

Concurso Auxiliar de 1era categoría

Informática

Damian Ariel Marotte

- 1 Interpretar el enunciado.

- 1 Interpretar el enunciado.
- 2 Datos y resultados.

- 1 Interpretar el enunciado.
- 2 Datos y resultados.
- 3 Metodología.

- 1 Interpretar el enunciado.
- 2 Datos y resultados.
- 3 Metodología.
- 4 Algoritmo.

- 1 Interpretar el enunciado.
- 2 Datos y resultados.
- 3 Metodología.
- 4 Algoritmo.
- 5 Prueba de escritorio.

- 1 Interpretar el enunciado.
- 2 Datos y resultados.
- 3 Metodología.
- 4 Algoritmo.
- 5 Prueba de escritorio.
- 6 Programa en C.

Práctica 7

Ejercicio 1

Dado un arreglo de 30 filas y 12 columnas de números enteros que se ingresan por teclado, calcular:

Práctica 7

Ejercicio 1

Dado un arreglo de 30 filas y 12 columnas de números enteros que se ingresan por teclado, calcular:

- La posición del máximo de cada fila, mediante la función MAXF.

Práctica 7

Ejercicio 1

Dado un arreglo de 30 filas y 12 columnas de números enteros que se ingresan por teclado, calcular:

- La posición del máximo de cada fila, mediante la función MAXF.
- Realizar lo mismo pero a través de una subrutina MAX_FILA.

Práctica 7

Ejercicio 1

Dado un arreglo de 30 filas y 12 columnas de números enteros que se ingresan por teclado, calcular:

- La posición del máximo de cada fila, mediante la función MAXF.
- Realizar lo mismo pero a través de una subrutina MAX_FILA.

Subalgoritmo función

Un subalgoritmo función es un subalgoritmo que recibiendo o no datos devuelve un único resultado.

Práctica 7

Ejercicio 1

Dado un arreglo de 30 filas y 12 columnas de números enteros que se ingresan por teclado, calcular:

- La posición del máximo de cada fila, mediante la función MAXF.
- Realizar lo mismo pero a través de una subrutina MAX_FILA.

Subalgoritmo función

Un subalgoritmo función es un subalgoritmo que recibiendo o no datos devuelve un único resultado.

Subalgoritmo subrutina

Una subrutina es un subalgoritmo que recibiendo o no datos permite devolver varios resultados, un resultado o ninguno.

- Datos: 360 valores enteros distribuidos en un arreglo de 30 filas y 12 columnas.

- Datos: 360 valores enteros distribuidos en un arreglo de 30 filas y 12 columnas.
- Resultados: Las posiciones de los valores maximos en cada una de las filas.

- Carga datos.
 - Para cada fila, el usuario debe ingresar un valor por cada columna.

- Carga datos.
 - Para cada fila, el usuario debe ingresar un valor por cada columna.
- Posición del máximo de una fila.

- Carga datos.
 - Para cada fila, el usuario debe ingresar un valor por cada columna.
- Posición del máximo de una fila.

1	0	2	7	3
8	6	9	8	5
6	7	1	8	2
6	0	3	6	4
1	7	3	7	6

- Carga datos.
 - Para cada fila, el usuario debe ingresar un valor por cada columna.
- Posición del máximo de una fila.

1	0	2	7	3
8	6	9	8	5
6	7	1	8	2
6	0	3	6	4
1	7	3	7	6

- Mostrar resultados.
 - Para cada fila, calculamos la posición del máximo con la función anterior.

```
SubAlgoritmo maxf(Entero arreglo[30][12], Entero fila) : Entero  
Variables
```

```
Inicio
```

```
Fin
```

```
SubAlgoritmo maxf(Entero arreglo[30][12], Entero fila) : Entero
```

```
Variables
```

```
Entero: posicion
```

```
Inicio
```

```
    posicion <- 1
```

```
Fin
```

```
SubAlgoritmo maxf(Entero arreglo[30][12], Entero fila) : Entero
```

```
Variables
```

```
Entero: posicion, columna
```

```
Inicio
```

```
    posicion <- 1
```

```
    Repetir Para columna <- 2 Hasta 12
```

```
        Fin Para
```

```
Fin
```

```
SubAlgoritmo maxf(Entero arreglo[30][12], Entero fila) : Entero
```

```
Variables
```

```
Entero: posicion, columna
```

```
Inicio
```

```
    posicion <- 1
```

```
    Repetir Para columna <- 2 Hasta 12
```

```
        Si arreglo[fila][columna] > arreglo[fila][posicion] Entonces
```

```
            Fin Si
```

```
    Fin Para
```

```
Fin
```

```
SubAlgoritmo maxf(Entero arreglo[30][12], Entero fila) : Entero
```

```
Variables
```

```
Entero: posicion, columna
```

```
Inicio
```

```
    posicion <- 1
```

```
    Repetir Para columna <- 2 Hasta 12
```

```
        Si arreglo[fila][columna] > arreglo[fila][posicion] Entonces
```

```
            posicion <- columna
```

```
        Fin Si
```

```
    Fin Para
```

```
    Devolver(posicion)
```

```
Fin
```

