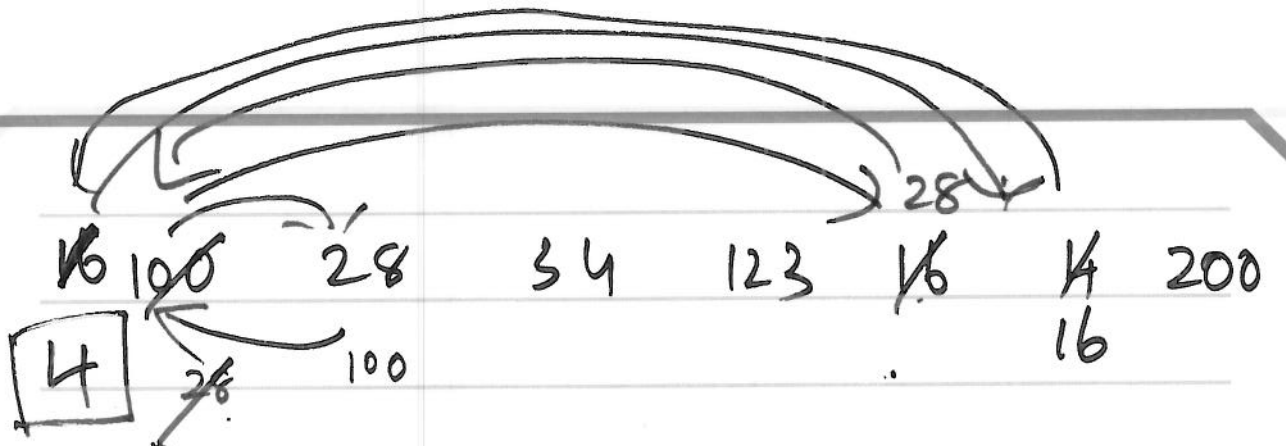
- Assignment 6 Posted
- Technical topics
  - sorting
  - maintain a database
- Midterm Exam.
- Recursion

SEARCHING

SORTING data. {arranging data}

- ascending ⇒ lower key values ordered first
- descending ⇒ higher key values ordered first.



BUBBLE SORT

└ on each pass through the list the "smallest" value "bubbles" to that position

```
int A[];
```

```
for (int i=0; i < MAXSIZE; i++)
    for(int j = i+1; j<MAXSIZE; j++)
        if (A[j] < A[i])
            swap(A[j], A[i]);
```

