EJERCICOS ORGANIGRAMAS (ALGORITMIA)

Una vez que tenemos una base con los conceptos básicos de algoritmos y de como se realiza mediante organigramas, ya es posible llevar a cabo diferentes ejercicios propuestos tanto en clase como en exámenes en modo de algoritmos. Estos pueden contener bucles for, sumatorios y productorios, matrices, vectores, condiciones como while y o if...

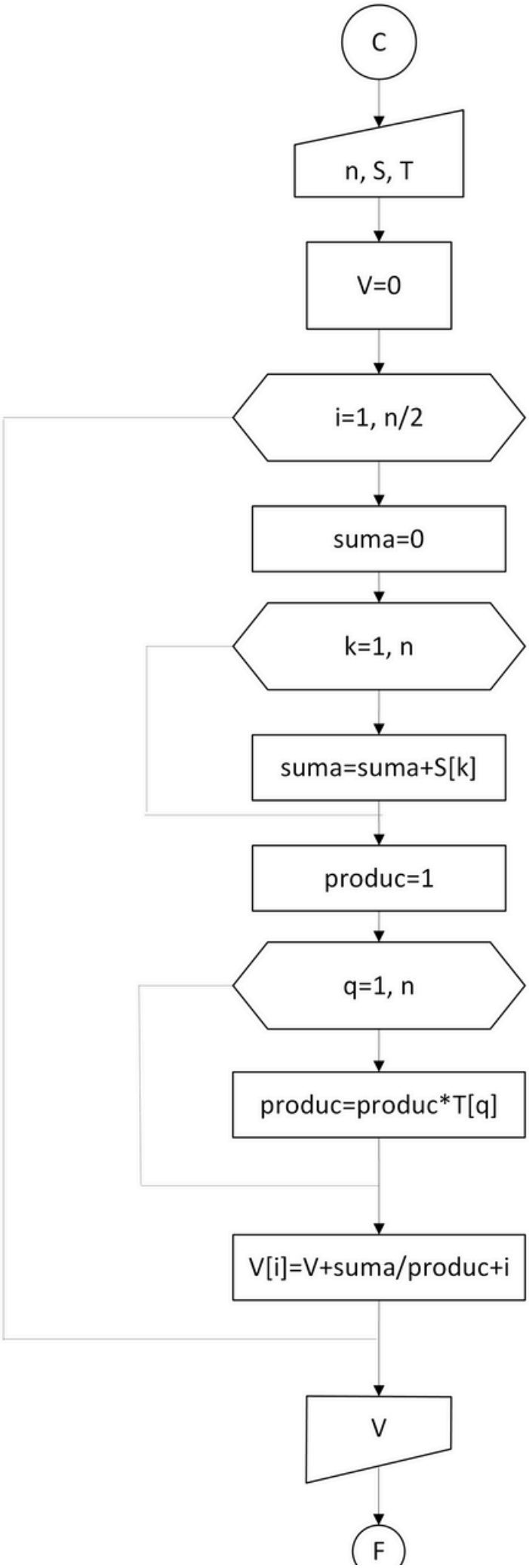
El fin de este recurso es que el alumno tenga un o varios ejemplos de los tipos de bloques que hay en algoritmia que sean entendibles y que posteriormente puedan ser usados por el alumnado para hacer diferentes ejercicios que se nos propondrán durante el semestre.

EJERCICIO CON SUMATORIOS Y PRODUCTORIOS

Ejercicio 4) Realizar un vector V[i] con elementos comprendidos entre 1 y p, siendo p la mitad de la longitud "n" de S o T

$$V[i] = \sum_{k=1}^{n} S[k] / \prod_{q=1}^{n} T[q] + i$$

Se nos pide calcular un vector V que se form siguiendo la ecuación. Primero detectamos los inputs, que serían n, S y T. Dividiremos la ecuación en dos: el sumatorio y el productorio, los que al final dividiremos.


C
n, S, T
V=0
i=1, n/2
suma=0
k=1, n
suma=suma+S[k]
produc=1
q=1, n
produc=produc*T[q]
V[i]=V+suma/produc+i
V
F

Una vez puestos los inputs hacemos que nuestro vector sea igual a 0 para ir metiendo nuestros cálculos ahí. Después abrimos un bucle for para i porque v depende de i. Este bucle lo cerraremos al haber hecho todos los cálculos. i va de 1 a n porque en el enunciado así lo pide. En el siguiente bloque comenzamos con los cálculos haciendo el sumatorio a 0. Luego, abrimos el bucle for con k como contador, anotamos todo lo que está dentro del sumatorio y cerramos el bucle. Lo siguiente es el productorio: lo igualamos a 1 y abrimos el bucle con q como contador, que va de 1 a n como k. Dentro del bucle anotamos todo lo que hay dentro del productorio. Cerramos el bucle for del productorio y procedemos a formar la ecuación completa del vector, incluyendo el sumatorio, el productorio y la suma con i. Finalmente tenemos como output el vector v.

