2019202050 이강현 챕터 22 과제제출

실습1

#include <stdio.h>//헤더파일 선언

struct point//point 구조체 정의

{

int xpos;//point 구조체를 구성하는 xpos

int ypos;//point 구조체를 구성하는 ypos

};

int main()//main함수 선언

{

struct point pos1, pos2;//point구조체 변수 선언

double distance;//double형 변수 선언

double circle;//double형 변수 선언

fputs("point1 pos: ", stdout);//출력문

scanf("%d %d", &pos1.xpos, &pos1.ypos);//pos1에 값 입력받기

fputs("point2 pos: ", stdout);//출력문

scanf("%d %d", &pos2.xpos, &pos2.ypos);//pos2에 값 입력받기

distance = sqrt((double)((pos1.xpos - pos2.xpos)\*(pos1.xpos - pos2.xpos) + (pos1.ypos - pos2.ypos)\*(pos1.ypos - pos2.ypos)));//distance 구하는 법 정의

printf("두 점의 거리는 %g 입니다. \n", distance);//출력문

distance = distance / 2;//distance값 대입

circle = distance \* distance\*3.14;//circle값 정의

printf("두 점을 지름으로 하는 원의 넓이 : %g", circle);//circle값 출력

return 0;//종료

}



한줄토의 : 구조체에 대한 정의를 알게 됨.

실습2

#include <stdio.h>//헤더파일 선언

struct person //person 구조체 정의

{

char name[20];//person 구조체를 구성하는 name[20]

char phoneNum[20];//person 구조체를 구성하는 phoneNum[20]

int age;//person 구조체를 구성하는 age

};

int main()//main함수 선언

{

struct person man1, man2;//person 구조체 변수 선언

strcpy(man1.name, "안성준");//man1.name에 안성준 입력받기

strcpy(man1.phoneNum, "010-1122-3344");//man1.phoneNum에 010-1122-3344입력받기

man1.age = 23;//man1.age값 대입

printf("이름 입력: "); scanf("%s", man2.name);//"이름 입력" 출력, man2.name에 이름 입력

printf("번호 입력: "); scanf("%s", man2.phoneNum);//"번호 입력"출력, man2.phoneNum에 번호 입력

printf("나이 입력: "); scanf("%d", &(man2.age));//"나이 입력"출력, man2.age에 나이 입력

printf("이름: %s \n", man1.name);//"이름" 출력, man1.name에 이름 입력

printf("번호: %s \n", man1.phoneNum);//"번호" 출력, man1.phoneNum에 번호 입력

printf("나이: %d \n", man1.age);////"나이" 출력, man1.age에 나이 입력

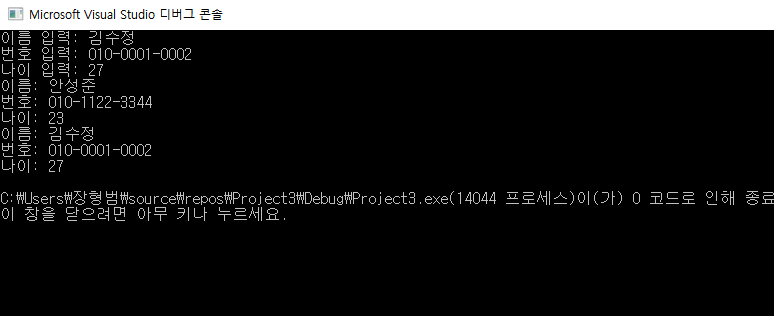
printf("이름: %s \n", man2.name);//"이름" 출력, man2.name에 이름 입력

printf("번호: %s \n", man2.phoneNum);//"번호" 출력, man2.name에 번호 입력

printf("나이: %d \n", man2.age);//"나이" 출력, man2.name에 나이 입력

return 0;//종료

}



한줄토의 : 구조체 배열에 대한 학습

실습3

#include <stdio.h>//헤더파일 선언

struct point//point 구조체 정의

{

int xpos;//point 구조체를 구성하는 xpos

int ypos;//point 구조체를 구성하는 ypos

};

struct person//person구조체 정의

{

char name[20];//person 구조체를 구성하는 name[20]

char phoneNum[20];//person 구조체를 구성하는 phoneNum[20]

int age;//person 구조체를 구성하는 age

char a[30];//배열 선언

};

struct building//building구조체 정의

{

char name[20];//building 구조체를 구성하는 name[20]

char bunji[20];//building 구조체를 구성하는 bunji[20]

char cheng[20];//building 구조체를 구성하는 cheng[20]

char price[20];//building 구조체를 구성하는 prince[20]