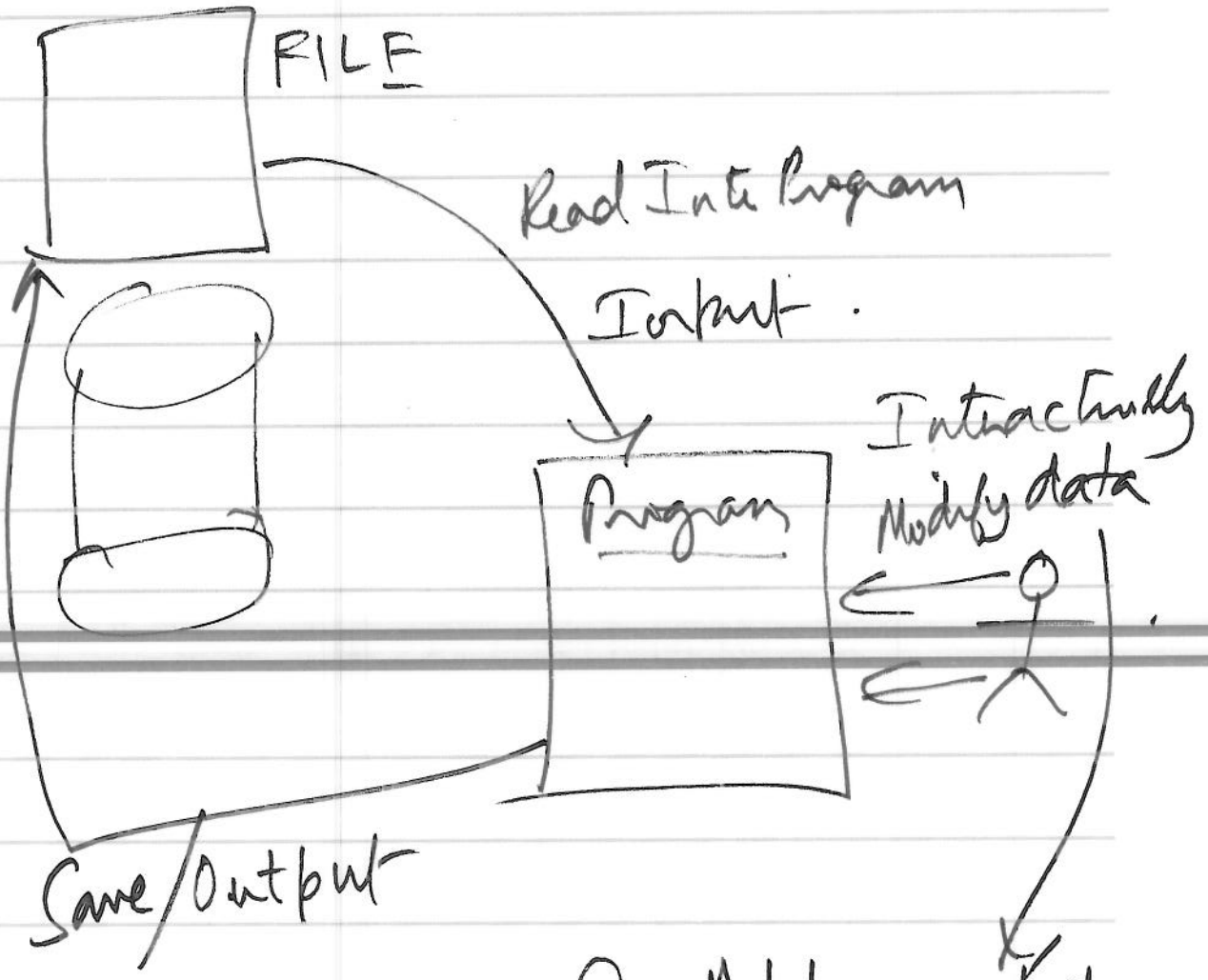
BUBBLE sort algorithm

$O(n^2)$

└→ Heap Sort
└→ Quick Sort

$O(n \log n)$

"Database"



Interactive commands

- Add new data
- Modify data
- Delete data
- Search for data
- Read from file (Input)
- Write/Save to file

⟹ interactive input

char command;

cin >> command; ✓

Sorting Data. ✓

/ Choice 1: Read all data

Use Bubble Sort ✓

Choice 2: Insertion Sort.

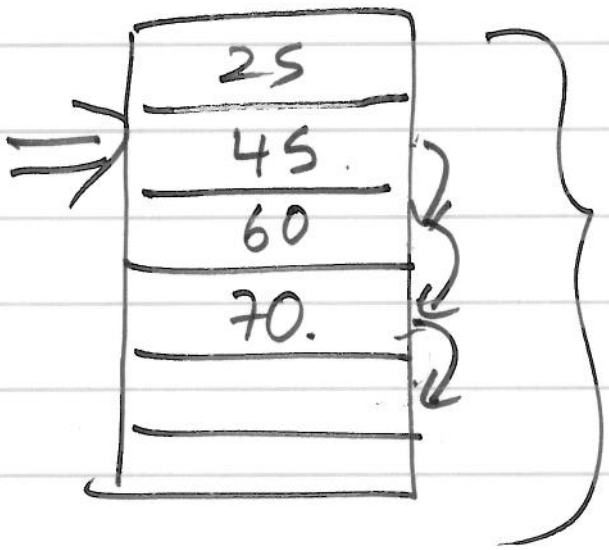
└ Insert elements in to the right place as they arrive

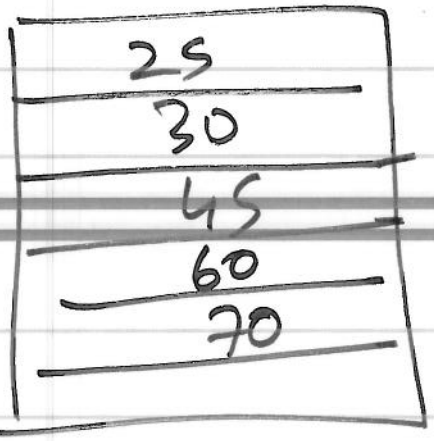
int A[];

