



EJERCICIOS PARA PREPARAR EL EXAMEN

¿Queda poco para tu próximo examen de Fundamentos de Programación? ¡No te preocupes! Aquí te traemos una serie de ejercicios explicados paso a paso para convertirte en un as de la algoritmia antes del primer parcial. Estos ejercicios están sacados de una serie de ejercicios propuestos y corregidos por Arturo.

¡Vamos a por ellos!

EJERCICIO 1

En una planta de generación de biocombustibles se está produciendo propanol. El propanol producido se almacena en 200 depósitos dispuestos en forma de matriz de 50 filas y 4 columnas, de manera que el depósito que está en la fila “ i ” columna “ j ” contiene una cantidad de propanol que se denota por $C[i,j]$. Se pide realizar un PSEUDO-CÓDIGO que, a partir de una matriz C de 50 filas y 4 columnas, cuyos elementos representan la cantidad de propanol almacenada en cada depósito, permita localizar la mínima cantidad de propanol y en qué fila y columna se encuentra. La cantidad mínima de propanol se almacenará en C_{min} y la fila y columna en que se encuentra en las variables i_{min} , j_{min} , respectivamente.

RESOLUCIÓN

Podemos ver que en este ejercicio se nos entremezclan distintos conceptos: bucles secuenciales, matrices y bucles condicionales de tipo IF. Parece mucho, pero vamos a ver lo fácil que son en realidad.

En primer lugar, podemos ver que en este caso nos dan las dimensiones de la matriz, se trata de una matriz de 50 filas y 4 columnas. Por lo tanto, a la hora de poner en marcha los bucles secuenciales sabemos que la variable que representa a las filas va a variar desde 1 hasta 50 ($i=1,50$), y que la que representa a las columnas va a ir desde 1 hasta 4 ($j=1,4$). ¡Ya tenemos nuestros bucles secuenciales!

Ahora hay que pensar como vamos a conseguir el valor mínimo y su posición en la matriz. Pues como bien hemos aprendido en clase, solo necesitamos hacer uso de los bucles condicionales de tipo IF. Sin embargo, antes de meternos en cualquier bucle debemos dar un valor inicial a las variables C_{min} , i_{min} y j_{min} en las que almacenaremos los valores que se nos están pidiendo. En este caso, como nos están pidiendo valores mínimos le diremos a nuestro algoritmo que el valor inicial de C_{min} es $C[1,1]$ y que, por lo tanto, $i_{min}=1$ y $j_{min}=1$. Parece que ya lo tenemos todo para resolver el ejercicio...¡Pues al lío!

Pseudo-Código

Inicio

Leer C

Cmin=C[1,1], imin=1, jmin=1

Para i desde 1 hasta 50

 Para j desde 1 hasta 4

 Si C[i,j]<Cmin

 Cmin=C[i,j], imin=i, jmin=j

 Fin de la condición

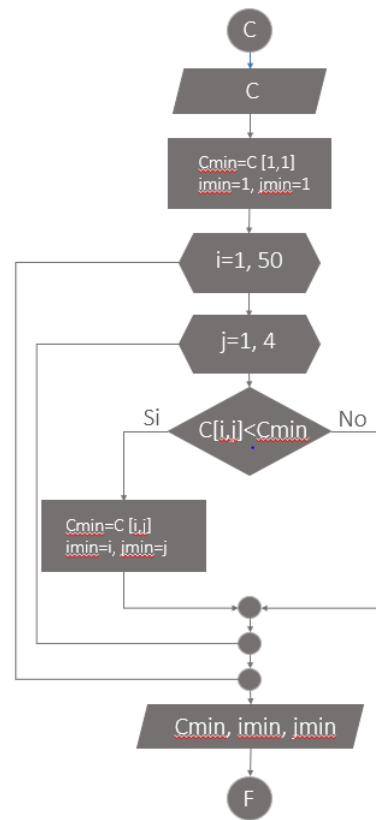
 Fin del bucle en j

Fin del bucle en i

Escribir Cmin, imin, jmin

Fin

Organigrama



EJERCICIO 2

Se está realizando un cultivo de bacterias de tipo E.coli con el fin de utilizarlas en un proceso de fabricación de biocombustibles. Se ha medido la cantidad de bacterias que hay en **NT** instantes de tiempo $(t_1, t_2, \dots, t_{NT})$ conocidos, obteniendo las cantidades: B_1, B_2, \dots, B_{NT} respectivamente, también conocidas. Se desea estimar la producción de bacterias en los instantes de tiempo: $\gamma_1, \gamma_2, \dots, \gamma_M$ conocidos y almacenar los valores obtenidos en un vector **BEST**, de **M** componentes. Para ello, se desea escribir UN PSEUDO-CÓDIGO que realice las siguientes operaciones:

