Assignment No.7

Name: **Dnyaneshwari Thombal**
Class: **SY-1**
Batch: **C**
PRN: **B25CE2005**

Problem Statement:
Web Crawling:

A. Breadth First Search (BFS): Application: Indexing web pages for search engines. Example: A web crawler uses BFS to visit web pages systematically, starting from a seed URL and exploring links level by level. Nodes represent web pages. Edges represent hyperlinks. BFS ensures that pages at the same "depth" (distance from the starting page) are visited before moving to deeper levels. Write a program to simulate the indexing of web pages for a search engine using a Breadth-First Search (BFS) algorithm.

Program:

```cpp
#include<iostream>
#define MAX 10

int queue[MAX], front = -1, rear = -1; int
visited[MAX] = {0}; int adj[MAX][MAX]; //
adjacency matrix

void enqueue(int v) { if (rear == MAX - 1)
    printf("Queue Overflow\n");
    else { if (front == -1) front
        = 0; queue[++rear] =
        v;
    } }  int dequeue() { if (front == -1 ||
front >
rear)  return -1;
    return queue[front++];
} void bfs(int start, int n) {

    enqueue(start); visited[start] = 1;  printf("\nIndexing
    pages starting from: %d\n", start);

    while (front <= rear) {  int
        v = dequeue();
        printf("Indexed page:
        %d\n", v);

        for (int i = 0; i < n; i++) {  if
            (adj[v][i] == 1 && !visited[i]) {
            enqueue(i); visited[i] = 1;
```

```
        }
      }
    }
} int main() { int n, e, from, to, start;
    printf("Enter number of web pages (max 10): "); scanf("%d", &n);

    printf("\nEnter number of hyperlinks: ");
    scanf("%d", &e);
    // initialize adjacency matrix for (int i
    = 0; i < n; i++) for (int j = 0; j < n; j++)
    adj[i][j] = 0;
    printf("\nEnter links (from to):\n"); for (int i = 0; i
    < e; i++) {  scanf("%d %d", &from, &to);
        adj[from][to] = 1;
    }  printf("\nEnter seed/start page number: ");
    scanf("%d",
     &start); bfs(start, n);
    return 0;
}
```

Output:

B. Depth First Search (DFS): Application: Web crawlers use DFS to explore web pages systematically, following links and indexing content for search engines. Write a simple

program to index web pages using Depth First Search (DFS). The program should simulate a web graph where pages are represented as nodes and hyperlinks as edges.

Program

```c
#include     <stdio.h>     #define     MAX  10     int
adj[MAX][MAX]; // adjacency matrix int visited[MAX];
// visited array  void dfs(int v, int n)
{ visited[v] = 1;  printf("\n Indexed
page: %d", v);

    for (int i = 0; i < n; i++) { if (adj[v][i] == 1 &&
        !visited[i]) { dfs(i, n);
        }
    }
}

int main() { int n, e, from, to, start;

    printf("Enter number of web pages (max 10): "); scanf("%d",
    &n); printf("\nEnter number of hyperlinks: ");
    scanf("%d", &e);
    // Initialize adjacency matrix for (int i
    = 0; i < n; i++) for (int j = 0; j < n; j++)
    adj[i][j] = 0;

    printf("\nEnter links (from to):\n");
    for (int i = 0; i < e; i++) {
    scanf("%d %d", &from, &to);
    adj[from][to] = 1;
    }     printf("\nEnter  seed  (start)  page  number:  "); scanf("%d",  &start);
printf("\n\nStarting DFS web crawling from page: %d\n", start); dfs(start, n); return 0;  }
```

Output: