

Laboratorio - 4

Alumno: Jackson Fernando Merma Portocarrero

Escuela Profesional: Ingeniería de Sistemas

CUI: 20202143

Correo: jmermap@unsa.edu.pe

Ejercicio 1: Número máximo

Dada la problemática presentada donde el algoritmo mantenía una complejidad de $O(n^2)$, se implementó uno donde este coste sea de $O(n)$; para esta solución se llama a una función “**foundMax**”, que recibe al arreglo de números y su longitud, además retorna un entero (número máximo):

Nombre de archivo: **max_number.cpp**

```
int foundMax(int a[], int len){  
    int aux=a[0];  
  
    for(int i=1;i<len;i++)  
        aux=max(aux,a[i]);  
    return aux;  
}
```

input:

3
4 9 1

output:

9

En esta solución, se inicializa al máximo en la primera posición, y a partir de la segunda, es donde el bucle reemplaza al mayor(aux), en caso de encontrar uno mayor que el actual.

***Observación:** Este algoritmo es con entrada de datos, donde la primera línea lee la cantidad de números a ingresar (n), y la siguiente línea lee ‘n’ números separados por espacios.

Ejercicio 2: Sumar si es par

En este problema, básicamente se imprime la suma del i-ésimo número par con los 'n' números de la lista, esto quiere decir que si se tiene 4 de 10 números pares en la lista de números, este algoritmo imprimirá 40 números.

Nombre del archivo: **sumar_par.cpp**

```
for(int i=0;i<n;i++){
    if(arr[i]%2==0){
        for(int u=0;u<n;u++){
            cout<<arr[i]<<"+"<<arr[u]<<"="<<(arr[u]+arr[i])<<"\n";
        }
    }
}
```

input:

3

1 2 3

output:

2+1=3

2+2=4

2+3=5

***Observación:** Este algoritmo es con entrada de datos, donde la primera línea lee la cantidad de números a ingresar (n), y la siguiente línea lee 'n' números separados por espacios.

Ejercicio 3: Suman 10

En este ejercicio, se verifica si dos números de un arreglo desordenado, suman 10. Para ello se llama a la función '**sum_ten**', que recibe al arreglo de números y su longitud, retornando un valor booleano.

```
bool sum_ten(int arr[], int n){
    for(int i=0;i<n;i++){
        for(int u=0;u<n;u++){
            if(i!=u&&arr[i]+arr[u]==10)
                return true;
        }
    }
    return false;
}
```

Input:

3

5 2 5

Output:

hay suma de 10

***Observación:** Este algoritmo es con entrada de datos, donde la primera línea lee la cantidad de números a ingresar (n), y la siguiente línea lee 'n' números separados por espacios.