EXPERIMENT 4

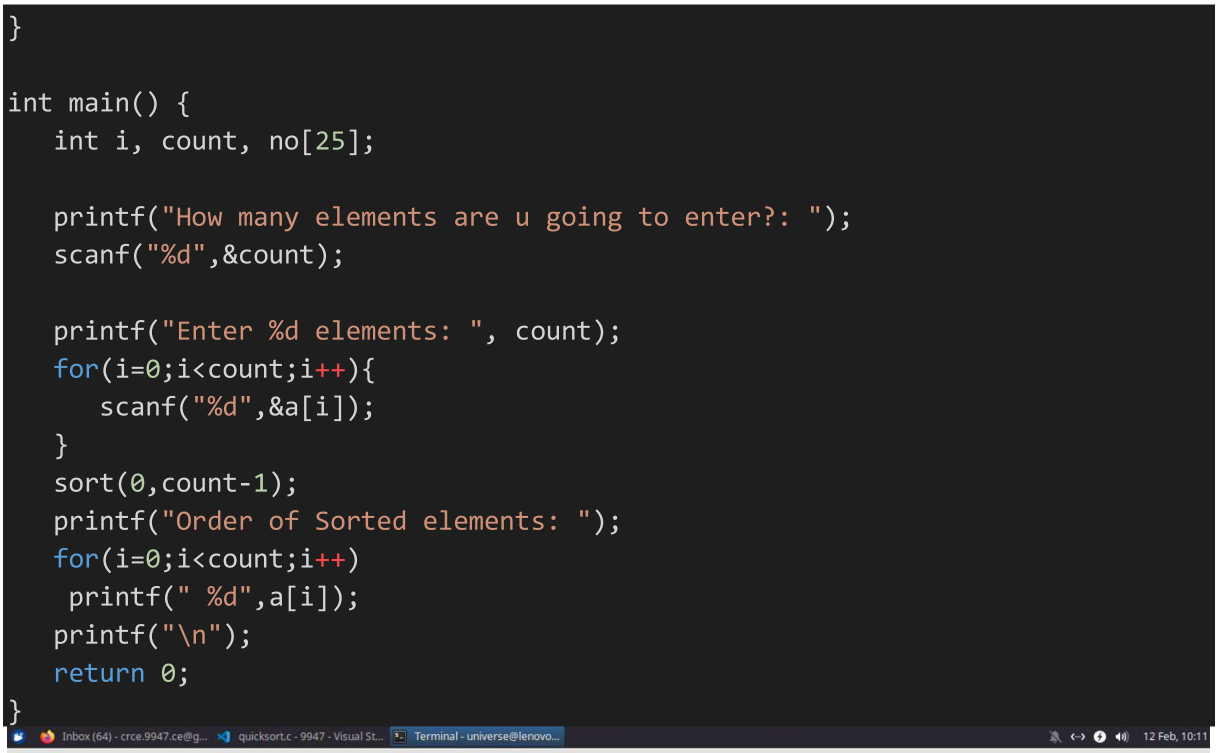
PRAKASH P. BISWAS

9947

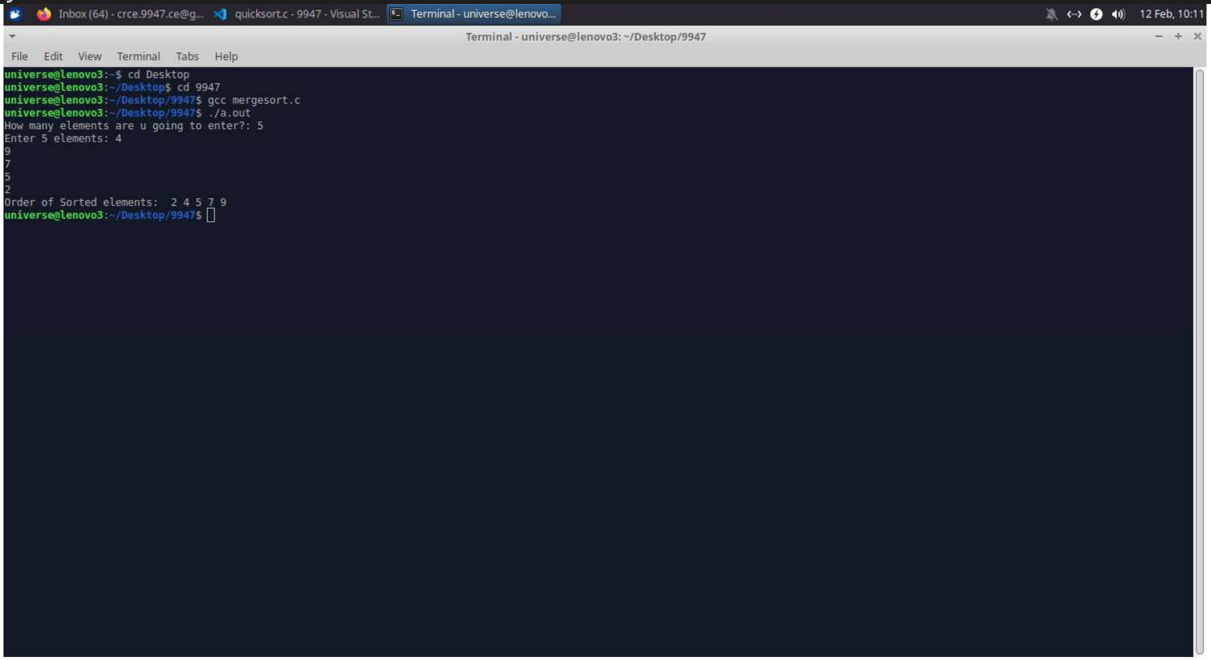
COMPS-B

MERGE SORT:

```
#include <stdio.h>

#define max 10

int a[11];
int b[10];


void merging(int low, int mid, int high) {
   int l1, l2, i;

   for(l1 = low, l2 = mid + 1, i = low; l1 <= mid && l2 <= high; i++) {
      if(a[l1] <= a[l2])
         b[i] = a[l1++];
      else
         b[i] = a[l2++];
   }

   while(l1 <= mid)
      b[i++] = a[l1++];

   while(l2 <= high)
      b[i++] = a[l2++];

   for(i = low; i <= high; i++)
      a[i] = b[i];
}

void sort(int low, int high) {
   int mid;

   if(low < high) {
      mid = (low + high) / 2;
      sort(low, mid);
      sort(mid+1, high);
      merging(low, mid, high);
   } else {
      return;
   }
```

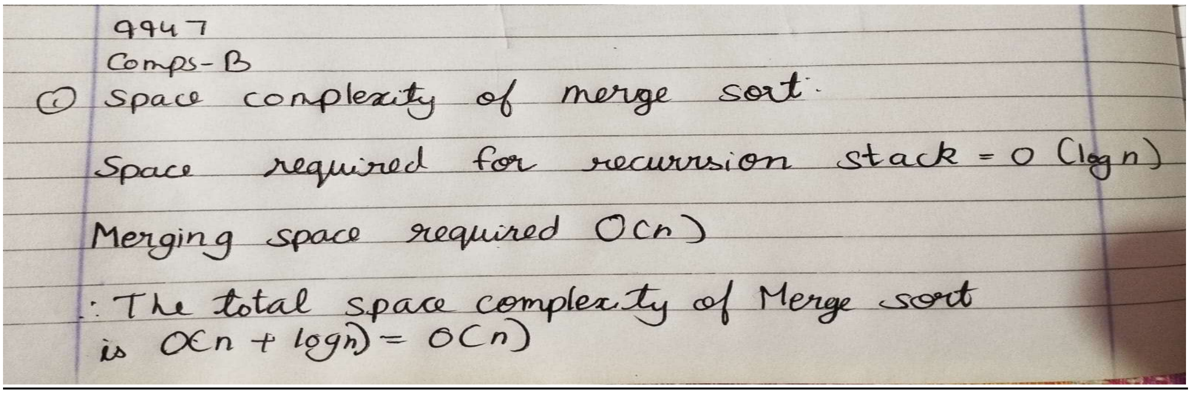


```
}

int main() {
   int i, count, no[25];

   printf("How many elements are u going to enter?: ");
   scanf("%d",&count);

   printf("Enter %d elements: ", count);
   for(i=0;i<count;i++){
      scanf("%d",&a[i]);
   }
   sort(0,count-1);
   printf("Order of Sorted elements: ");
   for(i=0;i<count;i++)
    printf(" %d",a[i]);
   printf("\n");
   return 0;
}
```


Inbox (64) - crce.9947.ce@g...
quicksort.c - 9947 - Visual St...
Terminal - universe@lenovo...
12 Feb, 10:11
Terminal - universe@lenovo3: ~/Desktop/9947
File
Edit
View
Terminal
Tabs
Help

```
universe@lenovo3:~$ cd Desktop
universe@lenovo3:~/Desktop$ cd 9947
universe@lenovo3:~/Desktop/9947$ gcc mergesort.c
universe@lenovo3:~/Desktop/9947$ ./a.out
How many elements are u going to enter?: 5
Enter 5 elements: 4
9
7
5
2
Order of Sorted elements:  2 4 5 7 9
universe@lenovo3:~/Desktop/9947$ 
```

POSTLAB:



9947
Comps-B
① Space conplexity of merge sort.

Space required for recursion stack = O (logn)

Merging space required O(n)

∴ The total space complexity of Merge sort is O(n + logn) = O(n)