|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2020\_1\_C\_11주 | 학번 : | 20175105 | 이름 : | 곽영주 |

* **강의 내용**

1. 구조체 배열 활용 - 정렬

|  |
| --- |
| [프로그램 소스]  #include <stdio.h>  typedef struct {  char food[20]; // 메뉴명  int cost; // 가격  }mlist; // 자료형 재정의  int main() {  mlist temp; // 임시변수  mlist menu[] = { {"라면",2000},{"떡볶이",1500},{"김밥",1000},{"돈까스",5000},{"비빔밥",4000},{"육개장",4500},{"순두부정식",3500} };    int i, j;  int size = sizeof(menu) / sizeof(mlist); // 배열 크기 계산    printf("메뉴 리스트 - 정렬 전 :\n");  printf("Foods === Cost(원)\n");  for (i = 0; i < size; i++)  printf("%-12s %5d\n", menu[i].food, menu[i].cost);  // 정렬 기준 - 가격  for (i = 0; i < size - 1; i++) {  for (j = i + 1; j < size; j++) {  if (menu[i].cost > menu[j].cost) {  temp = menu[i];  menu[i] = menu[j];  menu[j] = temp;  }  }  }  printf("\n메뉴 리스트 - 가격순 정렬 후:\n");  printf("Foods === Cost(원)\n");  for (i = 0; i < size; i++)  printf("%-12s %5d\n", menu[i].food, menu[i].cost);  return 0;  } |
| [실행 결과] |

1. 구조체 배열과 포인터 활용

|  |
| --- |
| [프로그램 소스]  #define \_CRT\_SECURE\_NO\_WARNINGS  #define size 3  #include <stdio.h>  #include <stdlib.h>  // 구조체 정의와 typedef 함께 사용  typedef struct {  int beon; // 번호  char name[10]; // 이름  int python, c; // 과목 점수  int hap; // 과목 합  }subject;  int main() {  subject sub[size];  subject \*subpt;  int i;  for (i = 0; i < size; i++) {  printf("번호 >> ");  scanf("%d", &sub[i].beon);  getchar();  printf("이름 >> ");  gets(sub[i].name); // 공백이 포함된 문자열 입력  printf("과목 점수(python, c) >> ");  scanf("%d %d", &sub[i].python, &sub[i].c);  sub[i].hap = sub[i].python + sub[i].c;  }  subpt = sub; // 구조체 배열 포인터  printf("\n 번호 이름 python c 총합\n");  printf("==========================================\n");  for (i = 0; i < size; i++) {  printf("%3d %10s %3d %3d %3d\n", (subpt + i)->beon, (\*(subpt + i)).name, subpt[i].python, (sub + i)->c, sub[i].hap);  }  printf("==========================================\n");  return 0;  } |
| [실행 결과] |

1. 구조체 배열을 함수로 전달

|  |
| --- |
| [프로그램 소스]  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  typedef struct {  char food[20];  int cost;  }mlist;  void menu\_list(mlist \*, int); // 메뉴 리스트 출력  void menu\_sort(mlist \*, int); // 메뉴 가격 오름차순 정렬  int main() {  mlist temp;  mlist menu[] = { {"라면",2000},{"떡볶이",1500},{"김밥",1000},{"돈까스",5000},{"비빔밥",4000},{"육개장",4500},{"순두부정식",3500} };  int i, j;  int size = sizeof(menu) / sizeof(mlist);  printf("메뉴 리스트 - 정렬 전 :\n");  printf("Foods === Cost(원)\n");    menu\_list(menu, size);  menu\_sort(menu, size);  printf("\n메뉴 리스트 - 가격순 정렬 후:\n");  printf("Foods === Cost(원)\n");  menu\_list(menu, size);  return 0;  }  void menu\_list(mlist \*list, int size) {  int i;  for (i = 0; i < size; i++)  printf("%-12s %5d\n", (list + i)->food, (list + i)->cost);  }  void menu\_sort(mlist\* menu, int size) {  int i, j;  mlist temp;  for (i = 0; i < size - 1; i++) {  for (j = i + 1; j < size; j++) {  if ((menu + i)->cost > menu[j].cost) {  temp = \*(menu + i); // temp = menu[i];  \*(menu + i) = \*(menu + j);  \*(menu + j) = temp;  }  }  }  } |
| [실행 결과] |