- a) Obtenga una matriz, denominada ECOLI, cuyos elementos vengan dados por:

$$ECOLI_{i,1} = B_i \quad (i = 1, 2, \dots, NT)$$
$$ECOLI_{i,j} = \frac{ECOLI_{i+1,j-1} - ECOLI_{i,j-1}}{t_{i+1} - t_i} \quad (j = 2, \dots, NT; i = 1, 2, \dots, NT - j + 1)$$

Los elementos de la matriz no calculados mediante las fórmulas anteriores tendrán valor 0, por lo que se recomienda inicializar **ECOLI=0**. Téngase en cuenta que la matriz tendrá **NT** filas y **NT** columnas.

b) Obtenga las componentes de un vector BEST de acuerdo con la fórmula:

$$BEST_k = ECOLI_{t_1} + \sum_{i=2}^{NT} \left(ECOLI_{t_i} \cdot \prod_{j=1}^{i-1} (y_k - t_j) \right), (k = 1, 2, \dots, M)$$

El resultado final del algoritmo será BEST

RESOLUCIÓN

Es normal que no sepamos ni por dónde empezar con tantos datos y esas ecuaciones que parecen sacadas de otro planeta. ¡Menos mal que es muchísimo más fácil de lo que parece! Vamos a ir diseccionando cada ejercicio paso por paso para demostrar lo sencillo que es.

Apartado a: Con este apartado básicamente están evaluando como de bien nos manejamos con los bucles secuenciales. Por ello nos dan una ecuación muy grande para trabajar.

Lo primero de todo es identificar cuales son las variables que se deben introducir al algoritmo. En este caso esas variables son la matriz **B**, el vector **t** y el valor **NT**.

Como bien nos dicen en el enunciado lo siguiente que debemos hacer es inicializar la matriz **ECOLI** a cero, un paso esencial para no cometer ningún posible error.

Una vez hemos hecho esto podemos pasar a escribir las ecuaciones:

$$ECOLI_{t_1} = B_i, (i = 1, 2, \dots, NT)$$

Vamos a traducir esta primera ecuación. Simplemente nos están diciendo que para i tomando valores desde 1 hasta NT, $ECOLI[i,1]=B[i]$. Muy fácil ¿no? Pues vamos a por la segunda fórmula.

$$ECOLI_{t_j} = \frac{ECOLI_{t_{j-1}} - ECOLI_{t_{j-1}}}{t_{i,j-1} - t_i}, (j = 2, \dots, NT; i = 1, 2, \dots, NT - j + 1)$$

Esta ecuación puede parecer mucho más complicada, pero únicamente tenemos que crear una estructura de bucles anidados y, en el interior introducir la ecuación que nos piden. Los bucles secuenciales que nos piden son concretamente: Para j tomando valores desde 2 hasta NT y para i tomando valores desde 1 hasta NT-j+1.

Vamos a ver como se vería en pseudo-código y organigrama:

