



# Tecnológico de Monterrey

## Algoritmos de ordenamiento

**Integrantes:** Armando Arredondo

A01424709

## QuickSort:

```
void quickSortName(vector<Alumno> &alumnos, int izq, int der) { //Funcion para ordenar por nombre
    int i = izq; //Variable para recorrer el vector
    int j = der; //Variable para recorrer el vector
    Alumno pivote = alumnos[(izq + der) / 2]; //Variable para guardar el pivote
    Alumno aux = alumnos[0]; //Variable auxiliar para hacer el intercambio
    while (i <= j) { //Mientras i sea menor o igual a j
        while (alumnos[i].getNombre() < pivote.getNombre()) { //Mientras el nombre del alumno en la posicion i sea menor al nom
            i++; //Incremento i
        } //Fin del while
        while (alumnos[j].getNombre() > pivote.getNombre()) { //Mientras el nombre del alumno en la posicion j sea mayor al nom
            j--; //Decremento j
        } //Fin del while
        if (i <= j) { //Si i es menor o igual a j
            aux = alumnos[i]; //Guardo el alumno en la posicion i en la variable auxiliar
            alumnos[i] = alumnos[j]; //Guardo el alumno en la posicion j en la posicion i
            alumnos[j] = aux; //Guardo el alumno en la variable auxiliar en la posicion j
            i++; //Incremento i
            j--; // Decremento j
        }
    }
    if (izq < j) { //Si la posicion izquierda es menor a j
        quickSortName(alumnos, izq, j); //Llamo a la funcion quickSortName con los parametros alumnos, izq y j
    }
    if (i < der) { //Si la posicion i es menor a la posicion derecha
        quickSortName(alumnos, i, der); //Llamo a la funcion quickSortName con los parametros alumnos, i y der
    }
}
```

```
50 int main(){
51     int dato;
52     string option;
53     alumnos.push_back(Alumno("Juan", "Perez", "Ingenieria", "1234", "a00000", "Mexico", "CDMX"));
54     alumnos.push_back(Alumno("Aldo", "Cabrera", "Ingenieria", "1234", "a00001", "Mexico", "TIJ"));
55     alumnos.push_back(Alumno("Pedro", "Gomez", "Ingenieria", "1234", "a00001", "Mexico", "SLP"));
56     alumnos.push_back(Alumno("Maria", "Garcia", "Ingenieria", "1234", "a00002", "Mexico", "MOR"));
57     alumnos.push_back(Alumno("Luis", "Hernandez", "Ingenieria", "1234", "a00003", "Mexico", "GUA"));
58     alumnos.push_back(Alumno("Jose", "Martinez", "Ingenieria", "1234", "a00004", "Mexico", "CDMX"));
59     alumnos.push_back(Alumno("Ana", "Lopez", "Ingenieria", "1234", "a00005", "Mexico", "CDMX"));
60     alumnos.push_back(Alumno("Miguel", "Rodriguez", "Ingenieria", "1234", "a00006", "Mexico", "CDMX"));
61     alumnos.push_back(Alumno("Carlos", "Diaz", "Ingenieria", "1234", "a00007", "Mexico", "CDMX"));
62     alumnos.push_back(Alumno("Ricardo", "Perez", "Ingenieria", "1234", "a00008", "Mexico", "CDMX"));
63     cout << "QuickSort: " << "\n";
64     cout << "-----" << "\n";
65     quickSortName(alumnos, 0, alumnos.size()-1);
66     //cout << "Bubble sort:" << "\n";
67     //bubbleSortName(alumnos);
68     for(int i = 0; i < alumnos.size(); i++){
```

PROBLEMAS SALIDA TERMINAL JUPYTER CONSOLA DE DEPURACIÓN

```
PS C:\Users\arman\OneDrive\Escritorio\CLASE ESTRUCTURA DE DATOS\CLASE 2 - ALGORITMOS_BUSQUEDA> cd "c:\Users\arman\OneDrive\Escritorio\CLASE ESTRUCTURA DE DATOS\CLASE 2 - ALGORITMOS_BUSQUEDA" ; if ($?) { g++ testMain.cpp -o testMain } ; if ($?) { .\testMain }
QuickSort:
```