};

int main()//main 함수 선언

{

struct point pos = { 10,20 };//pos값 초기화

struct person man = { "이승기", "010-1212-0001", 21,"성적 : 70점" };//man값 초기화

struct building build = { "광운대학교","21번지","3 층","매매가 100억" };//build값 초기화

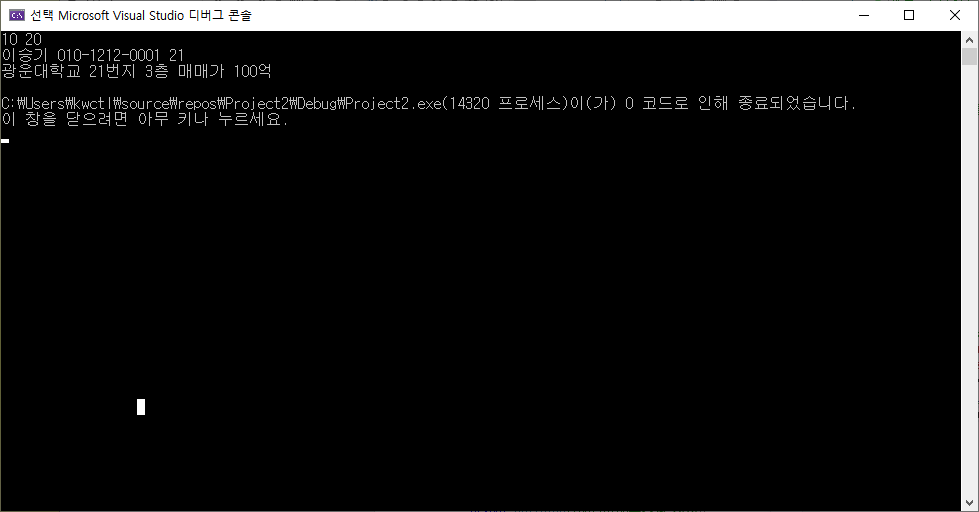
printf("%d %d \n", pos.xpos, pos.ypos);//point구조체 출력

printf("%s %s %d \n", man.name, man.phoneNum, man.age,man.a);//person구조체 출력

printf("%s %s %s %s \n", build.name, build.bunji, build.cheng, build.price);//building구조체 출력

return 0;//종료

}



한줄토의 : 구조체 변수의 초기화 방법을 학습했다.

실습4

#include <stdio.h>//헤더파일 선언

struct point//point구조체 선언

{

int xpos;//point 구조체를 구성하는 xpos

int ypos;//point 구조체를 구성하는 ypos

};

int main()//main함수 선언

{

struct point arr[3];//구조체 배열 선언

int i;//int형 변수 선언

double distance,distance1,distance2;//double형 변수 선언

for (i = 0; i < 3; i++)//3번 반복하는 반복문 선언

{

printf("점의 좌표 입력: ");//출력

scanf("%d %d", &arr[i].xpos, &arr[i].ypos);//i값 xpos,ypos에 입력

}

for (i = 0; i < 3; i++)//3번 반복하는 반복문 선언

printf("[%d,%d]", arr[i].xpos, arr[i].ypos);//i값 xpos, ypos출력

printf("\n");

distance = sqrt((double)((arr[0].xpos - arr[1].xpos)\*(arr[0].xpos - arr[1].xpos) + (arr[0].ypos - arr[1].ypos)\*(arr[0].ypos - arr[1].ypos)));//점 사이의 길이 계산

distance1 = sqrt((double)((arr[1].xpos - arr[2].xpos)\*(arr[1].xpos - arr[2].xpos) + (arr[1].ypos - arr[2].ypos)\*(arr[1].ypos - arr[2].ypos)));//점 사이의 길이 계산

distance2 = sqrt((double)((arr[2].xpos - arr[0].xpos)\*(arr[2].xpos - arr[0].xpos) + (arr[2].ypos - arr[0].ypos)\*(arr[2].ypos - arr[0].ypos)));//점 사이의 길이 계산