* **내용 점검**

1. 다음 각각에서 설명하는 문장을 작성하시오.
2. 년, 월, 일 정보로 구성된 구조체 date를 정의

struct date {

int y, m, d;

};

1. 위 구조체 date를 다시 자료형 datetype으로 재정의

typedef struct date datetype;

1. 이름과 주민번호 그리고 생년월일을 저장할 구조체 person을 정의, 위에서 정의한 자료형 datetype을 사용

struct person {

char name[10];

char number[20]; // 주민번호

datetype ymd; // 생년월일

};

1. 공간좌표 x, y, z를 표현하는 구조체 coordinate 정의, 멤버의 자료형은 double

struct coordinate {

double x, y, z;

};

1. 구조체 coordinate 형 변수 point를 선언하면서 초기 값으로 3.1, 4.7, -5.2 저장

struct coordinate point = { 3.1,4.7,-5.2 };

1. 다음 문장에서 오류가 있거나 적절하지 못한 문장을 찾아 수정하시오.
2. struct professor {char name[10], int number};

* struct professor {char name[10]; int number;};

1. typedef struct {char name[10]; int number;};

* typedef struct {char name[10]; int number;}professor;

1. struct grade {int mid = 96; int final = 86; char grade[2];};

* struct grade {int mid; int final; char grade[2];};

1. struct grade {int mid; int final;} g;

g = 1.5;

* struct grade {int mid; int final;} g;

g.mid = 1.5;

1. struct grade {int mid; int final;} g;

struct grade \*p = g;

* struct grade {int mid; int final;} g;

struct grade \*p = &g;

1. struct professor {char name[10]; int number;} prof;

struct professor \*p = &prof;

p.number = 10001;

* struct professor {char name[10]; int number;} prof;

struct professor \*p = &prof;

p->number = 10001;

1. struct professor {char name[10]; int number;} prof;

struct professor \*p = &prof;

p->name = "이혜인";

* struct professor {char name[10]; int number;} prof;

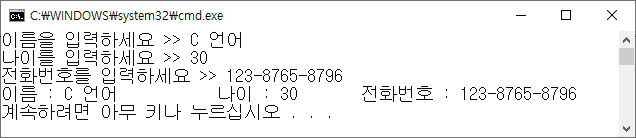
struct professor \*p = &prof;

strcpy(p->name, "이혜인");

* **프로그래밍 과제**

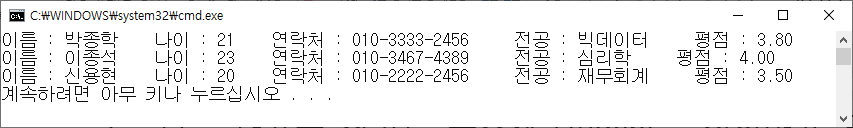
1. 구조체 person을 정의하고, 적당한 값을 입력 받아 구조체에 저장하고 출력하는 프로그램을 작성하시오.

* 구조체 person 멤버 구성: 이름, 나이, 전화번호



|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  struct person {  char name[10]; // 이름  int age; // 나이  char number[20]; // 전화번호  };  int main() {  struct person p;  printf("이름을 입력하세요 >> ");  gets\_s(p.name, sizeof(p.name));  printf("나이를 입력하세요 >> ");  scanf\_s("%d", &p.age);  getchar();  printf("전화번호를 입력하세요 >> ");  gets\_s(p.number, sizeof(p.number));  printf("이름 : %s 나이 : %d 전화번호 : %s", p.name, p.age, p.number);  return 0;  } |
| **[실행 결과]** |