EJERCICIO CON BUCLES FOR, IF Y MATRICES

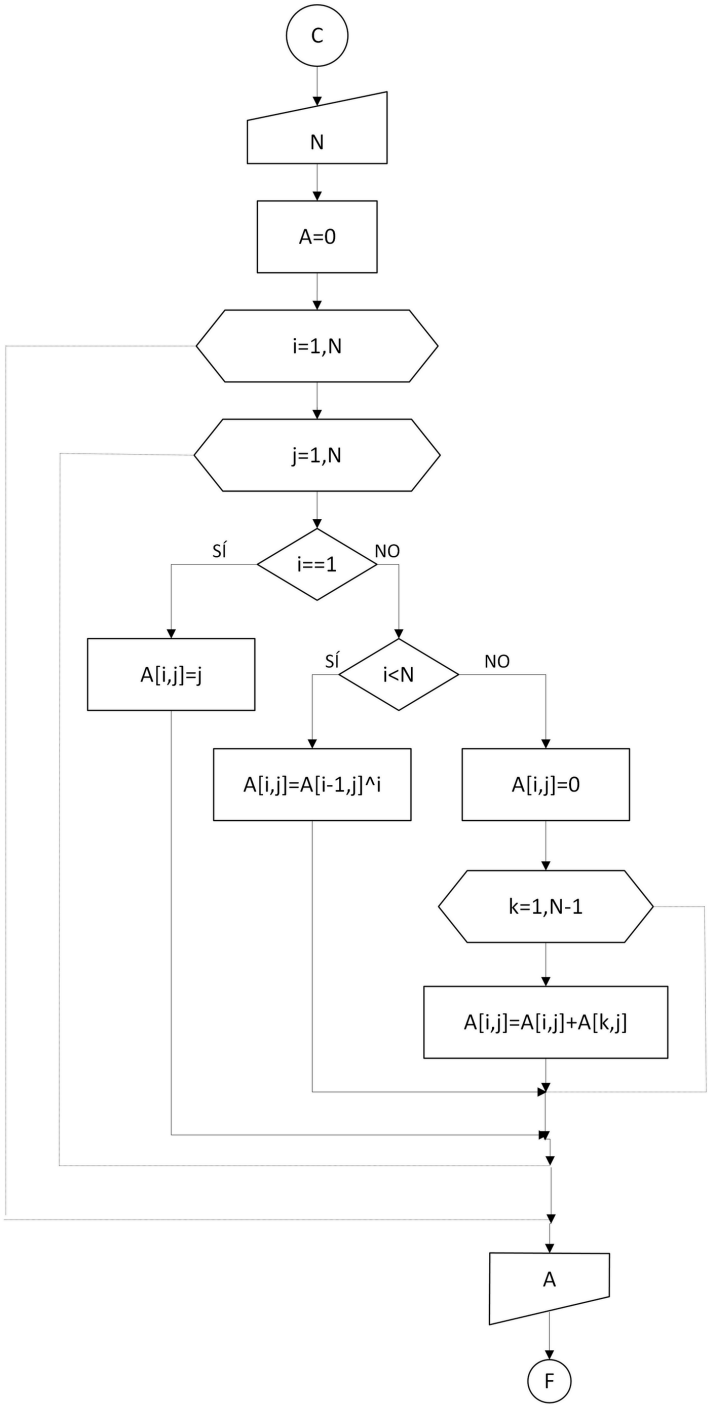
Realizar un algoritmo (organigrama o pseudo-código) que permita obtener una matriz de dimensiones (N,N) tal que:

- Su primera fila esté formada por los N primeros números naturales, comenzando por el número 1.
- La fila i (i=2,3,...,N-1) se obtenga elevando a i los elementos de la fila i-1.
- Cada elemento de la fila N se obtenga mediante la suma de los elementos de su misma columna y filas anteriores (ver ejemplo ilustrativo).

Ejemplo ilustrativo: (N=4)

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1^2 & 2^2 & 3^2 & 4^2 \\ 1^3 & 4^3 & 9^3 & 16^3 \\ 1+1^2+1^3 & 2+2^2+4^3 & 3+3^2+9^3 & 4+4^2+16^3 \end{pmatrix}$$

En el ejercicio se pide realizar una matriz de N filas y N columnas, en el que tendremos como contador la letra i (que irá de 1 a N) que será para las filas y otro contador j (que irá también de 1 a N) que será para columnas. Nombramos como A la matriz que queremos calcular que será: A[i,j]. Pondremos diferentes condiciones mediante bloques if, para así cumplir que la fila i sea los elementos de la fila anterior elevados a i y que la fila final N será la suma de todos los elementos de la misma columna pero diferentes columnas. Una vez entendidas las condiciones y qué es el producto final procedemos a crear el organigrama:



Vemos que al principio igualamos nuestra matriz A a 0 para que después se vaya rellenando.

