Sort

Arnaud Malapert, Gilles Menez, Marie Pelleau

Master Informatique, Université Côte d'Azur

Popular sorting algorithm

- **Insertion sort**
- Selection sort
- Bubble sort
- **Merge sort**
- Quicksort

Insertion sort

Using arrays

Insertion sort

Using arrays

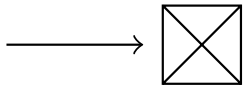
Insertion sort

Using arrays

Insertion sort

Using arrays

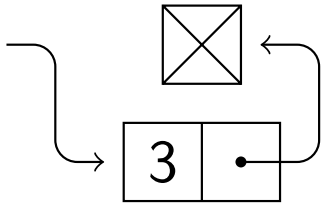
Using linked lists



Insertion sort

Using arrays

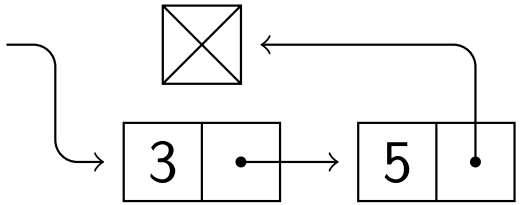
Using linked lists



Insertion sort

Using arrays

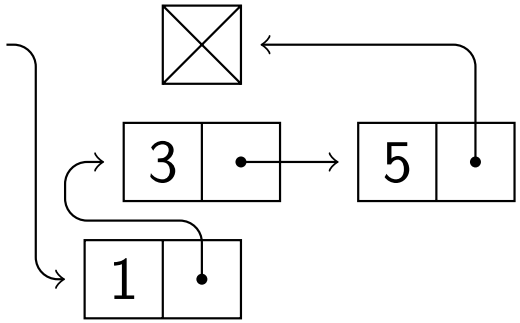
Using linked lists



Insertion sort

Using arrays

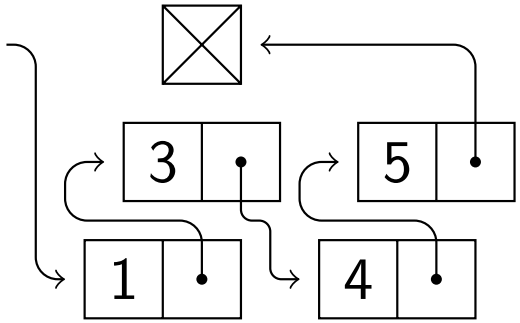
Using linked lists



Insertion sort

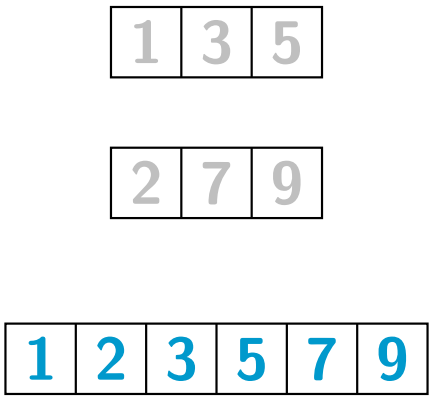
Using arrays

Using linked lists



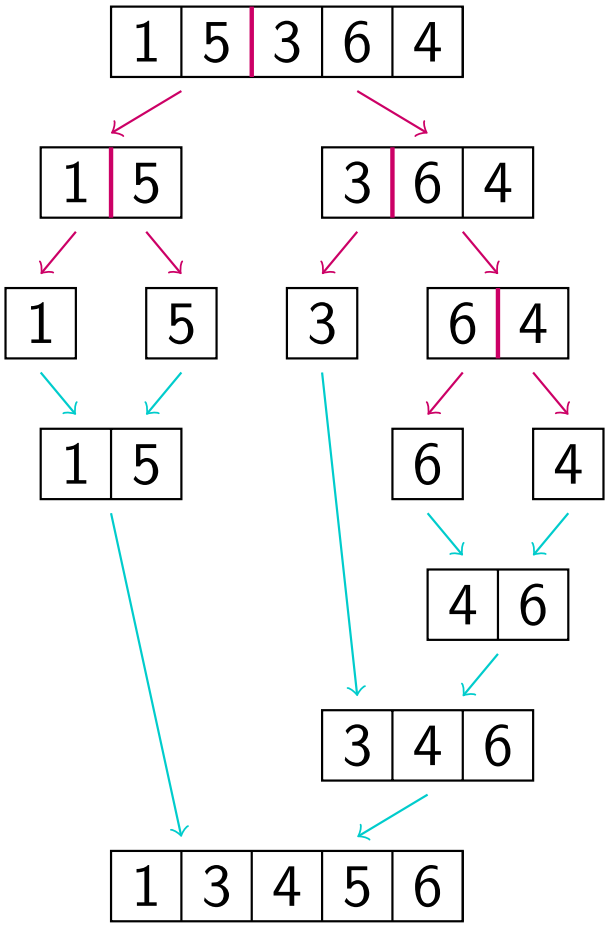
Merge sort

Merging two sorted arrays



Merge sort

Dichotomic sort



Exercise 1: Candy Distribution

Statement

Given N integers indicating the number of students in each of Alice's classes, and N integers corresponding to the price of a type of candy. Knowing that all students in the same class will receive the same kind of candy, compute the least amount of money Alice must spend to give a candy to each of her students.