Pseudo-Código

Inicio

Leer B, t, NT

ECOLI=0

Para i desde 1 hasta NT

$ECOLI[i,1]=B[i]$

Fin del bucle en i

Para j desde 2 hasta NT

 Para i desde 1 hasta NT-j+1

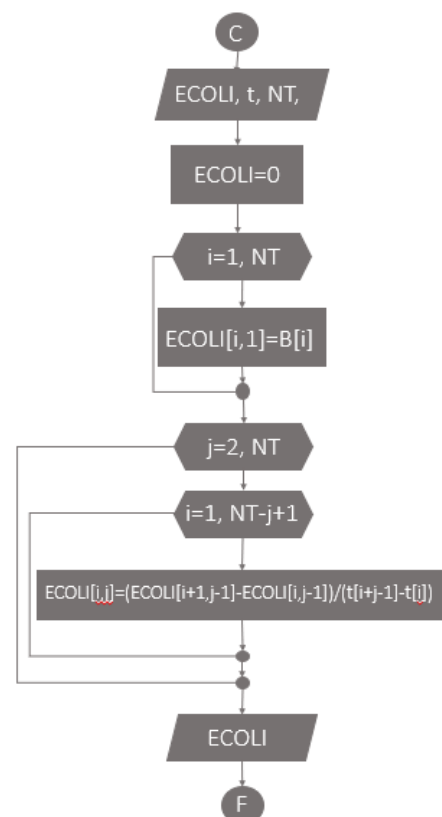
$ECOLI[i,j]=(ECOLI[i+1,j-1]-ECOLI[i,j-1])/(t[i+1]-t[i])$

 Fin del bucle en i

Fin del bucle en j

Escribir ECOLI

Fin



Apartado b: En este apartado tenemos un poco más de lo mismo. Sin embargo, la cosa se complica un poco más porque ahora nos piden que desarrollemos una ecuación en la que se incluyen los conceptos de sumatorio y productorio, pero no pasa nada, esto ya lo hemos visto muy bien en clase. En este caso vamos a observar paso a paso lo que debemos hacer.

Siguiendo los pasos del apartado anterior, lo primero es identificar las variables que se deben introducir al algoritmo, en este caso son la matriz **ECOLI** que obtuvimos en el apartado anterior, los vectores **t** e **y**, y los valores **NT** y **M**.

Ahora bien, antes de empezar a desarrollar la ecuación abrimos un bucle secuencial en el que k toma valores desde 1 hasta M.

Lo siguiente es que dentro del bucle en k abramos un nuevo bucle para i tomando valores desde 2 hasta NT. ***IMPORTANTE**, antes de entrar en este bucle debemos inicializar una variable **SUMA** a cero, ya que dentro del bucle en i vamos a llevar a cabo el sumatorio de la ecuación.

Una vez dentro del bucle en i, y antes de llevar a cabo el sumatorio, debemos resolver el productorio que se nos presenta. Para ello, inicializamos una variable **PROD** a 1 y, después nos metemos en otro bucle secuencial para j tomando valores desde 1 hasta i-1. Ahora si que si podemos empezar a resolver la ecuación.

Dentro del bucle en j calculamos el productorio: **PROD=PROD*(y[k]-t[j])**.

Una vez calculado, podemos salirnos de ese bucle y dentro del bucle en i calcular el sumatorio: **SUMA=SUMA+ECOLI[1,i]*PROD**.

Ahora nos salimos del bucle en i. En ese momento nos encontramos únicamente dentro del bucle en k. En ese momento calculamos la ecuación total: **BEST[k]=ECOLI[1,1]+SUMA**

Vamos a ver como se vería en pseudo-código y organigrama:

Pseudo-Código

Inicio

Leer ECOLI, t, NT, M

Para k desde 1 hasta M

 SUMA=0

 Para i desde 1 hasta NT

 PROD=1

 Para j desde 1 hasta i-1

 PROD=PROD*(y[k]-t[j])

 Fin del bucle en j

 SUMA=SUMA+ECOLI[1,i]*PROD

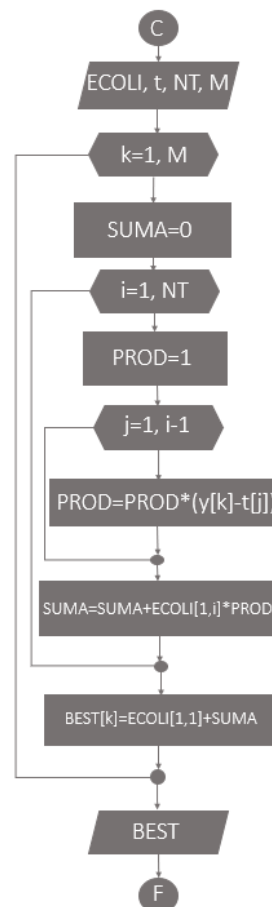
 Fin del bucle en j

 BEST[k]=ECOLI[1,1]+SUMA

Fin del bucle en k

Escribir BEST

Fin



NOTA: También es posible resolver ambos apartados en un solo algoritmo.