Input 25

Input 45

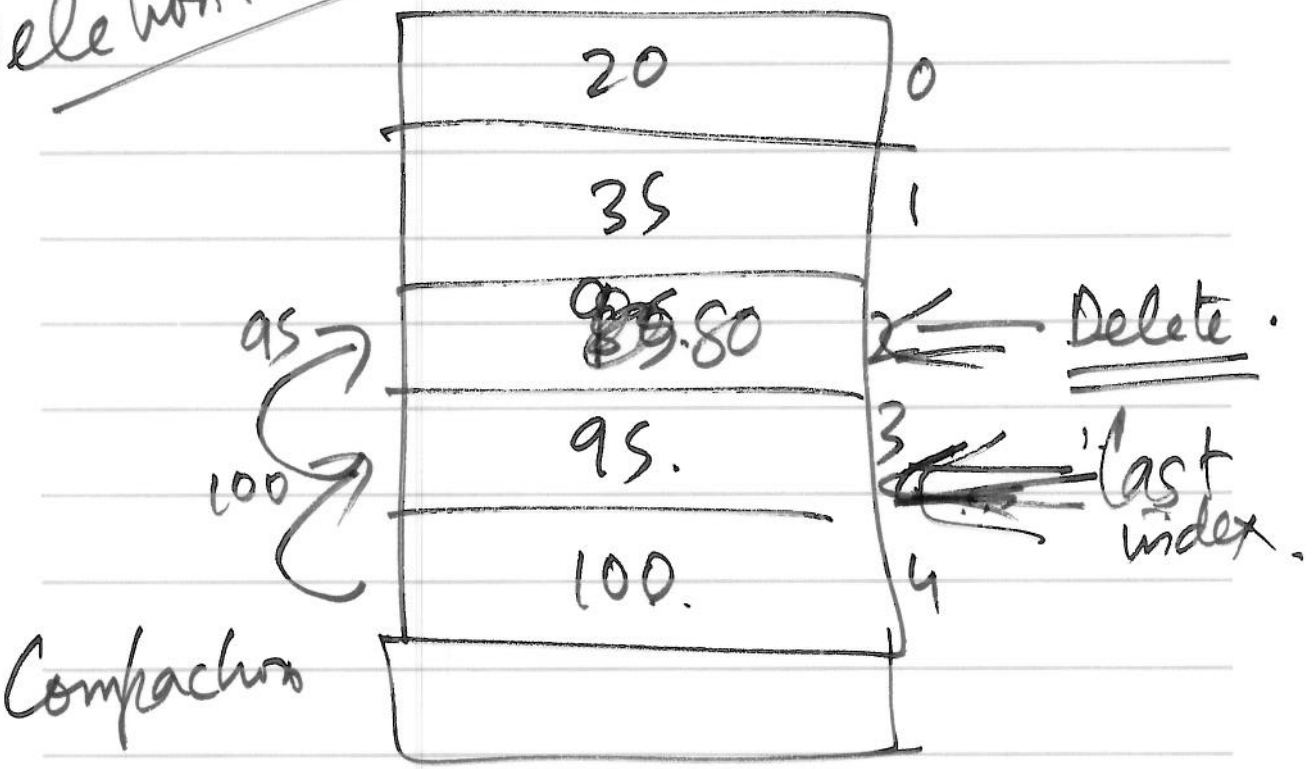
Input 30





Insertion Problem Solved

Deletion Problems



Insertion Sort

- Maintain sorted property of Array all the time

Midterm Exam

- MHP 106
- 6:30 p.m.

Exam: 75 minutes.

- Lecture after test.

Open Book; Open Notes

Bring Yourown

No programming
- full programs

2 sections
- Objective
- Code Understanding
- Hand Simulation
- Write Code snippets

Hand Simulation example

```
for (int i = 0; i < 5; i++)
{
    A[i] = i;
}
```

index i	0	1	2	3	4
A	0	1	2	3	4

Hand Simulation

Total = 11 X

Material

Everything up to 8 including "Database" lecture.

Main Focus upto Assignment 5

- int / string / chan
- if
- while / for loop
- switch
- Functions
  - reference
  - value
- Array
- Struct

Prehone:

— look @ assignments

— look @ examples on web

— Take sample test

— Compare to solutions

Preferably not on Thursday morning!

RECURSION:

When a function calls ITSELF

Factorial (n) = n x n-1 x n-2 x n-3 x ... x 2 x 1

Factorial (n) = n x Factorial (n-1)

Factorial (n-1) = (n-1) Factorial (n-2)

⋮

Factorial (3) = 3) Factorial (2)

Factorial (2) = 2 Factorial (1)

Factorial (1) = 1

Recursive formulation.

```
int factorial (int n)
{
    if (n <= 0) return 0;
    if (n == 1) return 1;
    return n * factorial (n-1);
}
```

Recursive Function

Factorial (4)

return 4 * Factorial (3)

return 3 * Factorial (2)

return 2 * Factorial (1)

factorial (1) = 1