

ARREGLOS

1. Escriba el código de una función que cumpla con el siguiente prototipo:

```
void rotaDerecha(int []);
```

Esta función recibe un arreglo de 10 enteros. La función mueve a cada elemento una posición adelante, colocando el último elemento del arreglo en la primera localidad del arreglo resultante. Por ejemplo, si el arreglo A tuviera los valores $A=[0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9]$, después de ejecutar la función `rotaDerecha(A)` el arreglo se modifica de la siguiente manera: $A=[9\ 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8]$. El programa debe poder ejecutarse n veces sobre el arreglo resultante, es decir, en la siguiente ejecución de `rotaDerecha(A)` el arreglo quedaría $A=[8\ 9\ 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7]$.

2. Escribir un programa que contenga una función con el siguiente prototipo:

```
int mayor(int A[], int inf, int sup);
```

La función regresa el **mayor valor** del arreglo dentro del rango definido por los parámetros de la función (inf y sup). Por ejemplo, si tenemos el siguiente arreglo: $A=[10\ 12\ 2\ 0\ 4\ 35\ 24]$, y la llamada a la función `mayor(A, 0, 3)`, dicha función regresaría como resultado 12 y `mayor(A, 0, 6)` regresa 35.

3. Crear una función que reemplace una submatriz de la matriz rectangular $A[6][6]$ por otra submatriz correspondiente a la matriz rectangular $B[6][6]$:

```
void dupmat(int rs, int ri, int cd, int ci, int A[][6], int B[][6]);
```

rs, ri: Índice superior e inferior de los renglones, respectivamente;

cd, ci: Índice superior e inferior de las columnas, respectivamente;

A, B: Matrices

A:

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

B:

60	61	62	63	64	65
66	67	68	69	70	71
72	73	74	75	76	77
78	79	80	81	82	83
84	85	86	87	88	89
90	91	92	93	94	95

```
dupmat(1, 4, 1, 3, A, B);
```

A:

1	2	3	4	5	6
7	67	68	69	11	12
13	73	74	75	17	18
19	79	80	81	23	24
25	85	86	87	29	30
31	32	33	34	35	36

elemento 1,1

elemento 4,3