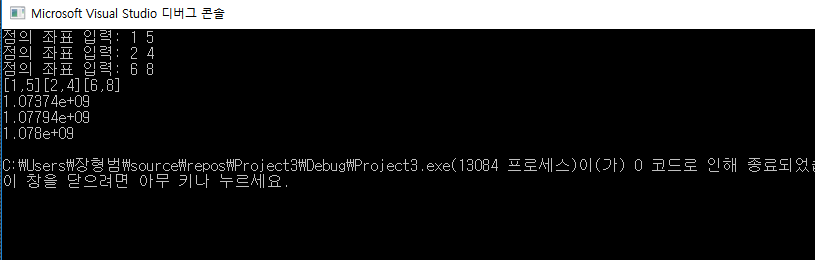
printf("%g \n", distance);//출력문

printf("%g \n", distance1);//출력문

printf("%g \n", distance2);//출력문

return 0;//종료

}



한줄토의 : 구조체 변수 배열을 알게 됨

실습 5

#include <stdio.h>//헤더파일 선언

struct person //person 구조체

{

char name[20];//person구조체를 구성하는 name

char phoneNum[20];//person구조체를 구성하는 phoneNum

int age;//person 구조체를 구성하는 age

int score;//person 구조체를 구성하는 score

};

struct building//building 구조체

{

char name[20]; //building 구조체를 구성하는 name

int bunji; //building 구조체를 구성하는 bunji

int floor; //building 구조체를 구성하는 floor

int price; //building 구조체를 구성하는 price

};

int main()//main함수 선언

{

/\*person 구조체 변수 arr배열 선언 및 초기화\*/

struct person arr[3] = {

{"이승기","010-1212-0001",21,90},

{"정지영","010-1313-0002",22,80},

{"한지수","010-1717-0003",19,70}

};

/\*building 구조체 변수 arr1배열 선언 및 초기화\*/

struct building arr1[3] = {

{"이승기",1,1,2},

{"정지연",2,2,3},

{"한지수",3,3,4}

};

int i; //변수 선언

for (i = 0; i < 3; i++)//구조체 출력 반복문

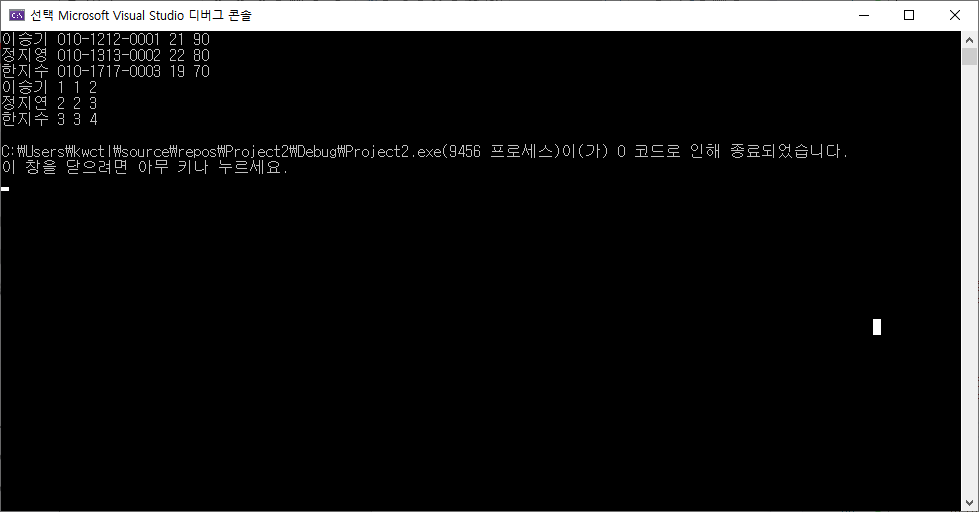
printf("%s %s %d %d \n", arr[i].name, arr[i].phoneNum, arr[i].age,arr[i].score);

for (i = 0; i < 3; i++)//구조체 출력 반복문

printf("%s %d %d %d \n", arr1[i].name, arr1[i].bunji, arr1[i].floor,arr1[i].price);

