#자료구조 과제

학번 : 1791258

이름 : 이병헌

소속 : 컴퓨터공학부

과목 : 자료구조

담당 교수 : 권영미

문제 1)

#include <stdio.h>

#include <math.h>

typedef struct student {

int id;

double eng;

double math;

}student;

double EngAvg(student \*a)

{

int i;

int sum = 0;

for (i = 0; i < 10; i++)

sum += a[i].eng;

return sum / 10;

}

double MathAvg(student \*a)

{

int i;

int sum = 0;

for (i = 0; i < 10; i++)

sum += a[i].math;

return sum / 10;

}

//영어 최고점 함수

void EngMaxscore(student \*a) {

int i;

double Max = 0;

int id;

for (i = 0; i < 10; i++) {

if (a[i].eng > Max)

{

Max = a[i].eng;

id = a[i].id;

}

}

printf("영어 최고 점수는 %.2lf 학번은 %d \n", Max, id);

}

//수학 최고점 함수

void MathMaxscore(student \*a) {

int i;

double Max = 0;

int id;

student temp;

for (i = 0; i < 10; i++)

if (a[i].math > Max)

{

Max = a[i].math;

id = a[i].id;

}

printf("수학 최고 점수는 %.2lf 학번은 %d \n", Max, id);

}

//영어 표준편차 구하기

double Engpyojun(student \*a) {

int i;

int avg = 0;

double square, stdDev = 0.0;

for (i = 0; i < 10; i++)

avg += a[i].eng;

avg = avg / 10; // 영어 성적 평균

for (i = 0; i<10; i++)

{

square = (a[i].eng - avg) \* (a[i].eng - avg);

stdDev += (double)(square) / 10;

}

stdDev = sqrt(stdDev);

return stdDev;

}

//수학 표준편차 구하기

double Mathpyojun(student \*a) {

int i;

int avg = 0;

double square, stdDev = 0.0;

for (i = 0; i < 10; i++)

avg += a[i].math;

avg = avg / 10; // 수학 성적 평균

for (i = 0; i<10; i++)

{

square = (a[i].math - avg) \* (a[i].math - avg);

stdDev += (double)(square) / 10;

}

stdDev = sqrt(stdDev);

return stdDev;

}

//총점 최고점, 학번 구하기

void TotalScore(student \*a)

{

int i , id;

int id\_array[10];

double Maxscore = 0;

double totalscore[10];

for (i = 0; i < 10; i++) //총점을 구해서 배열로 저장

{

totalscore[i] = a[i].eng + a[i].math; //영어와 수학 점수 합친 총점

id\_array[i] = a[i].id; //그 총점의 학생 학번

}

for (i = 0; i < 9; i++) //총점끼리 더해서 최대값 찾기

{

if (totalscore[i] > Maxscore)

{

Maxscore = totalscore[i]; //최대값

id = id\_array[i]; //최대값의 학번

}

}

printf("총점 최고점 학생의 학번과 점수는 : %d, %.2lf\n", id, Maxscore);

}

//학번입력으로 찾기

int search(int n, student \*a) {

int i;

for (i = 0; i < 10; i++)

{

if (n == a[i].id)

{

printf("%d 학번 학생의 ", n);

printf("수학 : %.2lf ",a[i].math);

printf("영어 : %.2lf ", a[i].eng);

printf("총점 : %.2lf\n", a[i].math+a[i].eng );

return 0;

}

}

printf("학번을 다시 입력\n");

return 1;

}

void main() {

int student\_num;

student student[10] = { { 1234, 78, 87 },{ 1235,65,55},{1365,88,95},

{2546,45,56},{3124,58,88},{1976,71,54},

{1657,32,55},{2547,90,43},{2456,66,78},{1765,54,99} }; //학생 10명의 데이터 저장

printf("영어의 평균 점수는 : %.2lf\n",EngAvg(student));

printf(" \n");

printf("수학의 평균 점수는 : %.2lf\n", MathAvg(student));

printf(" \n");