1. 다음 내용을 참고로 구조체 student를 정의하고, 구조체 배열로 학생 3명을 선언하여 적당한 값으로 초기화 한 후 출력하는 프로그램을 작성하시오. 단, 이름과 나이 전화번호는 1번 문제에서 정의된 구조체를 멤버로 사용한다



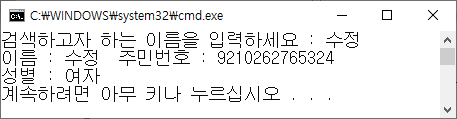
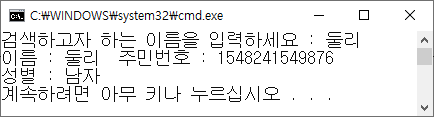
|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  struct person {  char name[10]; // 이름  int age; // 나이  char number[20]; // 전화번호  };  struct student {  struct person p; // 이름, 나이, 전화번호  char department[20]; // 전공  double grade; // 평점  };  int main() {  struct student stu[3] = { {{"박종학",21,"010-3333-2456"},"빅데이터",3.8 },  {{"이종석",23,"010-3467-4389"},"심리학",4.0 },  {{"신용현",20,"010-2222-2456"},"재무회계",3.5 } };  int i;  for (i = 0; i < 3; i++) {  printf("이름 : %s 나이 : %d 연락처 : %s 전공 : %10s 평점 : %.2f\n",  stu[i].p.name, stu[i].p.age, stu[i].p.number, stu[i].department, stu[i].grade);  }    return 0;  } |
| **[실행 결과]** |

1. 다음 내용을 참고로 구조체 human 을 정의하고 5명의 구조체 배열을 선언하면서 초기화 한 후 제시된 조건대로 처리하시오.

* 구조체 human 멤버 구성: 이름(char), 주민등록번호(char)
* 조건1 : 사용자로부터 입력된 이름과 일치하는 이름이 존재하면 주민등록 번호를 출력 한다. – strcmp()
* 조건2 : 조건1을 수행하면서 성별을 구분하여 출력 하시오 – 주민등록번호를 사용하여 남/여 구분

human st[] = { { "둘리", "1548241549876" }, { "길동", "6814651274935" }, { "마이콜", "2645182975483" },

