Week - 08

20/06/2024

PAGE NO :
DATE :

Heap Sort :-

```
#include <stdlib.h>
#include <stdio.h>
#include <time.h>

void ~~select~~ HeapSort(int a[], int n);
void heapify (int a[], int n, int i);

void main()
{
   int a[50000], n, i, j, ch, temp;
   clock_t start, end;

   while(1)
   {
      printf("\n1. for manual entry of N \n2. TO display time
            taken for sorting \n3. TO Exit ");
      Printf("\n Enter your choice: ");
      scanf("%d", &ch);
      switch(ch)
      {
         case 1 : printf("\n Enter the number of elements : ");
                  scanf("%d", &n);
                  printf("~~%d~~ Enter array elemnt : ");
                  for(i=0; i<n; i++){
                     scanf("%d", &a[i]);
                  }
                  start = clock();
                  heapSort(a, n);
                  end = clock();
                  printf("\nsorted array is: ");
                  for(i=0; i<n; i++)
                     printf("%d\t", a[i]);
                  Printf("\n Time taken to sort %d numbers
                   is %f secs", n, ((double) (end-start))/
                                                clocks_per_sec));
                  break;
```

```
case 2: n=500;
        while(n<=14500){
        for(i=0;i<n;i++){
            a[i]=n-i;
        }
        start = clock();
        heapSort(a,n);

        for(j=0; j<500000000; j++) { temp=38/600;}
        end=clock();
        heapSort(a,n);
        printf("In Time taken to sort %d numbers is
                %.f secs", n, ((double)(end-start))/
                                CLOCKS_PER_SEC));
            n=n+1000
        }
        break;
    case 3: exit(0);
    }
    getchar();
}
}

void heapify(int arr[], int n, int i){
    int temp, maximum, left_index, right_index;
    maximum = i;
    right_index = 2*i+2;
    left_index = 2*i+1;

    if(left_index<n && arr[left_index]>arr
                        [maximum])
        maximum = left_index;
    if(right_index<n && arr[right_index]>arr[maximum])
        maximum = right_index;
```

```
    if(maximum != i){
        temp = arr[i];
        arr[i] = arr[maximum];
        arr[maximum] = temp;
        heapify(arr, n, maximum);
    }
}

void heapSort(int arr[], int n){
    int i, temp;
    for(i=n/2-1; i>=0; i--){
        heapify(arr, n, i);
    }
    for(i=n-1; i>0; i--){
        temp = arr[0];
        arr[0] = arr[i];
        arr[i] = temp;
        heapify(arr, i, 0);
    }
}
```

Output :-

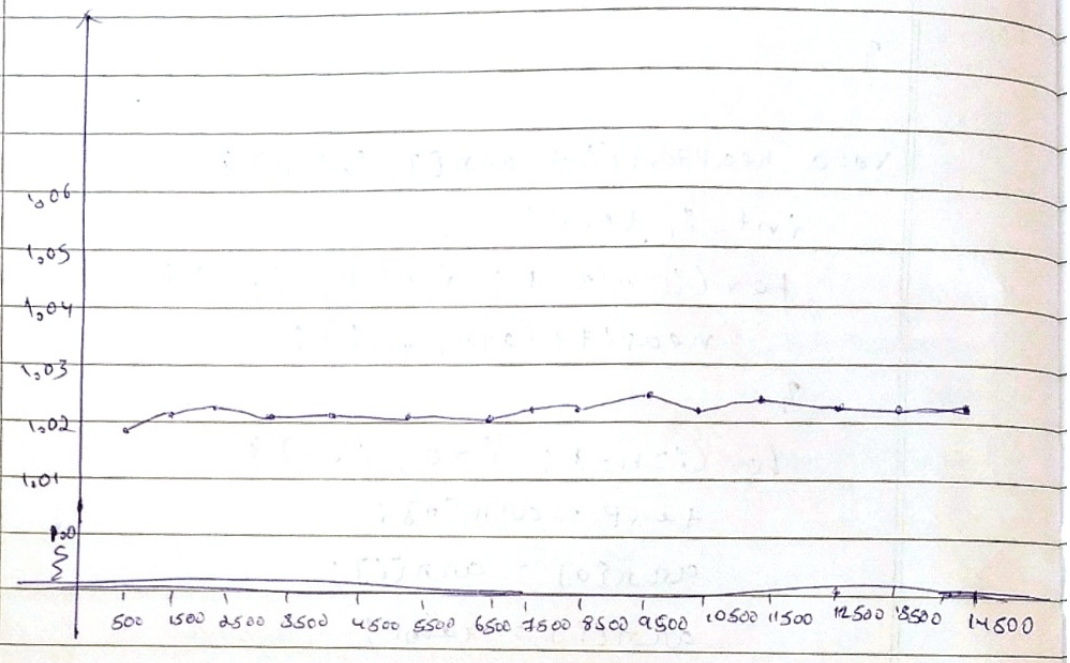
Time taken to sort 500 num is 1.019875 secs

1500		1.020781
2500		1.022064
3500		1.021494
4500		1.021169
5500		1.021648
6500		1.021886
7500		1.022838
8500		1.022405
9500		1.025502
10500		1.025285
11500		1.025886

20/6/24

PAGE NO :
DATE :

Time taken for sort 12500 number is 1.022814 secs
" 13500 " 1.024106 "
" 14500 " 1.025186 "



PAGE NO :
DATE :

```
#include <stdio.h>
#include <stdlib.h>

int cost[1000][1000];

void floyd(int n){
  int d[n][n];
  for(int i=0; i<n; i++){
    for(int j=0; j<n; j++){
      d[i][j] = cost[i][j];
    }
  }
  for(int k=0; k<n; k++){
    for(int i=0; i<n; i++){
      for(int j=0; j<n; j++){
        if(d[i][j] > d[i][k] + d[k][j]){
          d[i][j] = d[i][k] + d[k][j];
        }
      }
    }
  }
  printf("output : \n");
  for(int i=0; i<n; i++){
    for(int j=0; j<n; j++){
      printf("%d", d[i][j]);
    }
    printf("\n");
  }
}
```

```
int main() {
    int n;
    printf("Enter number of elements: ");
    scanf("%d", &n);
    printf("Enter elements: \n");
    for(int i=0; i<n; i++) {
        for(int j=0; j<n; j++) {
            scanf("%d", &cost[i][j]);
        }
    }
    floyd(n);
    return 0;
}
```

Output :-

Enter number of elements : 4

Enter elements :

0	999	3	999
2	0	999	999
999	7	0	1
6	999	999	0

output :

0	10	3	4
2	0	5	6
7	7	0	1
6	16	9	0