Después, abrimos dos bucles for: i para las filas y j para las columnas, que irán de 1 a N.

Con el bloque if ponemos la primera condición: para nuestra fila 1, el valor la columna (j) será igual a A[i,j]. Cuando no es la fila 1, debemos seguir haciendo cálculos.

El segundo bloque condicional obliga a que nuestra fila sea inferior que N, porque nuestro producto final es la fila N. Mientras la fila sea menor a N, nuestra matriz irá cogiendo el valor de la fila anterior a la que estamos (i-1,j) y elevándolo a sí mismo, obteniendo los valores que queremos en cada fila. Este proceso se irá repitiendo hasta que lleguemos a la fila N, porque ahí le imponemos otra condición.

Cuando i=N (cuando la fila es el número que se pide), haremos el sumatorio exigido. Para ello, primero igualamos nuestra matriz a 0 (como en todos los sumatorios) y después abrimos un nuevo bucle for con k como contador, que irá de 1 a N-1 (porque necesitamos todos los valores de la matriz menos la fila N). El sumatorio será nuestra matriz A[i,j]+A[k,j],que sumará todos los términos de la matriz en nuestra fila N de la matriz A[i,j].

Una vez hechos todos los cálculos tenemos la matriz A como output y se finaliza el organigrama.

EJERCICIO COMPLEJO CON DE TODO

EJERCICIO 1 (4 puntos) (Se recomienda comenzar leyendo la nota al final del enunciado)

Consideramos el vector N con m componentes y el vector R que contiene m valores en el conjunto {0,1}, que se han obtenido como el resto de dividir las componentes de los elementos del vector N entre 2. Tomando como datos de entrada: el vector N, el vector R, una variable m (dimensión de ambos vectores), se desea realizar un algoritmo que realice los siguientes pasos:

• Construir dos vectores llamados: Par, Impar, que contengan respectivamente los elementos pares e impares del vector N.

• Obtener los valores P, Q que sean, respectivamente, los valores máximos almacenados en los vectores Par e Impar.

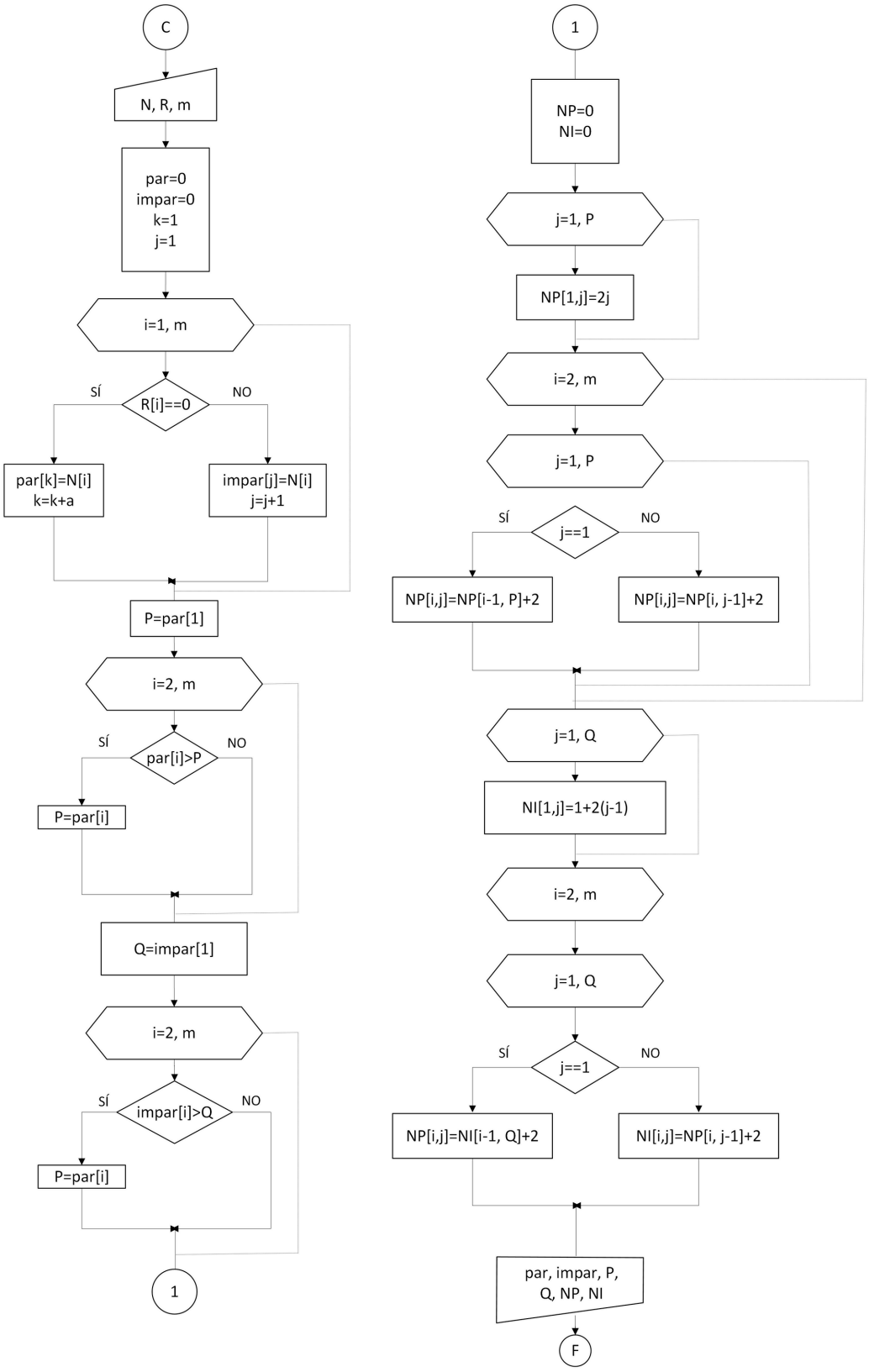
• Construir dos matrices, llamadas NP (de m filas y P columnas) y NI (de m filas y Q columnas), que contengan, respectivamente, números naturales pares consecutivos comenzando en el número 2 y números naturales impares consecutivos comenzando en el número 1.

