

```

/* *****
 * Solución al examen del día 29 de Enero de 2021
 *
 * Fundamentos de programación
 * ***** */
#include <stdio.h>

/* tamaño de los vectores que utiliza el programa */
#define TAM 10

// ----- PROTOTIPOS -----
void leer      (int vector[], int n_elem);
void mostrar   (int vector[], int n_elem, char mensaje[]);
void max_A_y_B (int *tamA, int *tamB);
int definir_C (int A[], int B[], int C[], int maxA, int maxB);

/* -----
 * Función que muestra el contenido de un vector de enteros
 *
 * PARÁMETROS:
 *   vector: vector a visualizar
 *   n_elem: número de elementos a visualizar
 *   mensaje: texto que se mostrará antes que los elementos del vector
 *----- */
void mostrar (int vector[], int n_elem, char mensaje[])
{
    int i;    // contador de elementos del vector

    printf("\nELEMENTOS -- %s -- ", mensaje);

    /* para cada componente del vector, desde la primera... */
    for(i=0; i<n_elem; i++)
        printf(" %d", vector[i]);    // ...se muestra su valor
}

/* -----
 * Función que lee datos del teclado y los almacena en un vector de enteros
 *
 * PARÁMETROS:
 *   vector: vector que almacenará los valores leídos
 *   n_elem: número de elementos a almacenar
 *----- */
void leer (int vector[], int n_elem)
{
    int i;    // contador de elementos del vector

    // se indica al usuario que debe proporcionar valores enteros
    printf("\nTeclee valores enteros para el vector: ");

    /* para cada componente del vector, desde la primera... */
    for(i=0; i<n_elem; i++)
    {
        // ...se indica en qué posición del vector se almacenará
        printf("\nvalor para la posición %d: ", i);

        // ...se lee y almacena un valor en la posición i-ésima
    }
}

```

```

        scanf("%d", &vector[i]);
    }
}

/* -----
 * Función que da valores al vector C a partir de los vectores A y B.
 *
 * PARÁMETROS:
 *   A, B: vectores que contienen información de entrada
 *   C: vector en el que se van a almacenar valores
 *   tamA: número de elementos del vector A (máximo TAM elementos)
 *   tamB: número de elementos del vector B (máximo TAM elementos)
 *
 * RETORNO:
 *   número de elementos copiados en C (<= TAM)
 * ----- */
int definir_C(int A[], int B[], int C[], int maxA, int maxB)
{
    int i, j;    // contador de elementos de los vectores

    /* si el vector A almacena más elementos que el B, se toma el vector
       A para dar valores a C */
    if(maxA > maxB)
    {
        /* se copia en C la resta de las componentes de A y B, considerando
           sólo las maxB primeras posiciones de ambos vectores */
        for(i=0; i< maxB; i++)
            C[i]= A[i] - B[i];

        return maxB; // el vector C contiene tantos elementos como el B
    }

    /* el caso contrario, se toma el vector B para dar valores a C */
    else
    {
        /* posición de C en la que se almacenará el primer dato: la
           última del vector */
        i= TAM-1;

        /* se recorre el vector B desde el último elemento hacia el primero,
           para copiar el contenido del vector B sobre el C*/
        for(j=maxB-1; j>=0; j--)
        {
            /* se copia el elemento j-ésimo de B en la posición i-ésima de C */
            C[i] = B[j];

            /* se pasa a la siguiente posición de C a la que se dará valor */
            i--;
        }

        /* se rellena el resto de componentes de C (hasta llegar a la primera
           * posición de este vector) almacenando un valor igual al doble
           * del índice que corresponde a la posición */
        for(; i>=0; i--)
            C[i] = 2*i;

        return TAM; // el vector C contiene TAM elementos
    }
}

```

```

/* -----
 * Función que solicita y lee el número de elementos que se
 * van a utilizar en los vectores A y B.
 * Según el enunciado del examen, se debe utilizar una misma
 * función para leer ambos valores
 *
 * PARÁMETROS:
 *   tamA: número de elementos del vector A (máximo TAM elementos)
 *   tamB: número de elementos del vector B (máximo TAM elementos)
 * ----- */
void max_A_y_B(int *tamA, int *tamB)
{
    /* se solicita y lee el número de elementos a almacenar
    en el vector A, que puede estar entre 0 y TAM */
    do
    {
        printf("\n¿Elementos de A? (entre 0 y %d) ", TAM);
        scanf("%d", tamA);
    }while(*tamA <0 || *tamA >TAM);

    /* se repiten las mismas operaciones, pero ahora para
    el vector B */
    do
    {
        printf("\n¿Elementos de B? (entre 0 y %d) ", TAM);
        scanf("%d", tamB);
    }while(*tamB <0 || *tamB >TAM);
}

void main()
{
    int A[TAM], B[TAM], // vectores de datos que proporcionará el usuario
        C[TAM];         // vector cuyos elementos calculará el programa
    int maxA, // número de elementos del vector A que se utilizarán en el programa
        maxB, // número de elementos del vector B que se utilizarán en el programa
        maxC; // número de elementos del vector B que se utilizarán en el programa

    /* 1- se determina el número de elementos que se utilizarán en
    los vectores A y B */
    max_A_y_B(&maxA, &maxB);

    /* 2- se leen valores proporcionados por el usuario y se almacenan en
    ambos vectores */
    leer(A, maxA);
    leer(B, maxB);

    // 3- se define el contenido del vector C
    maxC = definir_C(A, B, C, maxA, maxB);

    // se muestran los tres vectores
    mostrar(A, maxA, "vector A");
    mostrar(B, maxB, "vector B");
    mostrar(C, maxC, "vector C");
}

```