FIX BUGS OF THE APPLICATION

* **GITHUB link :** [FIX BUGS OF THE APPLICATION](https://github.com/Vamshi2169/JavaFSD/tree/main/Fix%20Bugs%20of%20the%20Application%20project)

1.First, I will fix the bugs for searching Algorithm. I will use BINARY SEARCH Algorithm.

Binary Search is a searching algorithm used in a sorted array by repeatedly dividing the search interval in half. The idea of binary search is to use the information that the array is sorted and reduce the time complexity to O (Log n).

Begin with the mid element of the whole array as a search key.

If the value of the search key is equal to the item, then return an index of the search key.

Or if the value of the search key is less than the item in the middle of the interval, narrow the interval to the lower half.

Otherwise, narrow it to the upper half.

Repeatedly check from the second point until the value is found or the interval is empty.

PSEUDO CODE:

binarySearch(arrayList, x, low, high) // x is the key that we are searching

repeat till low = high

mid = (low + high)/2

if (x == arrayList.get(mid))

return mid

else if (x > arrayList.get(mid)) // x is on the right side

low = mid + 1

else // x is on the left side

high = mid – 1

2.Second we do sort the ArrayList by using the Insertion Sort Algorithm

The arrayList is virtually split into a sorted and an unsorted part. Values from the unsorted part are picked and placed at the correct position in the sorted part.

To sort an array of size N in ascending order:

Iterate from arr[1] to arr[N] over the array.

Compare the current element (key) to its predecessor

If the key element is smaller than its predecessor, compare it to the elements before. Move the greater elements one position up to make space for the swapped element.

PSEUDO CODE:

for(int i=1;i<arrayList.size();i++)

int temp=arrayList.get(i)

int j=i-1

while(j>=0 && temp<arrayList.get(j)){

arrayList.set(j+1,arrayList.get(j)

j=j-1

arrayList.set(j+1,temp)