VTC Academy
Think Ahead, Beyond Boundaries

BASIC PROGRAMMING LANGUAGE

LESSON 8

Sorting and Searching Algorithms

CONTENT

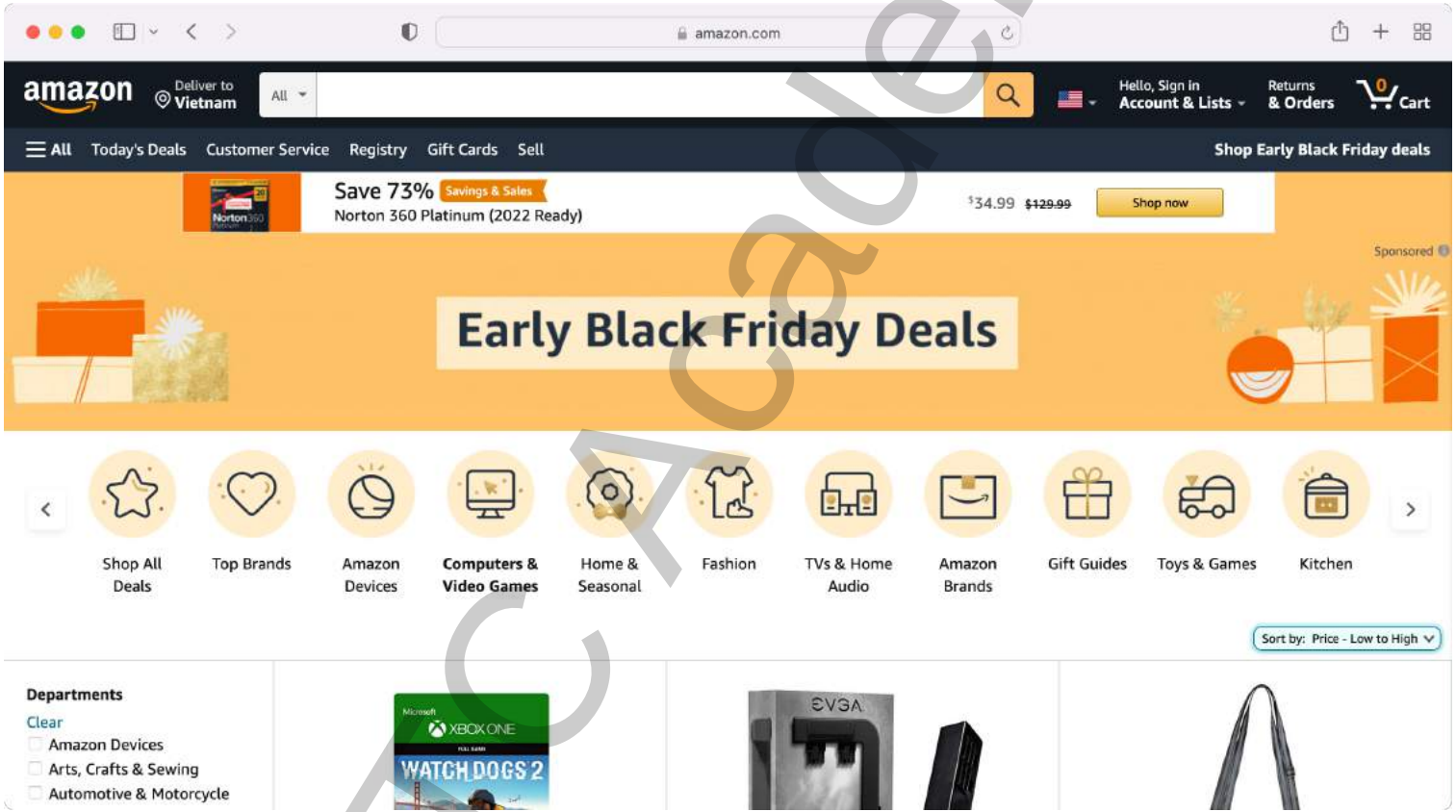
VTC Academy
Think Ahead, Beyond Boundaries

Introduction to Sorting and Searching

VTC Academy
Think Ahead, Beyond Boundaries

- Searching is very important work every day. For example: search a book, your bike, your clothes,…
- Same is the life of a computer, there is so much data stored in it, that whenever a user asks for some data, computer has to search it's memory to look for the data and make it available to the user.
- Sorting is nothing but arranging the data in ascending or descending order. The term sorting came into picture, as humans realised the importance of searching quickly.
- Sorting arranges data in a sequence which makes searching easier and more useful reports.

Introduction to Searching and Sorting



Sorting Algorithms

VTC Academy
Think Ahead, Beyond Boundaries

- Sorting involves arranging the array data in a specified order such as ascending or descending.
- Data in an array is easier to search when the array is sorted.
- A sorting algorithm is used to arrange elements of an array/list in a specific order.
- Two of methods to sort arrays: Selection Sort and Bubble Sort.

Bubble Sort

VTC Academy
Think Ahead, Beyond Boundaries

- Bubble sort is a simple sorting algorithm.
- This sorting algorithm is comparison based algorithm in which each pair of adjacent elements is compared and elements are swapped if they are not in order.
- Just like the movement of air bubbles in the water that rise up to the surface, each element of the array move to the end in each iteration. Therefore, it is called a bubble sort.

Bubble Sort

VTC Academy
Think Ahead, Beyond Boundaries

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

Selection Sort

VTC Academy
Think Ahead, Beyond Boundaries

- Selection sort is a simple sorting algorithm.
- Selection sort is a sorting algorithm that selects the smallest element from an unsorted list in each iteration and places that element at the beginning of the unsorted list.
- In every iteration of selection sort, the minimum element (considering ascending order) from the unsorted subarray is picked and moved to the sorted subarray.