return 0;//종료

}



한줄 토의: 구조체 내의 다양한 자료형을 배웠다

실습6

#include <stdio.h>//헤더파일 선언

struct point//point라는 이름의 구조체 정의

{

int xpos;//point 구조체를 구성하는 xpos

int ypos;//point 구조체를 구성하는 ypos

};

int main()//메인함수 선언

{

struct point pos1 = { 1,2 };//point 구조체 pos1 선언 후 초기화

struct point pos2 = { 100,200 };//point 구조체 pos2 선언 후 초기화

struct point \* pptr = &pos1;//pos1의 주소를 가리키는 포인터 변수 선언

struct point \* ppptr = &pos2;//pos2의 주소를 가리키는 포인터 변수 선언

double distance;

(\*pptr).xpos += 4;//xpos의 값 변경

(\*pptr).ypos += 5;//ypos의 값 변경

printf("[%d, %d] \n", pptr->xpos, pptr->ypos);//출력문

pptr->xpos += 1;//xpos의 값 변경

pptr->ypos += 2;//ypos의 값 변경

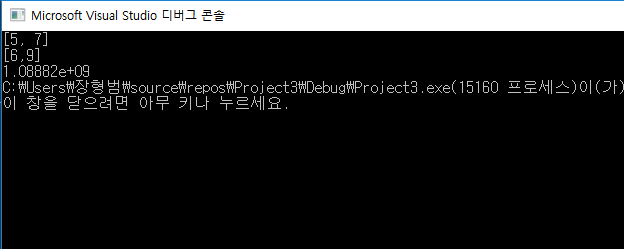
printf("[%d,%d] \n", (\*pptr).xpos, (\*pptr).ypos);//출력문

distance = sqrt((double)(((\*pptr).xpos - (\*ppptr).xpos)\*(pptr->xpos - ppptr->xpos) + ((\*pptr).ypos - (\*ppptr).ypos)\*(pptr->ypos - ppptr->ypos)));//점 사이의 길이 계산

printf("%g", distance);//distance값 출력문

return 0;//종료

}



한줄토의 : 구조체 변수와 포인터에 대해 학습함

실습 7

#include <stdio.h>//헤더파일 선언

struct point//point구조체 정의

{

int xpos;//int형 변수 xpos선언

int ypos;//int형 변수 ypos선언

};

struct circle//circle구조체 정의

{

double radius;//double형 변수 radius선언

struct point \*center;//포인터 변수 center 선언

};

int main()//메인함수 선언

{

double distance;

struct point cen = { 2,7 };//point구조체 can 변수 선언 후 초기화

double rad = 5.5;//double형 변수 선언 후 초기화

struct point cen2; //point구조체 cen2변수 선언 후 초기화

cen2.xpos = (cen.xpos) \* 5;//새로운 점의 xpos

cen2.ypos = (cen.ypos) \* 5;//새로운 점의 ypos

printf("새로운 점: %d %d \n", cen2.xpos, cen2.ypos);//새로운 점 출력

printf("직사각형의 넓이: %d \n", ((cen2.xpos) - (cen.xpos))\*((cen2.ypos) - (cen.ypos)));//직사각형의 넓이 출력

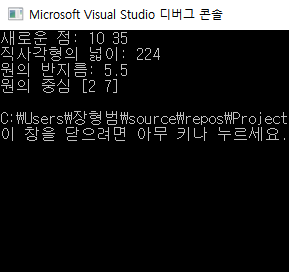
struct circle ring = { rad, &cen };//circle구조체 변수 ring 선언 후 초기화

printf("원의 반지름: %g \n", ring.radius);//출력문

printf("원의 중심 [%d %d] \n", (ring.center)->xpos, (ring.center)->ypos);//출력문

return 0;//종료

}



한줄토의: 구조체의 응용을 학습함.

