로고, 등록 상표, 폰트, 상징이(가) 표시된 사진

자동 생성된 설명

**데이터구조2**

**과제#1-2 maxheap**

|  |  |
| --- | --- |
| 과목명 | 데이터구조2 |
| 담당교수 | 남윤영 |
| 학과 | 컴퓨터공학과 |
| 학년 | 2 |
| 학번 | 20204101 |
| 이름 | 신동욱 |
| 제출일 | 09.19 |

**1. 소스코드**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define MAX\_VERTICES 50

#define TRUE 1

#define FALSE 0

int visted[MAX\_VERTICES];

typedef struct {

    int n;

    int adj\_mat[MAX\_VERTICES][MAX\_VERTICES];

} GraphType;

void init(GraphType \*g) {

    int r, c;

    g->n = 0;

    for (r = 0; r < MAX\_VERTICES; r++) {

        for (c = 0; c < MAX\_VERTICES; c++) {

            g->adj\_mat[r][c] = 0;

        }

    }

}

void insert\_vertex(GraphType\* g, int v) {

    if ((g->n) > MAX\_VERTICES) {

        fprintf(stderr, "정점 갯수 초과");

        return;

    }

    g->n++;

}

int insert\_edge(GraphType\* g, int start, int end) {

    if (start == end) {

        return FALSE;

    }

    if (start >= g->n || end >= g->n) {

        fprintf(stderr, "Error\n");

        return FALSE;

    }

    if (g->adj\_mat[start][end] == 1 || g->adj\_mat[end][start] == 1) {

        return FALSE;

    }

    g->adj\_mat[start][end] = 1;

    g->adj\_mat[end][start] = 1;

    return TRUE;

}

void print\_adj\_mat(GraphType\* g) {

    char a = 'A';

    for (int i = 0; i < g->n; i++) {

        printf("%c  ", a);

        a++;

        for (int j = 0; j < g->n; j++) {

            printf("%2d ", g->adj\_mat[i][j]);

        }

        printf("\n");

    }

}

void dfs\_mat(GraphType\* g, int v) {

    int w;

    char a = 'A';

    a = a + v;

    visted[v] = TRUE;

    printf("%c ", a);

    for (w = 0; w < g->n; w++) {

        if (g->adj\_mat[v][w] && !visted[w])

            dfs\_mat(g, w);

    }

}

int main() {

    GraphType \*g;

    g = (GraphType \*)malloc(sizeof(GraphType));

    init(g);

    srand(time(NULL));

    printf("정점의 개수는? ");

    int ver;

    scanf("%d", &ver);

    for (int i = 0; i < ver; i++) {

        insert\_vertex(g, i);

    }

    printf("랜덤 연결 그래프 생성\n");

    int max\_edge = ver \* (ver - 1) / 2;

    printf("간선 수 (%d 이하): ", max\_edge);

    int edge;

    scanf("%d", &edge);

    int i = 0;

    while (i < edge) {

        int randomX = rand() % ver;

        int randomY = rand() % ver;

        if (!insert\_edge(g, randomX, randomY)) {

            continue;

        }

        i++;

    }

    print\_adj\_mat(g);

    printf("DFS: ");

    dfs\_mat(g, 0);

    printf("\n\n");

    free(g);

    return 0;

}

텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명**2. 실행 결과**

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 타이포그래피이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

다양한 환경에서 테스트 하였습니다.