Selection Sort

VTC Academy
Think Ahead, Beyond Boundaries

- Algorithm:

```
selection sort (array, size)
    repeat (size – 1) times
    set the first unsorted element as the minimum
    for each of the unsorted elements
        if element < current minimum
            set element as new minimum
     swap minimum with first unsorted position
end selection sort
```

Selection Sort Example

VTC Academy
Think Ahead, Beyond Boundaries

```
int a[] = {5, 2, 8, 9, 1, 6};
int i, j, position, temp;
for (i = 0; i < 6; i++) {
    position = i;
    for (j = i + 1; j < 6; j++) {
        if (a[position] > a[j]) {
            position = j;
        }
    }
    if (position != i) {
        temp = a[i];
        a[i] = a[position];
        a[position] = temp;
    }
}
```

Searching Algorithms

VTC Academy
Think Ahead, Beyond Boundaries

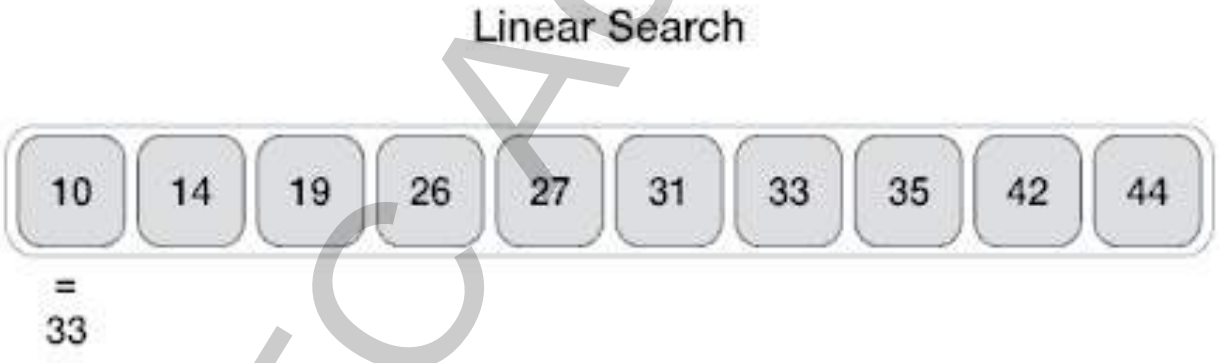
- Searching algorithms are designed to check for an element or retrieve an element from any data structure where it is stored.
- Searching a list for a particular item is a common task. In real applications, the list items often are records (e.g. student records), and the list is implemented as an array of objects
- The concept of efficiency (or complexity) is important when comparing algorithms
- For long lists and tasks, like searching, that are repeated frequently, the choice among alternative algorithms becomes important because they may differ in efficiency

Searching Algorithms

- **Sequential Search:** In this, the list or array is traversed sequentially and every element is checked.
- For example: Linear Search.
- **Interval Search:** These algorithms are specifically designed for searching in sorted data-structures. These type of searching algorithms are much more efficient than Linear Search as they repeatedly target the center of the search structure and divide the search space in half.
- For example: Binary Search.

Linear Searching

- Linear search is a very simple search algorithm. In this type of search, a sequential search is made over all items one by one.
- Every items is checked and if a match founds then that particular item is returned otherwise search continues till the end of the data collection.



Linear Searching

- Pseudocode:

```
procedure linear_search(list, value)
    for each item in the list
        if match item == value
            return the item's location
        end if
    end for
end procedure
```

Linear Searching Example

VTC Academy
Think Ahead, Beyond Boundaries

```c
int numbers[11], a, i, result = 0;
for (i = 0; i < 10; i++) {
    printf("Enter number: ");
    scanf("%d", &numbers[i]);
}
printf("Enter a number to search: \n");
scanf("%d", &a);
for (i = 0; i < 10; i++) {
    if (numbers[i] == a) {
        printf("%d is present at location %d\n", a, i + 1);
        result = 1;
        break;
    }
}
if (result == 0) {
    printf("%d isn't present in the array\n", a);
}
```

Binary Search

VTC Academy
Think Ahead, Beyond Boundaries

- Binary Search is a search algorithm that is used to find the position of an element in a sorted array. In this approach, the element is always searched in the middle of a portion of an array.
- This search algorithm works on the principle of divide and conquer.
- Binary search search a particular item by comparing the middle most item of the collection.
- Binary search implementation:
  - Iteration
  - Recursion

Binary Search

- Iteration algorithm:

```
do until the pointers low and high meet each other
    mid = (low + high) / 2
    if (x == arr[mid])
        return mid
    else if (x > arr[mid]) // x is on the right side
        low = mid + 1
    else // x is on the left side
        high = mid – 1
end loop
```

Binary Search

Binary Search Example

```
while (low <= high) {
    int mid = low + (high - low) / 2;
    if (array[mid] == x) {
        return mid;
    }
    if (array[mid] < x) {
        low = mid + 1;
    }
    else {
        high = mid - 1;
    }
}
return -1;
```

Summary

- Searching a list for a particular item is a common task. In real apps, the list items often are records.
- Linear search is a very simple search algorithm. In this type of search, a sequential search is made over all items one by one.
- Sorting involves arranging the array data in a specified order such as ascending or descending.
- Selection algorithm is a in-place comparison based algorithm in which the list is divided into two parts, sorted part at left end and unsorted part at right end.
- Bubble sort is comparison based algorithm in which each pair of adjacent elements is compared and elements are swapped if they are not in order.

VTC Academy
Think Ahead, Beyond Boundaries