

# ***SORTING***

*A Sorting Algorithm is used to rearrange a given array or list elements*

*according to a comparison operator on the elements. The comparison operator is used to decide the new order of element in the respective data structure.*

**Various algorithms are better suited to some of these situations**



## **Internal Sort**

- The data to be sorted is all stored in the computer's main memory.



## **External Sort**

- Some of the data to be sorted might be stored in some external, slower, device.



## **In Place Sort**

- The amount of extra space required to sort the data is constant with the input size.

# ***INSERTION SORT***

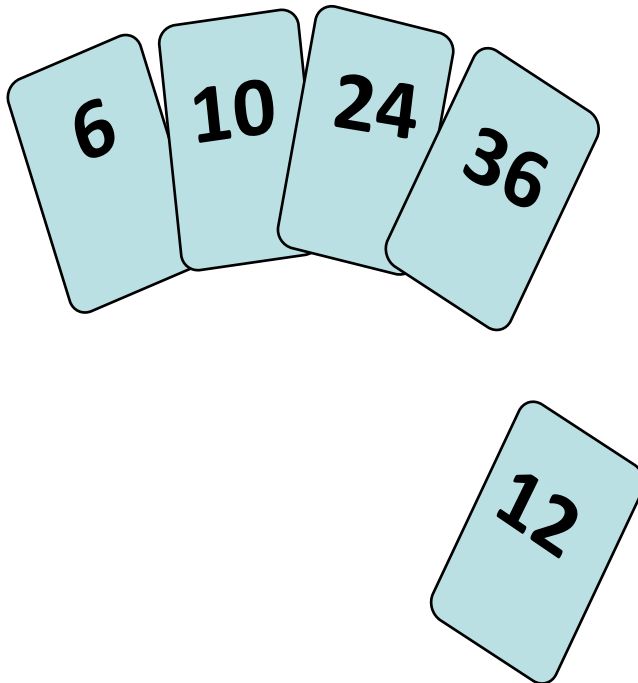


Insertion sort is a sorting algorithm that places an unsorted element at its suitable place in each iteration.

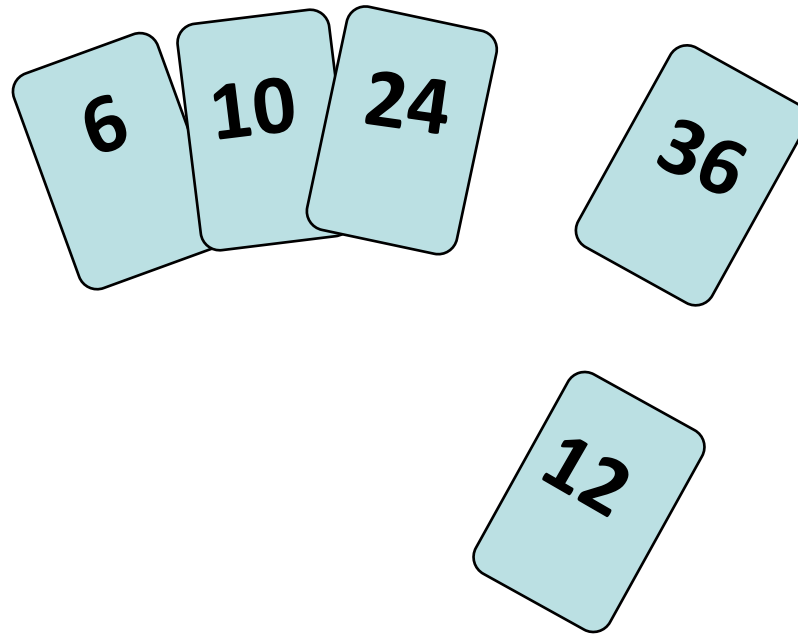
- Idea: like sorting a hand of playing cards
  - Start with an empty left hand and the cards facing down on the table.
  - Remove one card at a time from the table, and insert it into the correct position in the left hand
    - compare it with each of the cards already in the hand, from right to left
  - The cards held in the left hand are sorted
    - these cards were originally the top cards of the pile on the table

# ***INSERTION SORT***

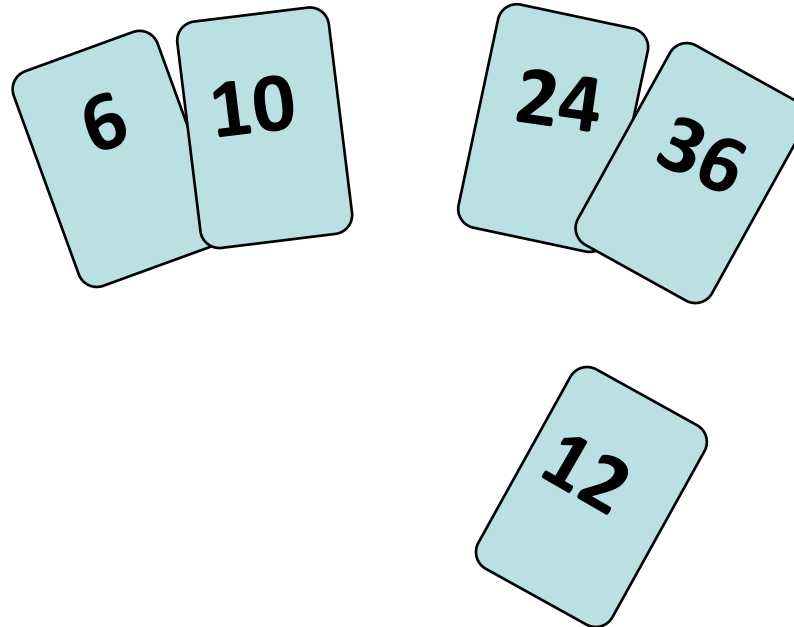
To insert 12, we need to make room for it by moving first 36 and then 24.



# ***INSERTION SORT***



# ***INSERTION SORT***



# ***INSERTION SORT***

Before going through the program, let's see the steps of insertion sort with the help of an **example**.

Input elements: **89 17 8 12 0**

*Step 1:* **89 17 8 12 0** (the bold elements are sorted list and non-bold unsorted list)

*Step 2:* **17 89 8 12 0** (each element will be removed from unsorted list and placed at the right position in the sorted list)

*Step 3:* **8 17 89 12 0**

*Step 4:* **8 12 17 89 0**

*Step 5:* **0 8 12 17 89**

# ***INSERTION SORT***

Algorithm:--

```
for(i=1;i<n;i++)  
{  
    k=a[i];  
    j=i-1;  
    while(j>=0 && a[j]>k)  
    {  
        a[j+1]=a[j];  
        j=j-1;  
    }  
    a[j+1]=k;  
}
```

# INSERTION SORT

```
#include<stdio.h>
int main()
{
    int i, j, count, temp, number[25];
    printf("How many numbers u are going to enter?: ");
    scanf("%d",&count);
    printf("Enter %d elements: ", count);
    // This loop would store the input numbers in array
    for(i=0;i<count;i++)
        scanf("%d",&number[i]);
    // Implementation of insertion sort algorithm
    for(i=1;i<count;i++)
    {
        temp=number[i];
        j=i-1;
        while((temp<number[j])&&(j>=0))
        {
            number[j+1]=number[j];
            j=j-1;
        }
        number[j+1]=temp;
    }
    printf("Order of Sorted elements: ");
    for(i=0;i<count;i++)
        printf(" %d",number[i]);
    return 0;
}
```