EJERCICIO 3

Se considera una matriz W de n filas y n columnas. Se pide realizar un algoritmo que realice las siguientes operaciones:

1. Construir un vector a cuyos elementos sean los que se encuentran en la columna $(n - 2)$ de la matriz W si $n > 2$ o los que se encuentran en la columna n en caso contrario.
2. Construir un vector b cuyos elementos sean los que se encuentran en la tercera fila de la matriz W si $n > 3$ o los que se encuentran en la columna primera en caso contrario.
3. Construir una matriz Z de n filas y dos columnas, siendo el vector a la primera columna y el vector b la segunda. Si hay posiciones vacías en la matriz Z se completarán con ceros.
4. Multiplicar entre sí todos los elementos de la primera columna de Z y multiplicar entre sí todos los elementos de la segunda columna de Z y almacenarlos en un vector V de dos componentes.

RESOLUCIÓN

Como vemos, este ejercicio es más sencillo que los anteriores, pero ¡¡ojo!! En los ejercicios en los que más confiados vamos son en los que solemos fallar más fácilmente. Así que analicemos lo que nos piden con detenimiento. Esta claro que en este ejercicio vamos a trabajar con matrices, vectores y bucles condicionales.

En el **primer punto** nos piden un vector que llamaremos a cuyas componentes son los valores de $W[i, n-2]$ si el valor de n es mayor que 2 y son $W[i, n]$ en caso contrario. Nos están pidiendo que hagamos un bucle IF en toda regla. Vamos a hacer su pseudo-código para visualizar la idea.

Inicio

 Escribir W, n

 Para i desde 1 hasta n

 Si $n > 2$

$a[i] = W[i, n-2]$

 Sino

$a[i] = W[i, n]$

 Fin de la condición

 Fin del bucle en i

 Escribir a

Fin

Ahora vamos a por el **segundo punto**. Si nos damos cuenta nos están pidiendo lo mismo, pero con condiciones distintas. En este caso, si el valor de n es mayor que 3 , el vector b toma los valores de $W[3, i]$, y si estamos ante el caso contrario, b toma los valores de $W[i, 1]$. Veamos como se haría en pseudo-código.

Inicio

Escribir W, n

Para i desde 1 hasta n

Si n>3

b[i]= W[3,i]

Sino

b[i]= W[i,1]

Fin de la condición

Fin del bucle en i

Escribir b

Fin

Teniendo los vectores **a** y **b** ya podemos pasar al **apartado tres**. En este punto nos piden construir una matriz **Z** cuya primera sean las componentes del vector **a** y su segunda columna las del vector **b**. Podríamos plasmar estas condiciones en forma de ecuación:

$Z[i,1]=a[i]$ (i=1, n)

$Z[i,2]=b[i]$ (i=1, n)

Antes de hacer el pseudo-código que no se nos olvide lo que nos dice el enunciado: Inicializar la matriz a 0.

Inicio

Escribir a, b, n

Z=0

Para i desde 1 hasta n

Z[i,1]=a[i]

Z[i,2]=b[i]

Fin del bucle en i

Escribir Z

Fin

Finalmente, abordemos el **apartado cuatro**. En este caso, nos piden que hagamos un productorio con los valores de la columna 1 de la matriz **Z** almacenándolos en la posición 1 de un vector **V**. Por otro lado, también debemos hacer otro productorio con los valores de columna 2 e incluirlos en la posición 2 del mismo vector. Podemos por lo tanto representarlo con la fórmula del productorio:

$$v[1] \prod_{i=1}^n Z[i, 1]$$
$$v[2] \prod_{i=1}^n Z[i, 2]$$

*Como se trata de productorios, es muy importante que inicialicemos V[1] y V[2] a 1.

En pseudo-código sería algo así:

Inicio

Escribir Z, n

$V[1]=1, V[2]=1$

Para i desde 1 hasta n

$V[1]=V[1]*Z[i,1]$

$V[2]=V[2]*Z[i,2]$

Fin del bucle en i

Escribir V

Fin

Ahora que sabemos como resolver cada apartado del ejercicio, vamos a realizar un organigrama en el que se exprese un algoritmo que incluya los cuatro apartados. ¡Vamos a por ello!

