I have completed this assignment individually, without support from anyone else. I hereby accept that only the below listed sources are approved to be used during this assignment:

(i) course textbook

(ii) all material that is made available to me by the professor

(iii) notes taken by me during lectures

I have not used, accessed or taken any unpermitted information from any other source. Hence; all effort belongs to me.

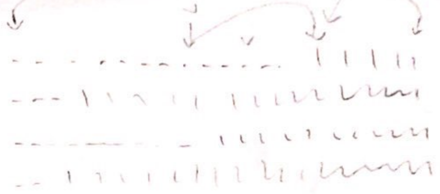
Emir Fatih Ayyıldız

Comp 202 assignment 1

$$\begin{bmatrix} a & a & a & b \\ a & a & a & a \\ a & b & b & b \\ a & a & b & b \end{bmatrix}$$

n by n matrix

count total number of "a"



use binary search to get result in $O(n(\underbrace{\log n}_{\text{1 row search time}}))$ time

colsize

row number

divide in half (logaritmic)

to find first b and last a, mid must be a and mid+1 must be b

as we are using binary search, search times is log(n) for 1row, when n is array size. as there are n rows, search time is (n log(n))

```
{ n= arr. size
    for (n)
        left = 0
        right = arr.length-1

  result = int(left, right, arr, n)
}

static int (left, right, arr, n){
        mid = (right + left)/2
        if (mid = a and mid+1 = b)
            return mid+1
        elif (mid = b)
            right = mid
            int (left, right, arr, n)
        else
            left = mid +1
            int (left, right, arr, n)
}
```

recursion is used

Space complexity of my method and acount method can be calculated.

acount has 1 integer

3 integers in loop

1 helper method in loop

$(1.4 + 3.4.n + n(2.4.\log(n)))$

Helper method has 2 integers and it is initialized log(n) times in the worst scenario

$= 8 + 12n + 8n.n\log(n)$

$= 8 + 20n.n\log(n)$

$= 20n^2\log(n) + 8$

space is occupied