실습8

#include <stdio.h>//헤더파일 선언

struct point//point 구조체 선언

{

int xpos;//point 구조체를 구성하는 xpos

int ypos;//point 구조체를 구성하는 ypos

struct point \*ptr;//point 구조체를 구성하면서 구조체를 가리키는 포인터

};

int main()//메인함수 선언

{

struct point pos1 = { 1,1 };//point 구조체 변수 pos1 선언 후 초기화

struct point pos2 = { 2,2 };//point 구조체 변수 pos2 선언 후 초기화

struct point pos3 = { 3,3 };//point 구조체 변수 pos3 선언 후 초기화

pos1.ptr = &pos2;//pos1의 구조체 중 포인터가 pos2를 가리키게 함

pos2.ptr = &pos3;//pos2의 구조체 중 포인터가 pos3를 가리키게 함

pos3.ptr = &pos1;//pos3의 구조체 중 포인터가 pos1를 가리키게 함

printf("점의 연결관계... \n");//출력문

printf("[%d,%d]와(과) [%d, %d]연결 \n",

pos1.xpos, pos1.ypos, pos1.ptr->xpos, pos1.ptr->ypos);//출력문

printf("[%d,%d]와(과) [%d, %d]연결 \n",

pos2.xpos, pos2.ypos, pos2.ptr->xpos, pos2.ptr->ypos);//출력문

printf("[%d,%d]와(과) [%d, %d]연결 \n",

pos3.xpos, pos3.ypos, pos3.ptr->xpos, pos3.ptr->ypos);//출력문

double distance12 = sqrt((double)((pos1.ptr->xpos - pos1.xpos)\*(pos1.ptr->xpos - pos1.xpos) + (pos1.ptr->ypos - pos1.ypos)\*(pos1.ptr->ypos - pos1.ypos)));//점1과 2를 연결하는 선의 길이

double distance23 = sqrt((double)((pos2.ptr->xpos - pos2.xpos)\*(pos2.ptr->xpos - pos2.xpos) + (pos2.ptr->ypos - pos2.ypos)\*(pos2.ptr->ypos - pos2.ypos)));//점2와 3을 연결하는 선의 길이

double distance31 = sqrt((double)((pos3.ptr->xpos - pos3.xpos)\*(pos3.ptr->xpos - pos3.xpos) + (pos3.ptr->ypos - pos3.ypos)\*(pos3.ptr->ypos - pos3.ypos)));//점3과 1을 연결하는 선의 길이

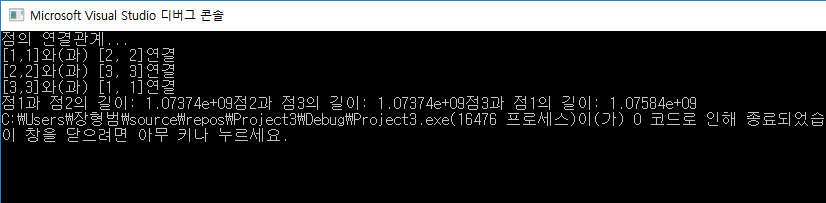
printf("점1과 점2의 길이: %g", distance12);//점 1과 2를 연결하는 선의 길이 출력문

printf("점2과 점3의 길이: %g", distance23);//점 2와 3을 연결하는 선의 길이 출력문

printf("점3과 점1의 길이: %g", distance31);//점 3과 1을 연결하는 선의 길이 출력문

return 0;//종료

}



한줄토의: 구조체에 대해 알게됨

실습 9

#include <stdio.h>

struct employee // employee 구조체 선언

{

char name[30]; // char형 일차원배열 name선언

char pin[30]; // char형 일차원배열 pin 선언

int sal; // int형 변수 sal 선언

};