EngMaxscore(student);

printf(" \n");

MathMaxscore(student);

printf(" \n");

printf("영어성적의 표준편차는 : %.2lf\n", Engpyojun(student));

printf(" \n");

printf("수학성적의 표준편차는 : %.2lf\n", Mathpyojun(student));

printf(" \n");

TotalScore(student);

printf(" \n");

do

{

printf("학생의 학번을 입력하세요 : ");

scanf("%d", &student\_num);

} while (search(student\_num, student));

}

문제 2

#include <stdio.h>

#include <math.h>

#define MAX(a,b) (((a)>(b))?(a):(b))

#define MAX\_DEGREE 101

typedef struct { //다항식 구조체 타입 선언

int degree; //다항식의 차수

float coef[MAX\_DEGREE]; //다항식의 계수

}polynomial;

//C = A+B여기서 A와 B는 다항식이다.

polynomial poly\_add1(polynomial A, polynomial B) // 두개의 다항식을 더해 새로운 다항식을 만드는 함수

{

polynomial C; //결과 다항식

int Apos = 0, Bpos = 0, Cpos = 0; //배열 인덱스 변수

int degree\_a = A.degree;

int degree\_b = B.degree;

C.degree = MAX(A.degree, B.degree); //결과 다항식 차수 둘의 차수 비교후 큰 차수가 C의 차수가됨

while (Apos <= A.degree && Bpos <= B.degree)

{

if (degree\_a > degree\_b) { //A항 > B항

C.coef[Cpos++] = A.coef[Apos++];

degree\_a--;

}

else if (degree\_a == degree\_b) { //A항 == B항

C.coef[Cpos++] = A.coef[Apos++] + B.coef[Bpos++];

degree\_a--; degree\_b--;

}

else { //B항 > A항

C.coef[Cpos++] = B.coef[Bpos++];

degree\_b--;

}

}

return C;

}

void eval(polynomial A, int n)

{

int i = 0;

int Poly = 0;

for (i = 0; i < A.degree+1; i++)

{

Poly += (A.coef[i] \* pow(n, A.degree - i)); // 미지수가 n일때 다항식의 계산을 하는법

}

printf("다항수에 미지수 %d를 대입한 결과 = %d\n",n ,Poly);

}

void delete(polynomial \*A) //배열 삭제 함수

{

int i,j;

int item;

for (i = 0; i < A->degree; i++)

{

item = A->coef[i];

if (item == 0)

{

for (j = i; j < (A->degree); j++)

A->coef[j] = A->coef[j + 1];

}

}

}

polynomial eraseMaxdegree(polynomial p) //최고차항 계수 없애고 항 내리기

{

int i, k = 0;

printf("다항식의 최대 차수 = %d\n", p.degree);

for (i = 0; i < p.degree + 1; i++)

printf("%.0lf ", p.coef[i]);

printf("\n");

polynomial result = p;

result.coef[0] = 0;

while (result.coef[k] == 0) {

k++;

}

for (i = 0; i<result.degree; i++) {

result.coef[i] = result.coef[k + i];

}

result.degree -= k;

/\*

result.coef[0] = 0; // 최고차항 계수 0으로 만들어서 삭제, 오류 주석처리 보류

for (i = 0; i < result.degree + 1; i++) //

{

if (result.coef[i] == 0)

{

result.degree--;

continue;

}

else if (result.coef[i] != 0)

break;

printf("%.0lf ", result.coef[i]);

}

printf("\n");

delete(&result); //계수가 0인 항 없애기

\*/

printf("다항식의 최대 차수 = %d\n", result.degree);

for (i = 0; i < result.degree + 1; i++)

printf("%.0lf ", result.coef[i]);

return result;

}

//주함수

int main()

