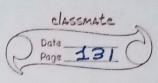
1.2.1 Bubble borting - Real world Application is not there e.g => 51413 2/1 1/2/3/4/5 Generally sort means ascending Algorithm 1. swap the adjacents of needed till we get the all the array sorted. I Largest at its position. 5/4/3/2/1 LA comparisons Observation: 1st largest gone to

last.



Classmate Date 132

工具工

N=5

4 inerations

$$Sh = 1 + 2 + 3 \dots (h-2) + (h-1)$$

$$Sn = n(n-1)$$
2

$$= n^2h$$

$$O(n) = O(n^2 - h)$$

Tome complexity.

VOID bubble Sort (vector < int > fv) {
int n = v.size();

4



Selection Soit

selecting minimum element and putting at right position.

44 | 33 55 | 22 | 11

1 11 | 33 | 55 | 44 22 | 44

(2) 11/22/55/33/44

(3) 11/22/33/55/44

(4) 11/22 33/44/55

Note: For ith iteration, pick smallest element from i to (n-1) index & swap it with ith element.

Time complexity + Och2)

code:

Void Selection Sort (vector lint > f v) ~ int n = v.size();

forcint =0 ; i < n-1; i++) ?

for int minIndex = i;

// ith element hi smallest hai

forcintj=i+1;j<h;j++)?

of CV[j] < V[min Index])?

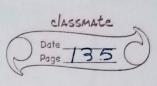
mihIndex=j;

J //swap ith and min Index elements;
y swap (V[i], V[min Index]);

Throa agil. Insertion Sout. 5 4/3/2/1 @ 4 15 3 2 11 3 314/5/2/1 4) 12131415/21 (5) 11/2/3/4/5/ Time complexity =0(n2) Space Complexity > 0 (1) Code:void insertionSort (vector int > f V) { int n= V. s9Ze(); 119=0, chhodh detahu forcihti=1; i<h;++i) < Int Key = VCII, ?ht; = 1-1 //move on elements of an [o...i-] that are greater than Il key to one position ahead of the current position while (\$>=off v()]>Key) { VCj+1] = VCj];

2 V[;+1) = Key; 11 insertion

3



Custom comparator
V
Custom camparator for as descending
Sorting
bool mycomp (9nt fa, 9ht fb) 2
retuin arbi
e.gn02
bool my compfortst Index (vector cint Aa Vector cint > 4b)
Vector <iht>fb)</iht>
return C[1] < b[1];
3