Một số nội dung Khi hỏi về Sort:

1. Ý tưởng
2. Thuật toán mã giả/cài đặt ngôn ngữ C++
3. Chạy tay (chi tiết từng bước)
4. Độ phức tạp
5. Trình bày ý tưởng, mã giả, code, chạy tay các thuật toán Sắp xếp
6. Dùng mảng 1 chiều để quản lý một bãi đậu xe mô tô gồm N xe. Biết rằng mỗi xe mô tô gồm các thông tin sau: **Biển số xe** (dạng chuỗi), **Phân khối xe** (kiểu nguyên).
7. Khai báo các cấu trúc dữ liệu cần thiết để quản lý danh sách xe.
8. Cài đặt hàm cho phép nhập và xuất mảng gồm N xe.
9. Sắp xếp danh sách xe **giảm dần** theo **Biển số xe** sử dụng thuật toán **Insertion Sort**.
10. Viết màn main để chạy các hàm trên.

| struct XE{      string BSX;      int PKX;  };  void InsertionSort(XE a[], int n) {      int pos;      int x; //lưu giá trị a[i] tránh bị ghi đè khi dời chỗ các phần tử.      for (int i=1; i<n; i++)   //đoạn a[0] đã sắp      {          x = a[i];          pos = i-1; // tìm vị trí chèn x          while ((pos >= 0) && (a[pos].BSX < x.BSX)) {              //kết hợp dời chỗ các phần tử sẽ đứng sau x trong dãy mới              a[pos + 1] = a[pos];              pos--;          }          a[pos+1] = x; // chèn x vào dãy      }  }  int main(){      int A[MAXN], n;      Input(A, n);      InsertionSort(A, n);      Output(A, n);      return 0;  } |
| --- |

1. Thực hiện chạy tay các thuật toán sắp xếp trên dãy số sau:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 0 | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |
|  |  |  |  |  |  |  |  |  |  |

# **SELECTION SORT**

…

# **INSERTION SORT**

Với i=1: Chèn 0 vào dãy gồm có: 5, vị trí chèn 0 là vị trí 0

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 5 | **0** | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |
| → | 0 | 5 | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |

Với i=2: Chèn 4 vào dãy gồm có: 0 5, vị trí chèn 4 là vị trí 1

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 5 | **4** | 12 | 8 | 1 | 9 | 15 | 7 | 6 |
| → | 0 | 4 | 5 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |

Với i=3: Chèn 12 vào dãy gồm có: 0 4 5, vị trí chèn 12 là vị trí 3

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 4 | 5 | **12** | 8 | 1 | 9 | 15 | 7 | 6 |
| → | 0 | 4 | 5 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |

Với i=4: Chèn 8 vào dãy gồm có: 0 4 5 12, vị trí chèn 8 là vị trí 3

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 4 | 5 | 12 | **8** | 1 | 9 | 15 | 7 | 6 |
| → | 0 | 4 | 5 | 8 | 12 | 1 | 9 | 15 | 7 | 6 |

Với i=5: Chèn 1 vào dãy gồm có: 0 4 5 8 12, vị trí chèn 1 là vị trí 1

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 4 | 5 | 8 | 12 | **1** | 9 | 15 | 7 | 6 |
| → | 0 | 1 | 4 | 5 | 8 | 12 | 9 | 15 | 7 | 6 |

Với i=6: Chèn 9 vào dãy gồm có: 0 1 4 5 8 12, vị trí chèn 9 là vị trí 5

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 4 | 5 | 8 | 12 | **9** | 15 | 7 | 6 |
| → | 0 | 1 | 4 | 5 | 8 | 9 | 12 | 15 | 7 | 6 |

Với i=7: Chèn 15 vào dãy gồm có: 0 1 4 5 8 9 12, vị trí chèn 15 là vị trí 7

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 4 | 5 | 8 | 9 | 12 | **15** | 7 | 6 |
| → | 0 | 1 | 4 | 5 | 8 | 9 | 12 | 15 | 7 | 6 |

Với i=8: Chèn 7 vào dãy gồm có: 0 1 4 5 8 9 12 15, vị trí chèn 7 là vị trí 4

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 4 | 5 | 8 | 9 | 12 | 15 | **7** | 6 |
| → | 0 | 1 | 4 | 5 | 7 | 8 | 9 | 12 | 15 | 6 |

Với i=9: Chèn 6 vào dãy gồm có: 0 1 4 5 7 8 9 12 15, vị trí chèn 6 là vị trí 4

|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 4 | 5 | 7 | 8 | 9 | 12 | 15 | **6** |
| → | 0 | 1 | 4 | 5 | 6 | 7 | 8 | 9 | 12 | 15 |

* Mảng đã có thứ tự: 0 1 4 5 6 7 8 9 12 15

# **QUICK SORT**

QuickSort dãy

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 0 | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |

l=0, r=9, m=(l+r)/2=4 => pivot=8. Bắt đầu 2 đường chạy: i=0, j=9

| 5 | 0 | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | i |  |  |  |  |  | j | Swap(a[i], a[j]) |

| 5 | 0 | 4 | 6 | 8 | 1 | 9 | 15 | 7 | 12 |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | i |  |  |  | j |  | Swap(a[i], a[j]) |

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 0 | 4 | 6 | 7 | 1 | 9 | 15 | 8 | 12 |  |
|  |  |  |  |  | j | i |  |  |  | (i>j) break |
|  |  |  |  |  |  |  |  |  |  |  |