NOTA: Al dividir un número entre 2 el resto es 0 si es par, y 1 si es impar.

Ejemplo:

$$\mathbf{N} = (8 \quad 5 \quad 6 \quad 7); \text{Par} = (8 \quad 6); \text{Impar} = (5 \quad 7);\ P = 8;\ Q = 7$$

$$\mathbf{NP} = \begin{pmatrix} 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 \\ 18 & 20 & 22 & 24 & 26 & 28 & 30 & 32 \\ 34 & 36 & 38 & 40 & 42 & 44 & 46 & 48 \\ 50 & 52 & 54 & 56 & 58 & 60 & 62 & 64 \end{pmatrix}; \quad \mathbf{NI} = \begin{pmatrix} 1 & 3 & 5 & 7 & 9 & 11 & 13 \\ 15 & 17 & 19 & 21 & 23 & 25 & 27 \\ 29 & 31 & 33 & 35 & 37 & 39 & 41 \\ 43 & 45 & 47 & 49 & 51 & 53 & 55 \end{pmatrix}$$



Podemos ver que este ejercicio es bastante complejo comparado con los anteriores, por eso lo explicaremos paso por paso.

Comenzamos pidiendo los inputs: N, R y m. Después daremos valor 0 a todas las variables que usaremos en el algoritmo. Abrimos un bucle for con i como contador que irá de 1 hasta m, ya que recorrerá todos los elementos del vector R (que depende de i), donde se almacenarán los restos de la división de los elementos de N entre 2.

Imponemos la primera condición mediante un bloque if: si nuestro vector R es igual a 0 significa que N es un número par, y tendremos que recoger ese número en el vector de números pares que se llama par. Al contrario, si R no es 0, significa que N es impar y tendremos que recoger el número en el vector de números impares. La condición que ponemos debajo es para que no queden ceros en los vectores que estamos creando, haciendo uso de k y j. A estos les hemos dado valor 1 al comienzo, para que el siguiente número N se vaya almacenando en la siguiente posición del vector.

Una vez tengamos el primer valor del vector y este se haya guardado en par[1] e impar[1], abrimos otro bucle pero esta vez empezando en 2 hasta m. Los primeros valor de los vectores par[i] e impar[j] son par[1] e impar[1]. Estos los nombraremos como P y Q para comparar cada valor de par[i] e impar[i] con estos. Si son mayores, se guardará en P el valor de par[i] y en Q el valor de impar[i].

En la segunda parte creamos dos variables nuevas: NP (matriz de números pares) y NI (matriz de números impares). Primero hacemos la primera fila de la matriz NP: los elementos se obtienen multiplicando por 2 los valores de su misma columna. Después, hacemos el cálculo para el resto de filas, que impondremos la siguiente condición: si es el primer elemento de esta, la calculamos, y si no lo es, lo obtenemos como el valor de la misma fila pero anterior columna más 2. Para la matriz de números impares hacemos lo mismo: primero la primera fila como 1 más 2 veces la columna-1 y el resto de filas como NP (el primer elemento de la fila sumando 2 al valor de la fila anterior, última columna y el resto como 2+el elemento de la misma fila y columna anterior).

Así obtenemos lo que queríamos: los vectores par e impar, los valores de P y Q y las matrices NP y NI, que serán el output.