

LEENA K CH.SC.U4CSE24122 CSE-B DAA LAB WORK

DESIGN ANALYSIS AND ALGORITHM-LAB WORK

1)Write a program for Bubble Sorting

Code:

```
1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 int main() {
4     int n, i, j, temp;
5     int arr[50];
6     printf("CH.SC.U4CSE24122\n");
7     printf("Enter number of elements: ");
8     scanf("%d", &n);
9     for(i = 0; i < n; i++){
10         scanf("%d", &arr[i]);
11     }
12     for(i = 0; i < n; i++){
13         for(j = 0; j < n - 1 - i; j++){
14             if(arr[j] > arr[j+1]){
15                 temp = arr[j];
16                 arr[j] = arr[j+1];
17                 arr[j+1] = temp;
18             }
19         }
20     }
21     printf("Sorted array: ");
22     for(i = 0; i < n; i++){
23         printf("%d ", arr[i]);
24     }
25     return 0;
26 }
```

Output:

```
CH.SC.U4CSE24122
Enter number of elements: 5
10
30
50
60
35
Sorted array: 10 30 35 50 60
-----
Process exited after 12.39 seconds with return value 0
Press any key to continue . . .
```

2)Write a program for Insertion Sorting

Code:

```

1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 int main() {
4     int n, i, j, key;
5     int arr[50];
6     printf("CH.SC.U4CSE24122\n");
7     printf("Enter number of elements: ");
8     scanf("%d", &n);
9     for(i = 0; i < n; i++){
10         scanf("%d", &arr[i]);
11     }
12     for(i = 1; i < n; i++){
13         key = arr[i];
14         j = i - 1;
15         while(j >= 0 && arr[j] > key){
16             arr[j+1] = arr[j];
17             j--;
18         }
19         arr[j+1] = key;
20     }
21     printf("Sorted array: ");
22     for(i = 0; i < n; i++){
23         printf("%d ", arr[i]);
24     }
25     return 0;
26 }
27

```

Output:

```
C:\Users\kpriy\Downloads\iut  X  +  v
CH.SC.U4CSE24122
Enter number of elements: 6
1
5
3
2
6
10
Sorted array: 1 2 3 5 6 10
-----
Process exited after 8.629 seconds with return value 0
Press any key to continue . . .
```

3)Write a program for Selection Sorting

Code:

```

1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 int main() {
4     int n, i, j, min, temp;
5     int arr[50];
6     printf("CH.SC.U4CSE24122\n");
7     printf("Enter number of elements: ");
8     scanf("%d", &n);
9     for(i = 0; i < n; i++){
10         scanf("%d", &arr[i]);
11     }
12     for(i = 0; i < n; i++){
13         min = i;
14         for(j = i + 1; j < n; j++){
15             if(arr[j] < arr[min]){
16                 min = j;
17             }
18         }
19         temp = arr[i];
20         arr[i] = arr[min];
21         arr[min] = temp;
22     }
23     printf("Sorted array: ");
24     for(i = 0; i < n; i++){
25         printf("%d ", arr[i]);
26     }
27     return 0;
28 }
29

```

Output:

```

CH.SC.U4CSE24122
Enter number of elements: 7
10
20
35
65
21
34
65
Sorted array: 10 20 21 34 35 65 65
-----
Process exited after 14.4 seconds with return value 0
Press any key to continue . . . |

```

4)Write a program for Bucket Sorting

Code:

```
1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 int main() {
4     int n, i, num;
5     int bucket[51];
6     printf("CH.SC.U4CSE24122\n");
7     printf("Enter number of elements: ");
8     scanf("%d", &n);
9
10    for(i = 0; i < 51; i++){
11        bucket[i] = 0;
12    }
13    printf("Enter elements (0 to 50):\n");
14    for(i = 0; i < n; i++){
15        scanf("%d", &num);
16        bucket[num]++;
17    }
18    printf("Sorted array: ");
19    for(i = 0; i < 51; i++){
20        while(bucket[i] > 0){
21            printf("%d ", i);
22            bucket[i]--;
23        }
24    }
25    return 0;
26 }
```

Output:

```

CH.SC.U4CSE24122
Enter number of elements: 3
Enter elements (0 to 50):
49
1
23
Sorted array: 1 23 49
-----
Process exited after 19.14 seconds with return value 0
Press any key to continue . . . |

