UNIVERSIDAD DE GRANADA

Departamento de Ciencias de la Computación e Inteligencia Artificial

Fundamentos de Programación (2017/18)
1º GII / GII-M / GII-ADE
Conv. Ordinaria - 19 de Enero de 2018

Normas para la realización del examen: **Duración: 2.5 horas**

- **Debe disponer de un documento oficial que acredite su identidad a disposición del profesor.**
- **El único material permitido durante la realización del examen es un bolígrafo o lápiz azul o negro.**
- **No olvide escribir su nombre completo y grupo en todos y cada uno de los folios que entregue.**
- **Dispone de 30 minutos para decidir si no se presenta al examen (y no le cuenta convocatoria)**

◁ Ejercicio 1 ▷ El Problema de la Asignación [2.5 puntos]

Una empresa de servicios dispone de n técnicos y n pedidos por atender. La empresa dispone de una matriz $B^{n \times n}$ de tarifas donde cada b_{ij} indica el precio que cobra el técnico i por atender el pedido j. Supondremos que b_{ij} es un entero mayor estricto que 0 y menor o igual que 100.

Resolver el problema implica asignar los técnicos a los pedidos y esta asignación se puede modelizar mediante una matriz $X^{n \times n}$ donde $x_{ij} = 1$ significa que el técnico i atiende el pedido j, y $x_{ij} = 0$ en caso contrario. Una asignación válida es aquella en la que a cada técnico solo le corresponde un pedido y cada pedido está asignado a un único técnico. Dada una asignación válida, definimos el coste de dicha asignación como:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \times b_{ij}$$

Se pide implementar un programa que permita obtener una asignación (asignar valores a la matriz X). Para ello, se utilizará la siguiente estrategia: asigne a cada técnico el pedido más económico entre los que están disponibles. Al final, el programa debe mostrar la asignación (la matriz X) y el coste de la misma.

Todas las variables y constantes (si las hubiera) deben estar correctamente definidas. Resuelva todo el ejercicio directamente en la funcion `main`. No hace falta asignar valores a la matriz B. Simplemente escriba ‘‘`cargar valores en matriz B`’’

Ejemplo: si

$$B = \begin{bmatrix} 21 & 12 & 31 \\ 16 & 14 & 25 \\ 12 & 18 & 20 \end{bmatrix} \quad \text{entonces (empezando por el técnico 1)} \quad X = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

El coste de dicha asignación es $12 + 16 + 20 = 48$.

◁ Ejercicio 2 ▷ Sub-secuencia débil [2.5 puntos]

Suponga que tiene ya implementado el código de una clase `SecuenciaCaracteres` que representa un conjunto de caracteres consecutivos.

SecuenciaCaracteres
Datos miembros privados:

`static const int TAMANIO = 100`
`char vector_privado[TAMANIO]`
`int total_utilizados`

Métodos públicos que NO hay que implementar y se pueden usar:

`SecuenciaCaracteres()`
`int TotalUtilizados()`
`int Capacidad()`
`void Aniade(char nuevo)`
`char Elemento(int indice)`

Sobre la clase `SecuenciaCaracteres`, se pide construir un método que compruebe si una secuencia `p` contiene *débilmente* a otra secuencia `q`. Todos los caracteres de `q` tienen que aparecer en `p` en el *mismo orden*, aunque *no tienen por qué estar consecutivos*. Por ejemplo, la secuencia `p = {'d','e','s','t','i','n','o'}` contiene débilmente a la secuencia `q = {'s','i'}` pero no a `q = {'i','s'}`.

Implemente todos los métodos auxiliares que estime oportuno.

No hace falta que construya el programa principal, pero incluya al menos la línea en la que se realizaría la llamada al método pedido.

◁ Ejercicio 3 ▷ ASCII Art [3 puntos]

Se dispone de las clases `SecuenciaEnteros`, `SecuenciaCaracteres`, `TablaRectangularEnteros` y `TablaRectangularCaracteres` descritas como se indica a continuación:

SecuenciaTIPO	
- const int	TAMANIO
- TIPO	vector_privado[TAMANIO]
- int	total_utilizados
+	SecuenciaTIPO()
+ int	TotalUtilizados()
+ TIPO	Elemento(int indice)
+ void	Aniade(TIPO nuevo)
+ void	EliminaTodos()

TablaRectangularTIPO	
- const int	NUM_FILS
- const int	NUM_COLS
- TIPO	matriz_privada[NUM_FILS][NUM_COLS]
- int	filas_utilizadas
- int	cols_utilizadas
+	TablaRectangularTIPO()
+	TablaRectangularTIPO(int num_columnas)
+ int	FilasUtilizadas()
+ int	ColumnasUtilizadas()
+ TIPO	Elemento(int indice_fila, int indice_columna)
+ SecuenciaTIPO	Fila(int indice_fila)
+ void	Aniade(SecuenciaTIPO fila_nueva)
+ void	EliminaTodos()

Suponemos que los valores que puede contener un objeto de la clase `TablaRectangularEnteros` están en el conjunto $[0, 255]$. Añadir a la clase `TablaRectangularEnteros` el método `ToASCII` para devolver un objeto de la clase `TablaRectangularCaracteres` de las mismas dimensiones que la tabla sobre la que se aplica, convirtiendo cada número a un carácter de acuerdo a las reglas:

[0, 70)	[70, 130)	[130,170)	[170, 210)	[210, 255]
'@'	'&'	':'	'*'	' '

No puede añadir otros métodos públicos a ninguna clase (aparte del método `ToASCII`)

◁ Ejercicio 4 ▷ Clase `Intervalo` [2 puntos]

Un intervalo es un espacio métrico comprendido entre dos valores o cotas, `a` y `b`, siendo `a` la cota inferior y `b` la cota superior. Cada extremo de un intervalo puede ser abierto o cerrado. Se quiere implementar la clase `Intervalo`.

1. Defina los datos miembro de la clase y los constructores que estime oportunos. Debe considerar el intervalo vacío como un intervalo válido y éste debe estar asociado al constructor sin parámetros.

   En este problema, no se consideran intervalos con extremos infinitos como por ejemplo $(-\infty, \infty)$.

   Considere que una vez construido el intervalo, no se permite su modificación.

2. Implemente los métodos que estime oportuno para conocer el estado completo del intervalo.

3. Implemente un método para comprobar si un intervalo es vacío.

4. Implemente un método que determine si un número pertenece al intervalo.

5. Defina únicamente la cabecera de un método que compruebe si un intervalo está dentro de otro. No hace falta implementar este método.

Escriba una función `main` que use la clase: debe crear un intervalo cualquiera y a continuación pedir una serie de números positivos (el terminador de entrada de datos es el -1) indicando por cada uno de ellos, si pertenece o no al intervalo dado.