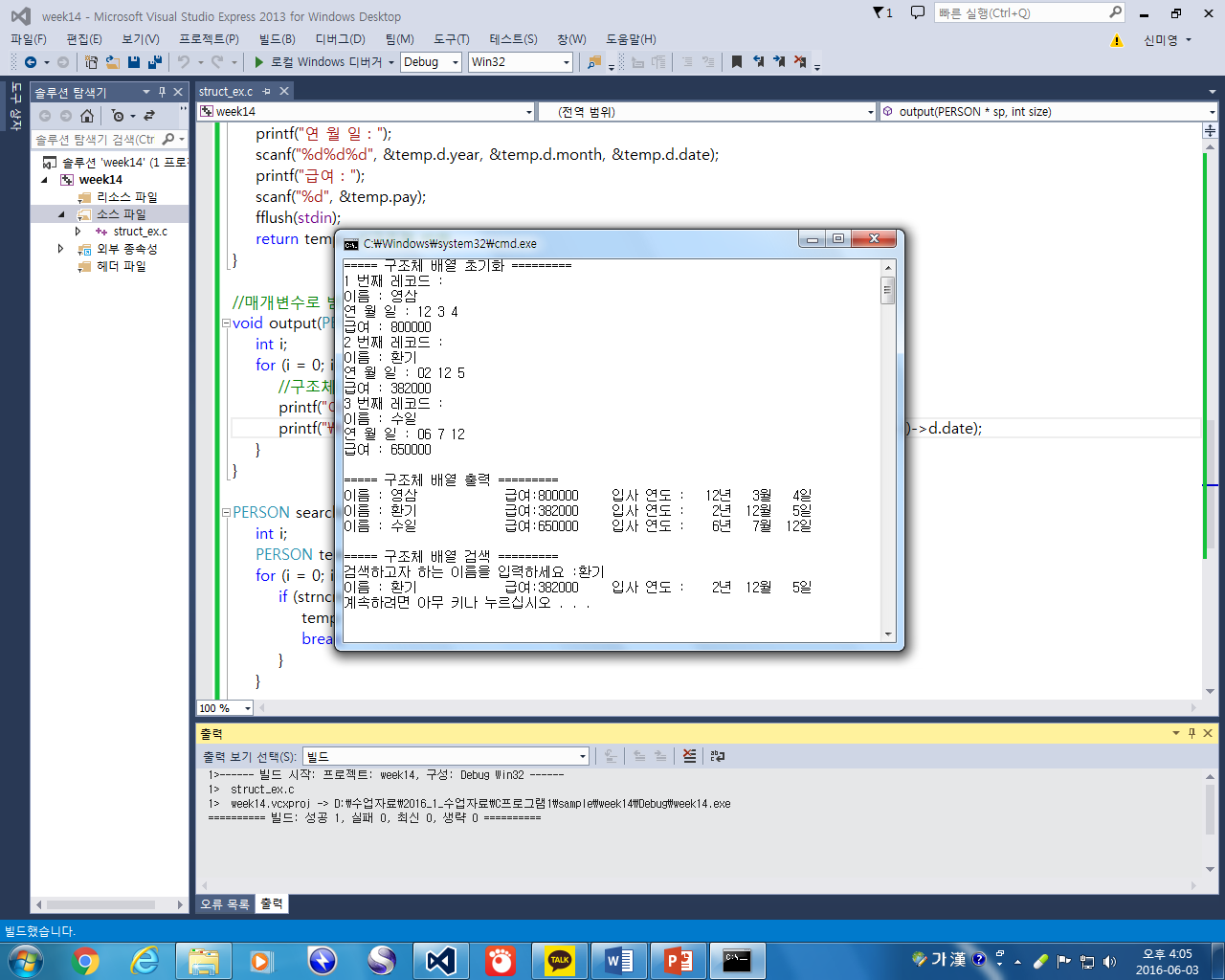
{ "수정", "9210262765324" }, { "또치", "7804161549357" }};

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  #include <string.h>  typedef struct {  char name[10]; // 이름  char number[20]; // 주민번호  }human;  int main() {  human st[5] = { { "둘리", "1548241549876" }, { "길동", "6814651274935" },  { "마이콜", "2645182975483" }, { "수정", "9210262765324" },  { "또치", "7804161549357" } };  char ch[10]; // 검색할 이름  int i;  char \*n; // 주민번호 포인터  printf("검색하고자 하는 이름을 입력하세요 : ");  scanf\_s("%s", ch, sizeof(ch));  for (i = 0; i < 5; i++) {  if (strcmp(ch, st[i].name) == 0) {  printf("이름 : %s 주민번호 : %s\n", st[i].name, st[i].number);    n = st[i].number; // 검색한 이름의 주민번호  if (\*(n+6) == '1') // 남자인 경우  printf("성별 : 남자\n");  else // 2 // 여자인 경우  printf("성별 : 여자\n");  }  }  return 0;  } |
| **[실행결과]** |

1. 5사람의 개인정보를 처리하는 프로그램을 제시된 조건대로 작성하시오

* 조건 1 : PERSON 구조체(이름, 년, 월, 일, 급여)를 정의하여 사용하며 크기가 5인 구조체 배열로 처리한다
* 조건 2 : 배열 초기화 시 input() 함수를 사용한다
* 조건 3 : 입력된 이름과 같은 레코드를 검색하여 출력한다. 단, 적당한 함수를 만들어 처리하며 매개변수로 배열과 찾고자 하는 이름을 전달한다. 검색한 레코드를 반환한다
* 조건 4 : 조건 3에서 해당 레코드를 검색하지 못하면 기본값으로 초기화된 구조체를 반환한다. 즉, 문자열은 “”, 숫자는 0으로 초기화