```

5) Write a program for Heap Sorting

Code:

```

1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 void heapify(int arr[], int n, int i){
4     int largest = i;
5     int left = 2*i + 1;
6     int right = 2*i + 2;
7     int temp;
8     if(left < n && arr[left] > arr[largest])
9         largest = left;
10    if(right < n && arr[right] > arr[largest])
11        largest = right;
12    if(largest != i){
13        temp = arr[i];
14        arr[i] = arr[largest];
15        arr[largest] = temp;
16        heapify(arr, n, largest);
17    }
18
19 int main() {
20     int n, i, temp;
21     int arr[50];
22     printf("CH.SC.U4CSE24122\n");
23     printf("Enter number of elements: ");
24     scanf("%d", &n);
25     for(i = 0; i < n; i++){
26         scanf("%d", &arr[i]);

```

```

27 | }
28 | for(i = n/2 - 1; i >= 0; i--){
29 |     heapify(arr, n, i);
30 | }
31 | for(i = n - 1; i > 0; i--){
32 |     temp = arr[0];
33 |     arr[0] = arr[i];
34 |     arr[i] = temp;
35 |
36 |     heapify(arr, i, 0);
37 | }
38 | printf("Sorted array: ");
39 | for(i = 0; i < n; i++){
40 |     printf("%d ", arr[i]);
41 | }
42 | return 0;
43 | }
44 |

```

Output:

```

CH.SC.U4CSE24122
Enter number of elements: 4
12
32
14
54
Sorted array: 12 14 32 54
-----
Process exited after 6.202 seconds with return value 0
Press any key to continue . . .

```

6)Write a program for BFS

Code:


```

1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 int main() {
4     int n, i, j, start;
5     int graph[10][10], visited[10], queue[20];
6     int front = 0, rear = 0;
7     printf("CH.SC.U4CSE24122\n");
8     printf("Enter number of nodes: ");
9     scanf("%d", &n);
10    printf("Enter adjacency matrix:\n");
11    for(i = 0; i < n; i++){
12        for(j = 0; j < n; j++){
13            scanf("%d", &graph[i][j]);
14        }
15    }
16    for(i = 0; i < n; i++){
17        visited[i] = 0;
18    }
19    printf("Enter starting node: ");
20    scanf("%d", &start);
21    queue[rear++] = start;
22    visited[start] = 1;
23    printf("BFS: ");
24    while(front < rear){
25        int node = queue[front++];
26        printf("%d ", node);
27        for(i = 0; i < n; i++){
28            if(graph[node][i] == 1 && visited[i] == 0){
29                queue[rear++] = i;
30                visited[i] = 1;
31            }
32        }
33    }
34    return 0;
35 }
36

```

Output:

CH.SC.U4CSE24122

Enter number of nodes: 4

Enter adjacency matrix:

0 1 1 0

1 0 1 1

1 1 0 0

0 1 0 0

Enter starting node: 0

BFS: 0 1 2 3

Process exited after 51.81 seconds with return value 0

Press any key to continue . . . |

7)Write a program for DFS

Code:

```
1 //CH.SC.U4CSE24122
2 #include <stdio.h>
3 void dfs(int node, int graph[10][10], int visited[10], int n){
4     int i;
5     visited[node] = 1;
6     printf("%d ", node);
7
8     for(i = 0; i < n; i++){
9         if(graph[node][i] == 1 && visited[i] == 0){
10             dfs(i, graph, visited, n);
11         }
12     }
13 }
14 int main() {
15     int graph[10][10], visited[10];
16     int n, start, i, j;
17     printf("CH.SC.U4CSE24122\n");
18     printf("Enter number of nodes: ");
19     scanf("%d", &n);
20     printf("Enter adjacency matrix:\n");
21     for(i = 0; i < n; i++){
22         for(j = 0; j < n; j++){
23             scanf("%d", &graph[i][j]);
24         }
25     }
```

```

26  for(i = 0; i < n; i++){
27      visited[i] = 0;
28  }
29  printf("Enter starting node: ");
30  scanf("%d", &start);
31
32  printf("DFS: ");
33  dfs(start, graph, visited, n);
34  return 0;
35  }
36

```

Output:

```

CH.SC.U4CSE24122
Enter number of nodes: 4
Enter adjacency matrix:
0 1 1 0
1 1 0 0
0 0 1 0
1 0 0 0
Enter starting node: 0
DFS: 0 1 2
-----
Process exited after 30.23 seconds with return value 0
Press any key to continue . . . |

```