Example

Input:

5
10 80 37 22 109
6 8 8 20 15

Output:

2120

Exercise 1: Candy Distribution

Statement

Given N integers indicating the number of students in each of Alice's classes, and N integers corresponding to the price of a type of candy. Knowing that all students in the same class will receive the same kind of candy, compute the least amount of money Alice must spend to give a candy to each of her students.

What problems can arise?

- What do we know of N?
- Of the number of students?
- Of the price of the candies?
- How great can the solution be?
  $\Longrightarrow$ Are integers big enough for the solution?

Exercise 1: Candy Distribution

Solution 1: Homemade

Using two lists or arrays with insertion sort

Solution 2: Integrated

Using two arrays and the sort procedure in your preferred language

Exercise 1: Candy Distribution

More test cases

Input:

4
1 10 2 1
1 2 4 2
5
10 80 37 22 89
6 8 8 20 15
0

Output:

20
2000

Exercise 2: Inversion Count

Statement

Given an array A of integers. If $i < j$ and $A[i] > A[j]$ then the pair (i, j) is called an inversion of A. Count the number of inversions of A

Example

Input:

2 3 8 6 1

Output:

5

Exercise 2: Inversion Count

Statement

Given an array A of integers. If $i < j$ and $A[i] > A[j]$ then the pair (i, j) is called an inversion of A. Count the number of inversions of A

What problems can arise?

- How great can the array be?
- How great can the values in the array be?
- How great can the solution be?
  $\Longrightarrow$ Are integers big enough for the solution?

Exercise 2: Inversion Count

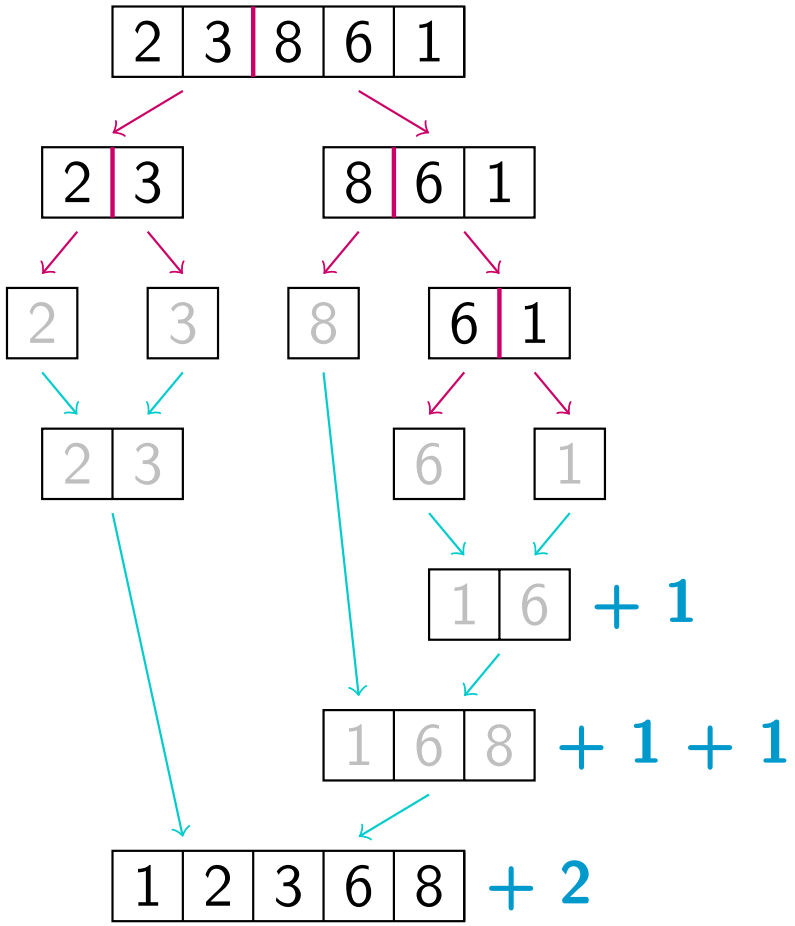
Solution 1: Brut Force

```
read n on standard input
read the array A on standard input

result ← 0

for i from 0 to n − 2
    for j from i + 1 to n − 1
        if A[i] > A[j] then
            result ← result + 1

print result
```

Exercise 2: Inversion Count

Solution 2: Using Merge sort

Key idea: if there are no inversions, then during the merge, all the elements of the left array should be added before any element of the right array



Exercise 2: Inversion Count

More test cases

Input:

```
3
6
1 2 3 4 5 6
8
5 1 4 2 6 2 6 2
1
1
```

Output:

```
0
11
0
```

Exercise 3: It's a Murder

Statement

Given an array of integers, for each number sum the previous strictly smaller number

Example

Input:	Output:
1 5 1 5 3 6 4	15

Solution: Elegant

```
Using Merge sort
```

Exercise 4: Yodaness Level

Statement

Given a statement as Yoda says it and the same statement as it should be said normally count the number of pairs of words that changed their order

Example

Input:

1
6
much to learn you still have
you still have much to learn

Output:

9

Solution: Elegant

Using Merge sort