Algoritmo Practica7Ej1
Variables

Inicio

Fin

Programa principal

Algoritmo Practica7Ej1

Variables

Entero: fila

Inicio

 // Ingresar datos

 Repetir Para fila <- 1 Hasta 30 Hacer

 Fin Para

Fin

Algoritmo Practica7Ej1

Variables

Entero: fila, columna

Inicio

 // Ingresar datos

 Repetir Para fila <- 1 Hasta 30 Hacer

 Repetir Para columna <- 1 Hasta 12

 Fin Para

 Fin Para

Fin

Algoritmo Practica7Ej1

Variables

Entero: fila, columna, arreglo[30][12]

Inicio

 // Ingresar datos

 Repetir Para fila <- 1 Hasta 30 Hacer

 Repetir Para columna <- 1 Hasta 12

 Mostrar("Ingrese fila ", fila, " columna ", columna)

 Leer(arreglo[fila][columna])

 Fin Para

 Fin Para

Fin

Algoritmo Practica7Ej1

Variables

Entero: fila, columna, arreglo[30][12]

Inicio

 // Ingresar datos

 Repetir Para fila <- 1 Hasta 30 Hacer

 Repetir Para columna <- 1 Hasta 12

 Mostrar("Ingrese fila ", fila, " columna ", columna)

 Leer(arreglo[fila][columna])

 Fin Para

 Fin Para

 // Mostrar resultados

 Repetir Para fila <- 1 Hasta 30 Hacer

 Fin Para

Fin

Algoritmo Practica7Ej1

Variables

Entero: fila, columna, arreglo[30][12]

Inicio

 // Ingresar datos

 Repetir Para fila <- 1 Hasta 30 Hacer

 Repetir Para columna <- 1 Hasta 12

 Mostrar("Ingrese fila ", fila, " columna ", columna)

 Leer(arreglo[fila][columna])

 Fin Para

 Fin Para

 // Mostrar resultados

 Repetir Para fila <- 1 Hasta 30 Hacer

 columna <- maxf(arreglo, fila)

 Mostrar("El maximo de la fila ", fila, " esta en ", columna)

 Fin Para

Fin

```
SubAlgoritmo max_fila(Entero arreglo[30][12], Entero fila)
```

```
Variables
```

```
Entero: posicion, columna
```

```
Inicio
```

```
    posicion <- 1
```

```
    Repetir Para columna <- 2 Hasta 12
```

```
        Si arreglo[fila][columna] > arreglo[fila][posicion] Entonces
```

```
            posicion <- columna
```

```
        Fin Si
```

```
    Fin Para
```

```
    Mostrar("El maximo de la fila ", fila, " esta en ", posicion)
```

```
Fin
```

Algoritmo Practica7Ej1

Variables

Entero: fila, columna, arreglo[30][12]

Inicio

 // Ingresar datos

 Repetir Para fila <- 1 Hasta 30 Hacer

 Repetir Para columna <- 1 Hasta 12

 Mostrar("Ingrese fila ", fila, " columna ", columna)

 Leer(arreglo[fila][columna])

 Fin Para

 Fin Para

 // Mostrar resultados

 Repetir Para fila <- 1 Hasta 30 Hacer

 max_fila(arreglo, fila)

 Fin Para

Fin

```
#include <stdio.h>

int maxf(int arreglo[30][12], int fila) {
    int posicion = 0;
    for (int columna = 1; columna < 12; columna++)
        if (arreglo[fila][columna] > arreglo[fila][posicion])
            posicion = columna;
    return posicion;
}

int main() {
    int arreglo[30][12];
    for (int fila = 0; fila < 30; fila++) // Ingresar datos
        for (int columna = 0; columna < 12; columna++) {
            printf("Ingrese fila %d columna %d: ", fila + 1, columna + 1);
            scanf("%d", &arreglo[fila][columna]);
        }
    for (int fila = 0; fila < 30; fila++) { // Mostrar resultados
        int columna = maxf(arreglo, fila);
        printf("El maximo de la fila %d esta en %d\n", fila + 1, columna + 1);
    }
}
```