{

char C;

int n;

polynomial a = { 5,{3,6,0,0,0,10} };

polynomial b = { 4,{7,0,5,0,1} };

polynomial c;

polynomial p = { 5,{10,0,0,0,6,1} };

c = poly\_add1(a, b);

scanf("%d", &n);

eval(p, n);

printf("\n");

eraseMaxdegree(a);

printf("\n");

eraseMaxdegree(b);

printf("\n");

eraseMaxdegree(p);

return 0;

}

문제3

#include <stdio.h>

#include <stdlib.h>

#define ROWS 3

#define COLS 3

#define MAX\_TERMS 10

typedef struct {

int row; //행

int col; //열

int value; //값

}element;

typedef struct {

element data[MAX\_TERMS];

int rows; //행의 개수

int cols; //열의 개수

int terms; //항의 개수

}SparseMastrix;

//희소 행렬 덧셈 함수

//c = a + b

SparseMastrix sparse\_matrix\_add2(SparseMastrix a, SparseMastrix b)

{

SparseMastrix c;

int ca = 0, cb = 0, cc = 0; //각 배열의 항목을 가리키는 인덱스

//배열 a와 배열 b의 크기가 같은지를 확인

if (a.rows != b.rows || a.cols != b.cols)

{

fprintf(stderr, "희소행렬 크기 에러\n");

exit(1);

}

c.rows = a.rows;

c.cols = a.cols;

c.terms = 0;

while (ca < a.terms && cb < b.terms)

{

//각 항목의 순차적인 번호를 계산한다.

int inda = a.data[ca].row \* a.cols + a.data[ca].col;

int indb = b.data[cb].row \* b.cols + b.data[cb].col;

if (inda < indb)

{

//a 배열 항목이 앞에 있으면

c.data[cc++] = a.data[ca++];

}

else if (inda == indb)

{

//a와 b가 같은 위치

if ((a.data[ca].value + b.data[cb].value) != 0)

{

c.data[cc].row = a.data[ca].row;

c.data[cc].col = a.data[ca].col;

c.data[cc++].value = a.data[ca++].value + b.data[cb++].value;

}

else

{

ca++;

cb++;

}

}

else

// b배열 항목이 앞에 있음

c.data[cc++] = b.data[cb++];

}

//배열 a나 b에 남아있는 항들을 배열c로 옮긴다.

for (; ca < a.terms;)

c.data[cc++] = a.data[ca++];

for (; cb < b.terms;)

c.data[cc++] = b.data[cb++];

c.terms = cc;

return c;

}

SparseMastrix change(SparseMastrix \*m, int i, int j) //i행과 j행 변환하는 함수

{

element temp;

for (int t = 0; t < m->terms; t++)

{

if (m->data[t].row == i) //입력받은 i행이 m.data[t].row 와 같으면 i와 j 변환

m->data[t].row = j;

else if (m->data[t].row == j) //입력받은 j행이 m.data[t].row 와 같으면 j와 i 변환

m->data[t].row = i;

for(int j = 0; j < m->terms; j++) //행 위주로 배열 재 정리? 왜안돼지 (1,1,5) (2,1,9) -> temp,a,b 세개 변수 사용 치환하듯

for (int t = 1; t < m->terms; t++)

if (m->data[t].row < m->data[j].row)

{

temp = m->data[j];

m->data[j] = m->data[t];

m->data[t] = temp;

}

}

}

int main() {

int i, j;

SparseMastrix m1 = { { { 1,1,5 },{ 2,2,9 } },3,3,2 };

SparseMastrix m2 = { { { 0,0,5 },{ 2,2,9 } },3,3,2 };

SparseMastrix m3;

m3 = sparse\_matrix\_add2(m1, m2);

printf("교환할 행을 입력하세요");

while(1)

{

scanf("%d %d", &i, &j);

if ((i < 0 || i>3) || (j < 0 || j>3))

printf("다시 입력하세요");

else

break;

