**로고, 폰트, 텍스트, 상징이(가) 표시된 사진

자동 생성된 설명**

**[데이터구조2 리포트]**

**이름: 황지상 학번: 20224060**

**과제#6 최소비용 신장트리(Prim의 MST)**

1. **코드**

#include <stdio.h>

#include <stdlib.h>

#define TRUE 1

#define FALSE 0

#define MAX\_VERTICES 100

#define INF 1000L

typedef struct GraphType {

int n; // 그래프의 정점 수를 나타내는 변수

int weight[MAX\_VERTICES][MAX\_VERTICES]; // 간선 가중치를 저장하는 2차원 배열

} GraphType;

int selected[MAX\_VERTICES]; // 선택된 정점을 나타내는 배열

int distance[MAX\_VERTICES]; // 시작 정점에서 각 정점까지의 최단 거리를 저장하는 배열

int get\_min\_vertex(int n) {

int v, i;

for (i = 0; i < n; i++) {

if (!selected[i]) { // 아직 선택되지 않은 정점 중에서

v = i; // 첫 번째로 발견된 정점을 선택

break;

}

}

for (i = 0; i < n; i++) {

if (!selected[i] && (distance[i] < distance[v])) // 선택되지 않은 정점 중에서 최소 거리를 가지는 정점을 찾음

v = i;

}

return (v); // 최소 거리를 가지는 정점의 인덱스 반환

}

void print\_selected\_distance(int n) {

int i;

printf("selected[]: ");

for (i = 0; i < n; i++) {

printf("%d ", selected[i]); // 선택된 정점 출력

}

printf("\n");

printf("distance[]: ");

for (i = 0; i < n; i++) {

printf("%d ", distance[i]); // 시작 정점으로부터의 최단 거리 출력

}

printf("\n\n");

}

void prim(GraphType\* g, int s) {

int i, u, v;

int total\_weight = 0; // MST의 간선 가중치의 합을 저장하기 위한 변수

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

distance[u] = INF; // 모든 정점의 거리를 무한대로 초기화

}

distance[s] = 0; // 시작 정점의 거리를 0으로 초기화

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

u = get\_min\_vertex(g->n); // 최소 거리를 가지는 정점을 선택

selected[u] = TRUE; // 해당 정점을 선택함을 표시

if (distance[u] == INF) return; // 선택된 정점이 없으면 종료

printf("정점 %d 추가\n", u); // 현재 선택한 정점을 출력

print\_selected\_distance(g->n); // 선택된 정점과 거리 정보 출력

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

if (g->weight[u][v] != INF) { // 현재 정점과 연결된 정점 중에서

if (!selected[v] && g->weight[u][v] < distance[v]) {

distance[v] = g->weight[u][v]; // 최단 거리 갱신

}

}

}

}

// MST의 간선 가중치의 합 계산

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

if (distance[u] != INF) {

total\_weight += distance[u]; // 선택된 간선의 가중치를 더함

}

}

printf("MST 간선의 가중치 합: %d\n", total\_weight); // MST의 간선 가중치의 합 출력

}

int main(void) {

// 그래프 초기화

GraphType g = { 9,

{{ 0, 35, INF, INF, 245, INF, INF, INF, INF },

{ 35, 0, 123, INF, INF, 154, INF, INF, INF },

{ INF, 123, 0, 117, INF, 165, INF, 222, INF },

{ INF, INF, 117, 0, INF, INF, INF, INF, INF },

{ 245, INF, INF, INF, 0, 87, 98, INF, INF },

{ INF, 154, 165, INF, 87, 0, 122, 155, INF },

{ INF, INF, INF, INF, 98, 122, 0, INF, INF },

{ INF, INF, 222, INF, INF, 155, INF, 0, 164 },

{ INF, INF, INF, INF, INF, INF, INF, 164, 0 }}

};

prim(&g, 1); // prim 알고리즘을 사용하여 MST 생성

return 0;

}

1. **결과**

**텍스트, 스크린샷, 폰트, 메뉴이(가) 표시된 사진

자동 생성된 설명인천 – 0, 서울 – 1, 원주 – 2, 강릉 – 3, 전주 – 4, 대전 – 5, 광주 – 6, 대구 – 7, 부산 – 8**