

SEARCHING AND SORTING ALGORITHMS

SEARCHING ALGORITHMS

Linear Search

```
int linear_search(int a[], int x){
    int i, n;
    n=10;

    for(i=0; i<n; i++){
        if(a[i] == x) return 1;
    }
    return 0;
}
```

Binary Search

```
int binary_search(int a[], int x){
    int lower, upper, middle, n;
    n=10;
    lower=0;
    upper=n-1;

    while(lower<=upper){
        middle=(lower+upper)/2;
        if(x>a[middle]) lower=middle+1;
        else if(x<a[middle]) upper=middle-1;
        else return 1;
    }
    return 0;
}
```

SORTING ALGORITHMS

are algorithms for arranging elements of a list in a certain order. Here are some simple sorting algorithms: the bubble sort, insertion sort and selection sort.

Bubble Sort

The bubble sort works by comparing each item in the list with the item next to it, and swapping them if required. The algorithm repeats this process until it makes a pass all the way through the list without swapping any items (in other words, all items are in the correct order). This causes larger values to "bubble" to the end of the list while smaller values "sink" towards the beginning of the list.

```
void bsort(int a[], int n){
    int i, j, temp;

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

Insertion Sort

It inserts each item into its proper place in the final list.

```
void isort(int a[], int n){
    int i, j, temp;

    for(i=1; i<n; i++){
        for(j=i; j>0; j--){
            if(a[j]<a[j-1]){
                temp = a[j];
                a[j] = a[j-1];
                a[j-1] = temp;
            }
            else break;
        }
    }
}
```

Selection Sort

The selection sort works by selecting the smallest unsorted item remaining in the list, and then swapping it with the item in the next position to be filled.

```
void ssort(int a[], int n){
    int i, j, temp;

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