Date
Page

LAB-5

① Selection Sort :-

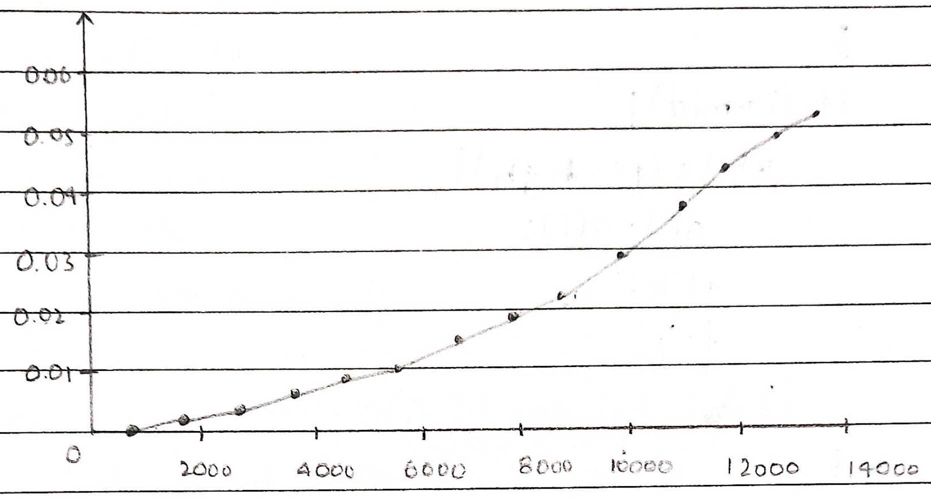
```
#include <stdio.h>
#include < time.h >
#include<stdlib.h>
void selsort (int n, int a[]);

void main(){
    int a[15000], n, i, j, ch, temp;
    clock_t start, end;
    while(1){
          printf("Enter your choice");
          scanf("%d",&ch);
          switch (ch){
          Case 1: printf("\n Enter no. of elecment:");
                scanf("%d", &n);
                printf("Enter array elements");
                for (i=0; i<n ;i++)
                        scanf("%d", &a[i]);
                start= clock();
                selsort (n,a);
                end =clock();
                printf("\n sorted array is:");
                for (i=0 ; i<n;i++)
                      printf("%d\t", a [i]);
                      printf("\n Time taken to sort %d no. is %.
                          secs", n, (((double)(end-start))/ clroacks_perse
                break;
          Case 2:
                n= 500;
                while(n<=14500){
```

```
            for(i=0; i<n; i++){
                a[i] = n-i;
            }.
            start=cloak();
            selsort(n,a);
            for (j=0; j<500000;j++){temp = 38/600;}
            end=clock();
            printf("\n Time taken to sort %d numbers is
                %f Secs",n,(((double)(end-start))/Cloaks_per-sec));
            n =n+1000;
            }. }
            break;
        case3: exit(0);
        }
        getchar();
        }
}
```

O/p:-

```
1: For manual Entry of N. and array elements.
2: To display time taken for sorting no. of elements
3. To exit.
Enter choice:1
Enter no. of elements: 5
Enter array elements: 45   53   33   87   16
Sorted array is: 16   33   45   53   87
Enter choice: 2.
Time taken to sort 500 no. is 0 sec.
                    1500          0.001 sec
                    2500          0.002 sec
                    3500          0.004 sec
                    4500          0.007 sec
```

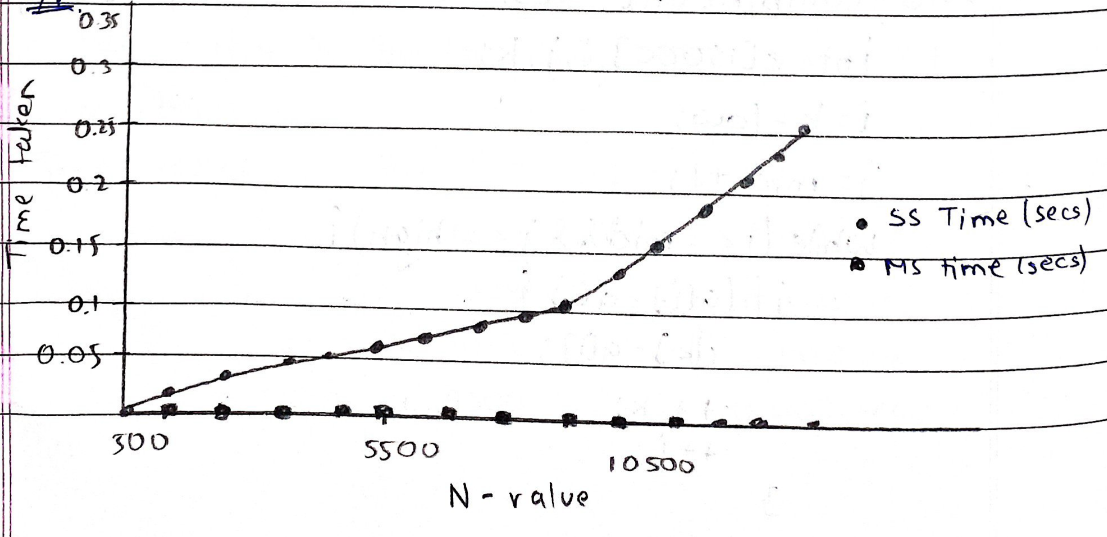
5500	0.010 sec.
6500	0.015 sec.
7500	0.019 sec.
9500	0.024 sec.



② Merge Sort

```
void split(int a[], int low, int high){
        int mid;
        if (low<high){
            mid =(low+ high)/2;
            split(a,low,mid);
            split(a,mid +1, high);
            combine(a,low,mid,high);
        }
}
void combine(int a[], int low, int mid, int high){
        int c[15000],i,j,k;
        i=k=low;
        j= mid +1;
        while (i< =mid&&j<= high){
            if (a[i]<a[i] {
                c[k]=a[i];
                ++ k;
                ++i;
            }
        else{
            c[k]=a[j];
            ++k;
            ++j;
        }
    }
    if (i>mid){
        while(j<=high){
            c[k]=a[j];
            ++k;
            ++j;
        }
    }
    if(j>high){
        while(i<=mid){
            c[k]=a[i];
            ++k;
            ++i;
        }
    }
    for(i=low;i<=high;i++){
        a[i]=c[i];
    }
}
```

O/P:-


Time taken
N - value
● SS Time (secs)
● MS time (secs)