#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <string.h> //문자열 처리

PERSON input(); //입력 받은 값으로 구조체초기화 한 후 반환

void output(PERSON \*, int); //구조체 배열과 크기를 매개변수로 받아 출력, 반환 값 없음

//구조체 검색 결과 반환, 구조체 배열, 크기, 찾고자 하는 이름을 매개변수로 전달

PERSON search(PERSON \*, char \*, int);

int main(){

//구조체 변수 선언 & 초기화

char ireum[10]; //검색하고자 하는 이름을 입력 받아 저장

printf("===== 구조체 배열 초기화 =========\n");

//본인작성

printf("\n===== 구조체 배열 출력 =========\n");

//본인작성

printf("\n===== 구조체 배열 검색 =========\n");

printf("검색하고자 하는 이름을 입력하세요 :");

gets\_s(ireum, 10);

//검색 결과로 반환 받은 구조체 출력 – 본인 작성

return 0;

}

//구조체를 반환하는 함수

PERSON input(){

//본인작성

}

//매개변수로 받은 구조체 출력

void output(PERSON \*sp, int size){

//본인작성

}

PERSON search(PERSON \*sp, char \*name, int size){

//본인작성

}

|  |
| --- |
| **[프로그램 소스]**  #define \_CRT\_SECURE\_NO\_WARNINGS  #include <stdio.h>  #include <string.h>  #define SIZE 5 // 크기 5  typedef struct {  char name[10]; // 이름  int y, m, d; // 년 월 일  int salay; // 급여  }PERSON;  PERSON input(); // 입력 받은 값으로 구조체 초기화 한 후 반환  void output(PERSON \*, int); // 구조체 배열과 크기를 매개변수로 받아 출력  PERSON search(PERSON \*, char \*, int); // 구조체 검색 결과 반환  int main() {  PERSON p[SIZE];  char ireum[10]; //검색하고자 하는 이름을 입력 받아 저장  int i;  printf("===== 구조체 배열 초기화 =========\n");  for (i = 0; i < SIZE; i++) {  printf("%d 번째 레코드 :\n", i + 1);  if (i > 0)  getchar(); // 두 번째부터 이름을 제대로 출력하기 위해 getchar();  p[i] = input();  }  printf("\n===== 구조체 배열 출력 =========\n");  output(p,SIZE);  printf("\n===== 구조체 배열 검색 =========\n");  printf("검색하고자 하는 이름을 입력하세요 :");  getchar();  gets\_s(ireum, 10);  PERSON a = search(p, ireum, SIZE);  printf("이름 : %10s 급여 : %10d 입사연도 : %2d년 %2d월 %2d일\n", a.name, a.salay, a.y, a.m, a.d);  return 0;  }  PERSON input() { //구조체를 반환하는 함수  PERSON data;  printf("이름 : ");  gets\_s(data.name, sizeof(data.name));  printf("년 월 일 : ");  scanf\_s("%d %d %d", &data.y, &data.m, &data.d);  printf("급여 : ");  scanf\_s("%d", &data.salay);  return data;  }  void output(PERSON\* sp, int size) { //매개변수로 받은 구조체 출력  int i;  for (i = 0; i < size; i++) {  printf("이름 : %10s 급여 : %10d 입사연도 : %2d년 %2d월 %2d일\n",  (sp + i)->name, (sp + i)->salay, (sp + i)->y, (sp + i)->m, (sp + i)->d);  }  }  PERSON search(PERSON\* sp, char\* name, int size) {  PERSON data = { "",0,0,0,0 }; // 기본값으로 초기화  int i;  for (i = 0; i < size; i++) {  if (strcmp((sp + i)->name, name) == 0) // 찾는 내용이 있으면  data = \*(sp + i); // data에 저장  // 찾는 내용이 없으면 처음 초기화한 기본값 { "",0,0,0,0 } 반환  }  return data;  } |
| **[실행결과]** |