

Soluciones a los Ejercicios de Clase

Universidad Nacional del Altiplano

Ejercicio 1

Declara un arreglo de enteros de tamaño 5 e inicialízalo con los valores [1, 2, 3, 4, 5].

Imprime cada elemento en el arreglo utilizando un bucle.

```
```cpp
#include <iostream>

using namespace std;

int main() {

 int arreglo[5] = {1, 2, 3, 4, 5};

 for (int i = 0; i < 5; i++) {

 cout << arreglo[i] << " ";

 }

 return 0;

}
```
```

Ejercicio 2

Crea un arreglo de enteros de tamaño n (donde n es un valor ingresado por el usuario)

y solicita al usuario que ingrese los elementos del arreglo. Calcula e imprime la suma

de todos los elementos en el arreglo.

```
```cpp
#include <iostream>

using namespace std;
```

```

int main() {

 int n, suma = 0;

 cout << "Ingrese el tamaño del arreglo: ";

 cin >> n;

 int arreglo[n];

 cout << "Ingrese los elementos del arreglo:" << endl;

 for (int i = 0; i < n; i++) {

 cin >> arreglo[i];

 suma += arreglo[i];

 }

 cout << "La suma de los elementos es: " << suma << endl;

 return 0;

}

...

```

### Ejercicio 3

Dado un arreglo de enteros, encuentra e imprime el elemento máximo y el elemento mínimo en el arreglo.

```

```cpp

#include <iostream>

using namespace std;

int main() {

    int n;

    cout << "Ingrese el tamaño del arreglo: ";

    cin >> n;

    int arreglo[n];

```

```

cout << "Ingrese los elementos del arreglo:" << endl;

for (int i = 0; i < n; i++) {

    cin >> arreglo[i];

}

int max = arreglo[0], min = arreglo[0];

for (int i = 1; i < n; i++) {

    if (arreglo[i] > max) max = arreglo[i];

    if (arreglo[i] < min) min = arreglo[i];

}

cout << "Máximo: " << max << ", Mínimo: " << min << endl;

return 0;

}

...

```

Ejercicio 4

Dado un arreglo de enteros y un número x, cuenta cuántas veces aparece x en el arreglo.

```

```cpp

#include <iostream>

using namespace std;

int main() {

 int n, x, count = 0;

 cout << "Ingrese el tamaño del arreglo: ";

 cin >> n;

 int arreglo[n];

 cout << "Ingrese los elementos del arreglo:" << endl;

 for (int i = 0; i < n; i++) {

```

```

 cin >> arreglo[i];

 }

 cout << "Ingrese el número a buscar: ";

 cin >> x;

 for (int i = 0; i < n; i++) {

 if (arreglo[i] == x) count++;

 }

 cout << "El número " << x << " aparece " << count << " veces." << endl;

 return 0;

}

...

```

## Ejercicio 5

Dado un arreglo de enteros, crea un nuevo arreglo con los elementos en orden inverso.

Imprime el arreglo original y el invertido.

```

```cpp

#include <iostream>

using namespace std;

int main() {

    int n;

    cout << "Ingrese el tamaño del arreglo: ";

    cin >> n;

    int arreglo[n];

    cout << "Ingrese los elementos del arreglo:" << endl;

    for (int i = 0; i < n; i++) {

        cin >> arreglo[i];
    }
}

```

```

    }

    cout << "Arreglo invertido: ";

    for (int i = n - 1; i >= 0; i--) {

        cout << arreglo[i] << " ";

    }

    cout << endl;

    return 0;

}

```

```

## Ejercicio 6

Dado un arreglo de enteros y un número x, elimina la primera aparición de x en el arreglo y muestra el arreglo resultante.

```

```cpp

#include <iostream>

using namespace std;

int main() {

    int n, x;

    cout << "Ingrese el tamaño del arreglo: ";

    cin >> n;

    int arreglo[n];

    cout << "Ingrese los elementos del arreglo:" << endl;

    for (int i = 0; i < n; i++) {

        cin >> arreglo[i];

    }

    cout << "Ingrese el número a eliminar: ";

```

```

cin >> x;

bool found = false;

for (int i = 0; i < n; i++) {

    if (arreglo[i] == x && !found) {

        found = true;

        continue;

    }

    cout << arreglo[i] << " ";

}

cout << endl;

return 0;

}

```

```

## Ejercicio 7

Crea un arreglo y permite al usuario insertar un valor y en una posición p específica.

Si p es mayor que el tamaño del arreglo, muestra un mensaje de error.

```

```cpp

#include <iostream>

using namespace std;

int main() {

    int n, p, y;

    cout << "Ingrese el tamaño del arreglo: ";

    cin >> n;

    int arreglo[n + 1];

    cout << "Ingrese los elementos del arreglo:" << endl;

```

```

for (int i = 0; i < n; i++) {

    cin >> arreglo[i];

}

cout << "Ingrese el valor a insertar y su posición: ";

cin >> y >> p;

if (p > n) {

    cout << "Error: posición fuera del rango" << endl;

} else {

    for (int i = n; i > p; i--) {

        arreglo[i] = arreglo[i - 1];

    }

    arreglo[p] = y;

    cout << "Arreglo actualizado: ";

    for (int i = 0; i <= n; i++) {

        cout << arreglo[i] << " ";

    }

    cout << endl;

}

return 0;

}

...

```

Ejercicio 8

Dado un arreglo de enteros, verifica si el arreglo está en orden ascendente. Muestra true si está ordenado y false si no lo está.

```

```cpp

#include <iostream>

```

```
using namespace std;
```

```
int main() {

 int n;

 cout << "Ingrese el tamaño del arreglo: ";

 cin >> n;

 int arreglo[n];

 bool ascendente = true;

 cout << "Ingrese los elementos del arreglo:" << endl;

 for (int i = 0; i < n; i++) {

 cin >> arreglo[i];

 }

 for (int i = 1; i < n; i++) {

 if (arreglo[i] < arreglo[i - 1]) {

 ascendente = false;

 break;

 }

 }

 cout << (ascendente ? "true" : "false") << endl;

 return 0;

}
```

```
...
```

## Ejercicio 9

Permite al usuario ingresar dos índices *i* y *j* y luego intercambia los elementos en esas posiciones en el arreglo.

Imprime el arreglo resultante.

```
```cpp
```



```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int n, i, j;
```

```
    cout << "Ingrese el tamaño del arreglo: ";
```

```
    cin >> n;
```

```
    int arreglo[n];
```

```
    cout << "Ingrese los elementos del arreglo:" << endl;
```

```
    for (int k = 0; k < n; k++) {
```

```
        cin >> arreglo[k];
```

```
    }
```

```
    cout << "Ingrese los índices a intercambiar: ";
```

```
    cin >> i >> j;
```

```
    if (i < n && j < n) {
```

```
        swap(arreglo[i], arreglo[j]);
```

```
        cout << "Arreglo resultante: ";
```

```
        for (int k = 0; k < n; k++) {
```

```
            cout << arreglo[k] << " ";
```

```
        }
```

```
    } else {
```

```
        cout << "Índices fuera de rango" << endl;
```

```
    }
```

```
    return 0;
```

```
}
```

```
...
```