}

printf("\n");

printf("{{{%d,%d,%d},{%d,%d,%d}},%d,%d,%d\n", //기존 SparseMastrix m1

m1.data[0].row, m1.data[0].col, m1.data[0].value,

m1.data[1].row, m1.data[1].col, m1.data[1].value,

m1.rows, m1.cols, m1.terms);

change(&m1, i, j); //포인터로 해야 얘가 아예 데이터 부터 수정됨

printf("{{{%d,%d,%d},{%d,%d,%d}},%d,%d,%d\n", //변경 SparseMastrix m1

m1.data[0].row, m1.data[0].col, m1.data[0].value,

m1.data[1].row, m1.data[1].col, m1.data[1].value,

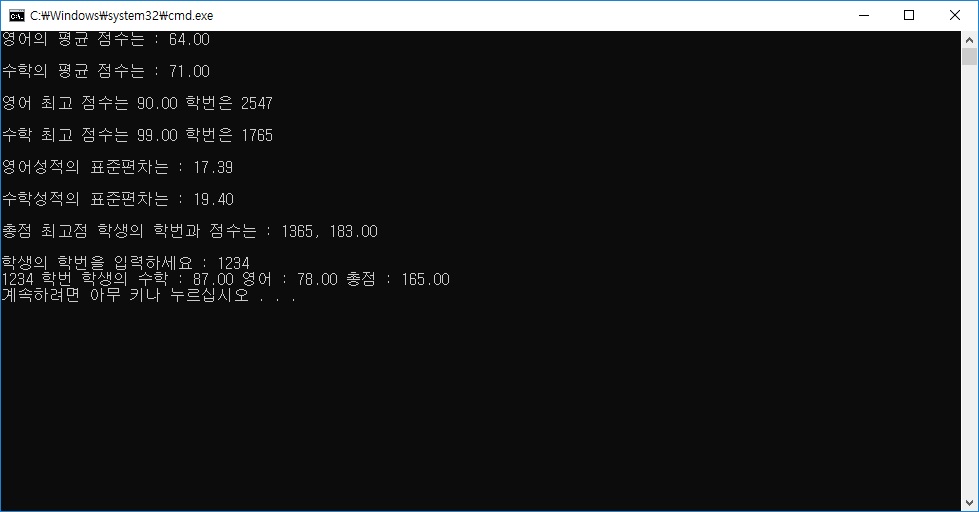
m1.rows, m1.cols, m1.terms);

return 0;

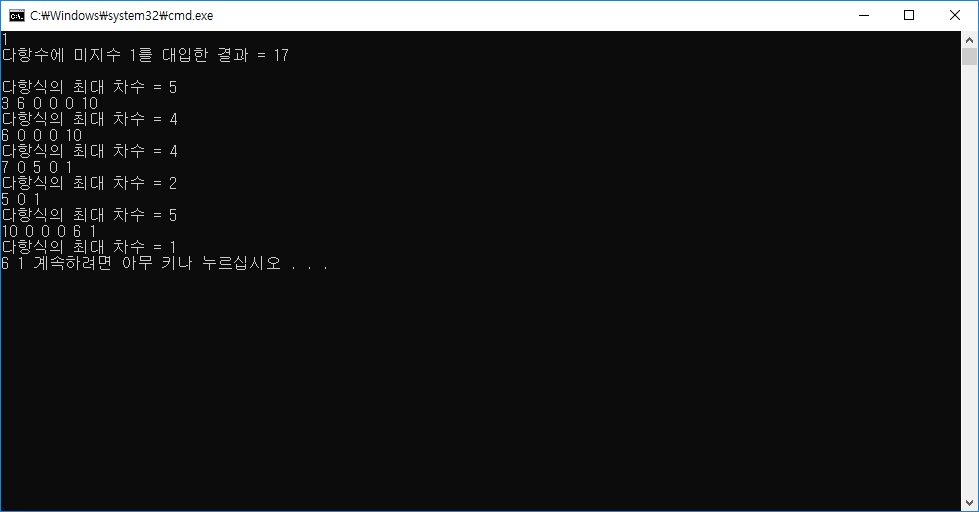
}

실행 화면

문제1)



문제2)



문제3)

