main.c

```c
#include<stdio.h>
#include <time.h>
void insertq(int q[],int node, int *f, int *r)
{
    if((*f==-1) && (*r==-1))
    {
     (*f)++, (*r)++, q[*f]=node;
    }
    else
    {
     (*r)++, q[*r]=node;
     }
}

int deleteq(int q[],int *f,int *r)
{
     int temp;
     temp=q[*f];
     if(*f == *r) *f=*r=-1;
     else (*f)++;
     return temp;
}

void bfs(int n, int adj[][10],int src, int visited[])
{
     int q[20], f=-1,r=-1,v,i;
     insertq(q,src,&f,&r);
     while((f <=r ) && (f != -1))
     {
          v=deleteq(q,&f,&r);
```

main.c

```
30              v=deleteq(q,&f,&r);
31              if(visited[v]!=1)
32              {
33                  visited[v]=1;
34                  printf("%d",v);
35              }
36              for(i=1;i<=n;i++)
37                  if((adj[v][i]==1) && (visited[i] !=1))
38                  insertq(q,i,&f,&r);
39          }
40      }
41
42      int main()
43      {
44          int n,i,j,adj[10][10],src,visited[10];
45          clock_t start, end;
46          double t;
47          printf("Enter number of vertices:\n");
48          scanf("%d",&n);
49          printf("Enter adjacency matrix:\n");
50          for(i=1;i<=n;i++)
51          {
52              visited[i]=0;
53            for(j=1;j<=n;j++)
54              scanf("%d",&adj[i][j]);
55          }
56          printf("Enter starting vertex:\n");
57          scanf("%d",&src);
58          printf("The nodes reachable from source are : ");
59          bfs(n,adj,src,visited);
```

input

```
41
42  int main()
43  {
44      int n,i,j,adj[10][10],src,visited[10];
45      clock_t start, end;
46      double t;
47      printf("Enter number of vertices:\n");
48      scanf("%d",&n);
49      printf("Enter adjacency matrix:\n");
50      for(i=1;i<=n;i++)
51      {
52          visited[i]=0;
53        for(j=1;j<=n;j++)
54          scanf("%d",&adj[i][j]);
55      }
56      printf("Enter starting vertex:\n");
57      scanf("%d",&src);
58      printf("The nodes reachable from source are : ");
59      bfs(n,adj,src,visited);
60      start = clock();
61      bfs(n,adj,src,visited);
62      end = clock();
63      t = ((double) (end - start)) / CLOCKS_PER_SEC;
64      printf("\n");
65      printf("\nTime taken by BFS : %lf\n", t);
66      printf("\n");
67      return 0;
68
69  }
70
```

```
Enter number of vertices:
3
Enter adjacency matrix:
0 1 1
1 0 1
1 1 0
Enter starting vertex:
1
The nodes reachable from source are : 123

Time taken by BFS : 0.000002


...Program finished with exit code 0
Press ENTER to exit console.
```