

Unit Six Lecture Three:

Topic 1: An Array Search

During an array search, you scan the array in an attempt to find specific data.

Topic 2: The Sequential (Linear) Search

In a sequential search, you check each element of the array, one at a time starting with index 0 and continuing until you find a match or you reach the end of the array. This is the only way to search an unsorted array.

The Big O notation for a sequential search is $O(n)$, because the maximum number of visits to an array of size n would be n , if the target is the last element.

Below you will find the code for a sequential search method. This method is sent the number to find (target) and returns the number of visits to the array. A -1 will be returned if the target is not found. The array 'a' is an instance variable of the class of which this method is a part.

```
public int Sequential(int target)
{
    for (int i=0; i<a.length; i++)
    {
        if (a[i] == target)
            return i+1;
    }
    return -1;
}
```

Topic 3: The Binary Search

In a binary search, you continue to cut the array in half until you find the target. This search will only work in a sorted array.

The Big O notation for a binary search is $O(\log(n))$. This means that for an array of size 20, the maximum number of visits to the array should be $\log_2 20 = 4.32$ or 5 visits.

For information on how this Big O notation was determined, read pages 725 & 726 in your text.

Below you will find the code for a binary search method. This method is sent the number to find (target) and returns the number of visits to the array. A -1 will be returned if the target is not found. The array 'a' is an instance variable of the class of which this method is a part. This method assumes the array is in ascending order.

```
public int Binary(int target)
{
    int low = 0;
    int high = a.length - 1;
    int mid, dif, count = 0;

    while (low <= high)
    {
        count++;
        mid = (low + high)/2;
        dif = a[mid] - target;

        if (dif == 0)
            return count;
        else if (dif < 0)
            low = mid + 1;
        else
            high = mid - 1;
    }
    return -1;
}
```

Unit Six Assignment Three: Comparing the Sequential & Binary Searches