-----

Aldo  
Ana  
Carlos  
Jose  
Juan  
Luis  
Maria  
Miguel  
Pedro  
Ricardo

```
PS C:\Users\arman\OneDrive\Escritorio\CLASE ESTRUCTURA DE DATOS\CLASE 2 - ALGORITMOS_BUSQUEDA> |
```

## Bubble Sort:

```
void bubbleSortName(vector<Alumno> &alumnos) { //Funcion para ordenar por nombre
    Alumno aux = alumnos[0]; //Variable auxiliar para hacer el intercambio
    for (int i = 0; i < alumnos.size(); i++) { //Recorro el vector
        for (int j = 0; j < alumnos.size() - 1; j++) { //Recorro el vector
            if (alumnos[j].getNombre() > alumnos[j + 1].getNombre()) { //Si el nombre del alumno en la posicion j es mayor que el de la posicion j + 1
                aux = alumnos[j]; //Guardo el alumno en la posicion j en la variable auxiliar
                alumnos[j] = alumnos[j + 1]; //Guardo el alumno en la posicion j + 1 en la posicion j
                alumnos[j + 1] = aux; //Guardo el alumno en la variable auxiliar en la posicion j + 1
            } //Fin del if
        } //Fin del for
    } //Fin del for
}
```

```
53 alumnos.push_back(Alumno("Juan", "Perez", "Ingenieria", "1234", "a00000", "Mexico", "CDMX"));
54 alumnos.push_back(Alumno("Aldo", "Cabrera", "Ingenieria", "1234", "a00001", "Mexico", "TIJ"));
55 alumnos.push_back(Alumno("Pedro", "Gomez", "Ingenieria", "1234", "a00001", "Mexico", "SLP"));
56 alumnos.push_back(Alumno("Maria", "Garcia", "Ingenieria", "1234", "a00002", "Mexico", "MOR"));
57 alumnos.push_back(Alumno("Luis", "Hernandez", "Ingenieria", "1234", "a00003", "Mexico", "GUA"));
58 alumnos.push_back(Alumno("Jose", "Martinez", "Ingenieria", "1234", "a00004", "Mexico", "CDMX"));
59 alumnos.push_back(Alumno("Ana", "Lopez", "Ingenieria", "1234", "a00005", "Mexico", "CDMX"));
60 alumnos.push_back(Alumno("Miguel", "Rodriguez", "Ingenieria", "1234", "a00006", "Mexico", "CDMX"));
61 alumnos.push_back(Alumno("Carlos", "Diaz", "Ingenieria", "1234", "a00007", "Mexico", "CDMX"));
62 alumnos.push_back(Alumno("Ricardo", "Perez", "Ingenieria", "1234", "a00008", "Mexico", "CDMX"));
63 //cout << "QuickSort: " << "\n";
64 //quickSortName(alumnos, 0, alumnos.size()-1);
65 cout << "Bubble sort:" << "\n";
66 cout << "-----" << "\n";
67 bubbleSortName(alumnos);
68 for(int i = 0; i < alumnos.size(); i++){
69     cout << alumnos[i].getNombre() << endl;
70 }
71 return 0;
```

PROBLEMAS SALIDA TERMINAL JUPYTER CONSOLA DE DEPURACIÓN

```
PS C:\Users\arman\OneDrive\Escritorio\CLASE ESTRUCTURA DE DATOS\CLASE 2 - ALGORITMOS_BUSQUEDA> cd "c:\Users\arman\OneDrive\Escritorio\CLASE 2 - ALGORITMOS_BUSQUEDA\" ; if ($?) { g++ testMain.cpp -o testMain } ; if ($?) { .\testMain }
Bubble sort:
```

-----

Aldo  
Ana  
Carlos  
Jose  
Juan  
Luis  
Maria  
Miguel  
Pedro  
Ricardo