preguntas contestadas erróneamente restan ½ punto. (2 pts)



- | | | |
|---|----------------------------------|---|
| a. $F_b \in O(F_a)$ (V – F) | d. $F_b \in \Omega(1)$ (V – F) | g. $F_a = F_b$ (V – F) |
| b. $F_a \in \Omega(F_b)$ (V – F) | e. $F_a \in \Theta(1)$ (V – F) | h. $\Omega(F_a) \subset O(F_b)$ (V – F) |
| c. $O(F_b) \subset \Theta(F_a)$ (V – F) | f. $F_a \in \Theta(F_b)$ (V – F) | i. $\Theta(F_a) = \Theta(F_b)$ (V – F) |

3) Demuestra por inducción que $\sum_{i=1}^n i^2 = \frac{n \cdot (n+1) \cdot (2n+1)}{6}$ (o lo que es lo mismo:

$$\sum_{i=1}^n i^2 = \frac{2n^3 + 3n^2 + n}{6} \text{), indicando el principio de inducción utilizado. (2 pts)}$$

- Tipo de inducción:
- Demostración:

4) Considera los siguientes esquemas algorítmicos y responde a las preguntas (2 pts)

ESQUEMA ALG 1

METODO:

```
    si <condición de fin> entonces <solucion>
    sino
        F ← <elementos posibles>
        mientras <existan elementos de F> ^ !<solucion> hacer
            e ← <selecciona elemento de F>
            F ← F - {e}
            <añade elemento al resultado>
            si <resultado factible> entonces
                ALG1(n-1)
                si !<solucion> entonces <elimina elemento del resultado>
            fsi
        fmientras
    fsi
fALG1
```

ESQUEMA ALG 2

METODO:

```
    Repetir
        F ← <elementos posibles>
        si <existen elementos de F> entonces
            e ← <selecciona elemento de F>
            <añade elemento al resultado>
        fsi
    hasta que !<existen elementos de F> v <solucion>
fALG2
```

ESQUEMA ALG 3

METODO:

```
    si <condición de fin> entonces <solucion particular>
    sino
        <resultado> = <resultado> + ALG 3( $\lfloor n/2 \rfloor$ )
    fsi
fALG3
```

a) Indica el tipo de algoritmo de que se trata: Recursivo, Divide y Vencerás, Voraz o Vuelta a atrás.

ALG 1:

ALG 2:

ALG:3

b) Indica el orden del Coste Espacial (S) de cada algoritmo, suponiendo que el tamaño de la entrada es n y la condición de fin se produce cuando $n \leq 1$.

$S_{ALG1}(n)$:

$S_{ALG2}(n)$:

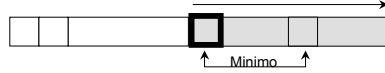
$S_{ALG3}(n)$:

c) Ordena de menor a mayor los algoritmos anteriores según posible coste temporal (no es necesario que calcules ni escribas el coste temporal de ninguno de ellos)

5) Responde a las siguientes preguntas (2 pts)

a) Escribe una versión **RECURSIVA** del algoritmo de ordenación por Selección Directa. Recuerda que el funcionamiento del algoritmo es el siguiente:

Selección Directa: Se recorre el algoritmo de Izq a Dch y se va generando un sub-vector ordenado, seleccionando el elemento mínimo del sub-vector no ordenado y situándolo en la primera posición de dicho sub-vector.



b) Calcula el coste temporal, en asignaciones sobre el vector, para el algoritmo anterior.

c) En el caso del algoritmo iterativo ¿los coste temporales, en comparaciones sobre el vector, para los casos Mejor, Peor y Medio son iguales o diferentes?, ¿Y en el caso de la versión recursiva?

NOMBRE: _____ APELLIDOS: _____

PROBLEMAS (1:30)**Utiliza 1 hoja para cada problema.****6) Resuelve las siguientes ecuaciones de recurrencias. (5 pts)**

$$a) T(n) = \begin{cases} n & n \leq 1 \\ 4T(n-2) & n > 1 \end{cases}$$

$$b) T(n) = \begin{cases} 0 & n \leq 1 \\ 3T\left(\frac{n}{2}\right) + 3 & n > 1 \end{cases}$$

7) Dado el siguiente algoritmo. Responde a las diferentes cuestiones (5 pts)

ALGORITMO EXAMEN

DATOS: A:vector[1..n] de {0,1,2}

RES: R:N

METODO:

R ← 0

para i ← 1 hasta n-2 hacersi A[i] = 2 entoncespara j ← 1 hasta A[i] hacer // Cada vez que se ejecuta el bucle 'para'

R ← R + A[i+j] // se realiza 1 comparación con el vector.

fparafsifpara

fEXAMEN

a) Indica cual es el tamaño del problema y el número de instancias.

b) Indica cual es el caso mejor y calcula el coste temporal en asignaciones y comparaciones sobre el vector

c) Indica cual es el caso peor y calcula el coste temporal en asignaciones y comparaciones sobre el vector

d) Calcula el coste temporal en asignaciones y comparaciones sobre el vector para el caso medio teniendo en cuenta que todas las cifras del 0 al 2 son equiprobables. (Es decir, todas tienen la misma □ probabilidad de darse)

NOTA.- Cotas asintóticas más usuales:

$$\sum_{i=1}^n i = \frac{n}{2} \cdot (n+1) \quad ; \quad \sum_{i=1}^n r^i = \frac{(r^{n+1} - r)}{(r-1)} \quad ; \quad \sum_{i=1}^n 2^{i-1} = n2^n - 2^n + 1 \quad ; \quad \sum_{i=1}^n 2^{-i} = 2 - \frac{n}{2^n} - \frac{2}{2^n}$$