int main(void) // 메인함수 선언

{

struct employee priv; // 구조체 변수 arr[3] 선언

printf("이름: "); // 출력문

scanf("%s", priv.name); // priv.name 출력

printf("주민번호: "); // 출력문

scanf("%s", priv.pin); // priv.pin 출력

printf("급여: "); // 출력문

scanf("%d", &priv.sal); //priv.sal에 입력값 저장

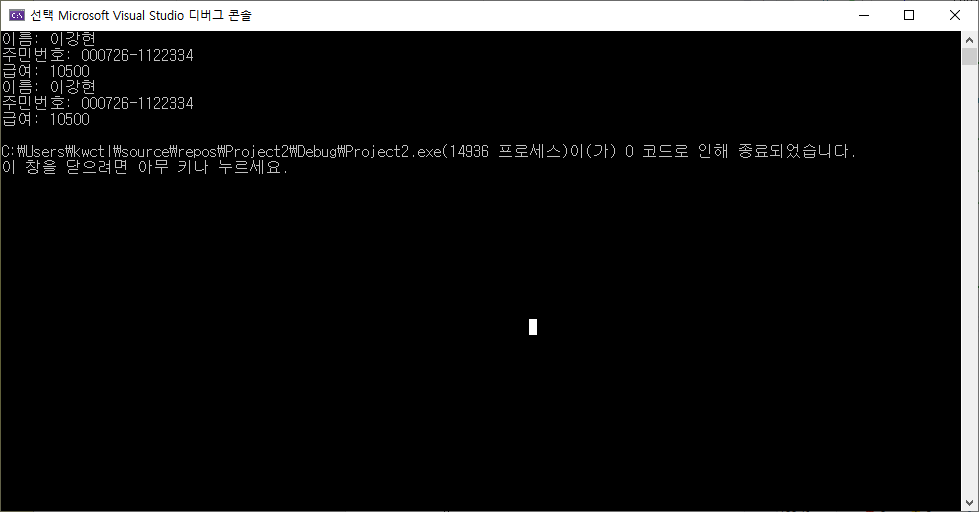
printf("이름: %s\n", priv.name); // priv.name 출력

printf("주민번호: %s \n", priv.pin); // priv.pin 출력

printf("급여: %d\n", priv.sal); // priv.sal 출력

return 0; //종료

}



한줄 토의: 구조체를 통한 정보입력을 해보았다.

실습 10

#include <stdio.h>

struct employee // employee 구조체 선언

{

char name[30]; // char형 일차원배열 name선언

char pid[30]; // char형 일차원배열 pid 선언

int salary; // int형 변수 salary 선언

};

int main(void) // 프로그램 시작 함수

{

struct employee arr[3]; // 구조체 변수 arr[3] 선언

int i; // int형 변수 i 선언

for (i = 0; i < 3; i++) // i가 0부터 2까지 세 번 반복하는 for문

{

printf("이름: "); // 출력문

scanf("%s", arr[i].name); // arr[i].name 출력

printf("주민번호: "); // 출력문

scanf("%s", arr[i].pid); // arr[i].pid 출력

printf("급여: "); // 출력문

scanf("%d", &arr[i].salary); //arr[i].salary에 입력값 저장

}

for (i = 0; i < 3; i++) // // i가 0부터 2까지 세 번 반복하는 for문

{

printf("이름: %s\n", arr[i].name); // arr[i].name 출력

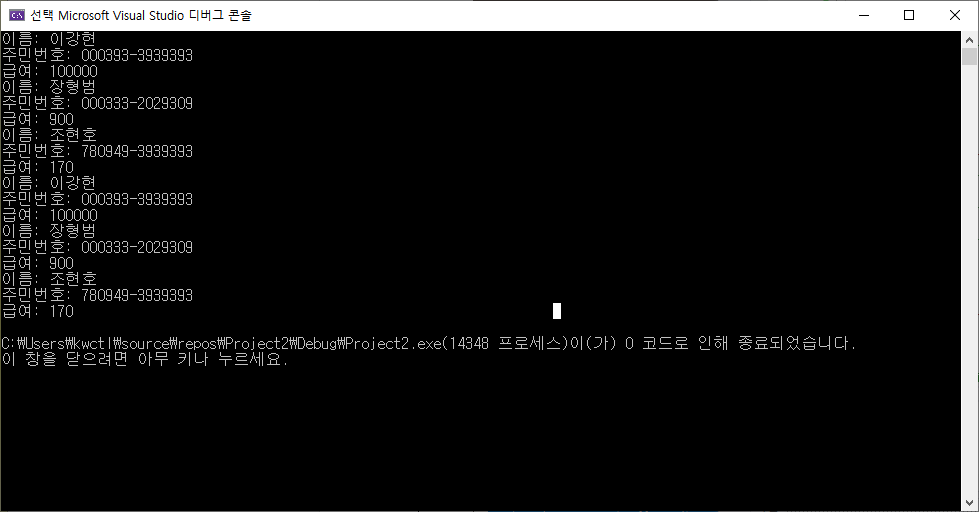
printf("주민번호: %s \n", arr[i].pid); // arr[i].pid 출력

printf("급여: %d\n", arr[i].salary); // arr[i].salary 출력

}

return 0; // 종료

}



한줄토의: 반복문을 통해 실습9예제의 응용을 해보았다.