* QuickSort đoạn

| 0 | 1 | 2 | 3 | 4 | 5 |
| --- | --- | --- | --- | --- | --- |
| 5 | 0 | 4 | 6 | 7 | 1 |

l=0, r=5, m=(l+r)/2=2 => pivot=4. Bắt đầu 2 đường chạy: i=0, j=5

| 5 | 0 | 4 | 6 | 7 | 1 |  |
| --- | --- | --- | --- | --- | --- | --- |
| i |  |  |  |  | j | Swap(a[i], a[j]) |

| 1 | 0 | 4 | 6 | 7 | 5 |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | i  j |  |  |  | Swap(a[i], a[j]) |
|  |  |  |  |  |  |  |
| 0 | 1 | 2 | 3 | 4 | 5 |  |
| 1 | 0 | 4 | 6 | 7 | 5 |  |
|  | i |  | j |  |  | (i>j) break |

* QuickSort đoạn

| 0 | 1 |
| --- | --- |
| 1 | 0 |

l=0, r=1, m=(l+r)/2=0 => pivot=1. Bắt đầu 2 đường chạy: i=0, j=1

| 1 | 0 |  |
| --- | --- | --- |
| i | j | Swap(a[i], a[j]) |

| 0 | 1 |  |
| --- | --- | --- |
| j | i | (i>j) break |

* QuickSort đoạn

| 3 | 4 | 5 |
| --- | --- | --- |
| 6 | 7 | 5 |

l=3, r=5, m=(l+r)/2=4 => pivot= a[4]=7. Bắt đầu 2 đường chạy: i=3, j=5

| 6 | 7 | 5 |  |
| --- | --- | --- | --- |
|  | i | j | Swap(a[i], a[j]) |

| 3 | 4 | 5 |  |
| --- | --- | --- | --- |
| 6 | 5 | 7 |  |
|  | j | i | (i>j) break |

* QuickSort đoạn

| 3 | 4 |
| --- | --- |
| 6 | 5 |

l=3, r=4, m=(l+r)/2=3 => pivot= a[3]=6. Bắt đầu 2 đường chạy: i=3, j=4

| 6 | 5 |  |
| --- | --- | --- |
| i | j | Swap(a[i], a[j]) |

| 6 | 5 |  |
| --- | --- | --- |
| i | j | (i>j) break |

* QuickSort đoạn

| 6 | 7 | 8 | 9 |
| --- | --- | --- | --- |
| 9 | 15 | 8 | 12 |

l=6, r=9, m=(l+r)/2=7 => pivot=15. Bắt đầu 2 đường chạy: i=6, j=9

| 9 | 15 | 8 | 12 |  |
| --- | --- | --- | --- | --- |
|  | i |  | j | Swap(a[i], a[j]) |

| 6 | 7 | 8 | 9 |  |
| --- | --- | --- | --- | --- |
| 9 | 12 | 8 | 15 |  |
|  |  | j | i | (i>j) break |

* QuickSort đoạn

| 6 | 7 | 8 |
| --- | --- | --- |
| 9 | 12 | 8 |

l=6, r=8, m=(l+r)/2=7 => pivot=12. Bắt đầu 2 đường chạy: i=6, j=8

| 9 | 12 | 8 |  |
| --- | --- | --- | --- |
|  | i | j | Swap(a[i], a[j]) |

| 6 | 7 | 8 |  |
| --- | --- | --- | --- |
| 9 | 8 | 12 |  |
|  | j | i | (i>j) break |

* QuickSort đoạn

| 6 | 7 |
| --- | --- |
| 9 | 8 |

l=6, r=7, m=(l+r)/2=6 => pivot=9. Bắt đầu 2 đường chạy: i=6, j=7

| 9 | 8 |  |
| --- | --- | --- |
| i | j | Swap(a[i], a[j]) |

| 6 | 7 |  |
| --- | --- | --- |
| 8 | 9 |  |
| j | i | (i>j) break |

* Mảng đã được sắp xếp: 0 1 4 5 6 7 8 9 12 15

# **MERGE SORT (Đệ quy)**

…

# **MERGE SORT (Không đệ quy)**

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 0 | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |

**Với i=1:**

| A = | 5 | 0 | 4 | 12 | 8 | 1 | 9 | 15 | 7 | 6 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| B = | 5 | 4 | 8 | 9 | 7 |  |  |  |  |  |
| C = | 0 | 12 | 1 | 15 | 6 |  |  |  |  |  |
| →A= | 0 | 5 | 4 | 12 | 1 | 8 | 9 | 15 | 6 | 7 |

**Với i=2:**

| A = | 0 | 5 | 4 | 12 | 1 | 8 | 9 | 15 | 6 | 7 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| B = | 0 | 5 | 1 | 8 | 6 | 7 |  |  |  |  |
| C = | 4 | 12 | 9 | 15 |  |  |  |  |  |  |
| →A= | 0 | 4 | 5 | 12 | 1 | 8 | 9 | 15 | 6 | 7 |

**Với i=4:**

| A = | 0 | 4 | 5 | 12 | 1 | 8 | 9 | 15 | 6 | 7 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| B = | 0 | 4 | 5 | 12 | 6 | 7 |  |  |  |  |
| C = | 1 | 8 | 9 | 15 |  |  |  |  |  |  |
| →A= | 0 | 1 | 4 | 5 | 8 | 9 | 12 | 15 | 6 | 7 |

**Với i=8:**

| A = | 0 | 1 | 4 | 5 | 8 | 9 | 12 | 15 | 6 | 7 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| B = | 0 | 1 | 4 | 5 | 8 | 9 | 12 | 15 |  |  |
| C = | 6 | 7 |  |  |  |  |  |  |  |  |
| →A= | 0 | 1 | 4 | 5 | 6 | 7 | 8 | 9 | 12 | 15 |

# **HEAP SORT (xem file bài tập về Heap sort)**