**C 프로그래밍II 과제**

**개인 정보**

|  |  |
| --- | --- |
| 과제번호(과제제출시스템 상의 일련번호) | *27* |
| 제출자 학번 | *201632034* |
| 제출자 성명 | *한영빈* |
| 소속학과 | *소프트웨어공학과* |
| 학년 | *1* |
| 제출 의무일 | *2016. 11. 23* |
| 실제 제출일 | *2016. 11. 19.* |

**평가 항목**

|  |  |  |
| --- | --- | --- |
| 항목 | 만점 | 점수 |
| 제출일 관련 점수 | 2 |  |
| 원하는 결과가 출력되나? | 2 |  |
| 프로그램 코드가 문제의 논리에 맞게 구성되었나? | 5 |  |
| 문제 해결 방법이 바람직한가?(평가참고항목 문제해결방법에 따라 점수부여) | 4 |  |
| 프로그램 코드가 읽기 쉬운 형태로 잘 구성되어 있나? (변수 이름, 들여쓰기, 주석등) | 2 |  |
| 과제 제출 양식을 잘 맞추었나? | 3 |  |
| 해결하기 어려웠던 부분에 대한 설명이 잘 되어있나? (어려웠던 부분이 없었던 경우, 이를 명시.) | 2 |  |
|  | 20 |  |

**평가 참고 항목(해당 칸에 표기하시오)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 자신이 판단한 문제 난이도 | | 상 | 중 O | | 하 |
| 문제 해결에 투여한 시간 | | 1 시간 | | | |
| 문제해결 방법(해당 사항에 0표 한다.) | 책을 참고하여 혼자서 해결 | O | | 4점 | |
| 인터넷을 참고하여 혼자서 해결 |  | | 4점 | |
| ( )의 지도를 받음[[1]](#footnote-1) |  | | 3점 | |
| 타인(친구, 선배, 인터넷)의 코드를 복사 또는 조금 수정(모든 항목 0점처리되어 과제 점수가 0점이 됨) |  | | 0점 | |

**C 프로그래밍II 과제**

과제출제자: 홍은지

과제출제일: 2016-11-16

과제 내용

1. 혼자 해결하는 과제(search.c)
   1. 학번, 이름, 학과를 저장하는 구조체를 정의한다. typedef를 이용한다.
   2. 한글 이름은 2바이트로 표현되므로, 10개의 한글을 저장하기 위해서는 20 바이트가 필요하다.
   3. main() 함수에서
      1. (1)의 구조체를 10개 저장하는 배열을 정의한다. 적당한 값으로 초기화 한다.
      2. 검색하고 싶은 학생의 이름을 입력 받는다.
      3. 배열에서 해당 이름이 있으면, 해당 학생의 학번, 이름, 학과를 보기 좋게 출력한다.
      4. 만약 해당 학생이 없으면, 해당 학생이 존재하지 않는다고 출력한다.
   4. 과제 소스 파일 명: search.c
   5. 과제 실행 파일 명: search.exe

소스코드 – search.c

/\*

이름 : 한영빈

학과 : 소프트웨어공학과

학번 : 201632034

제목 : 학생 정보 조회

작성일 : 2016.11.19

\*/

#include<stdio.h>

#include<string.h>

// 학생 정보를 저장할 사용자 정의 자료형을 구조체로 정의

typedef struct{

int id;

char \* name;

char \* major;

}STUDENT;

int main(void){

// STUDENT 형 자료 10개 담는 배열 초기화

STUDENT students[] = {

{201632034, "한영빈", "소프트웨어공학과"},

{201534028, "추건우", "소프트웨어공학과"},

{201232016, "배다슬", "소프트웨어공학과"},

{201632032, "최강식", "소프트웨어공학과"},

{201632033, "최준호", "소프트웨어공학과"},

{201632035, "판사님", "소프트웨어공학과"},

{000032034, "고양이", "소프트웨어공학과"},

{000132034, "홍길동", "소프트웨어공학과"},

{111132032, "임꺽정", "소프트웨어공학과"},

{000132034, "김홍도", "소프트웨어공학과"},

};

char nameToSearch[20]; // 입력받은 문자열 저장할 변수

int i; // 반복문에 사용할 변수

// 검색 키워드 입력받기

fputs("검색할 학생의 이름을 입력하세요.\n", stdout);

fgets(nameToSearch, sizeof(nameToSearch), stdin);

// 반복문을 이용하여 하나씩 대조

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

// 일치하면 출력

if(strncmp(nameToSearch, students[i].name, sizeof(students[i].name))==0){

printf("학번 : %d\n", students[i].id);

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

printf("학과 : %s\n", students[i].major);

// 출력 후 종료

return 0;

}

}

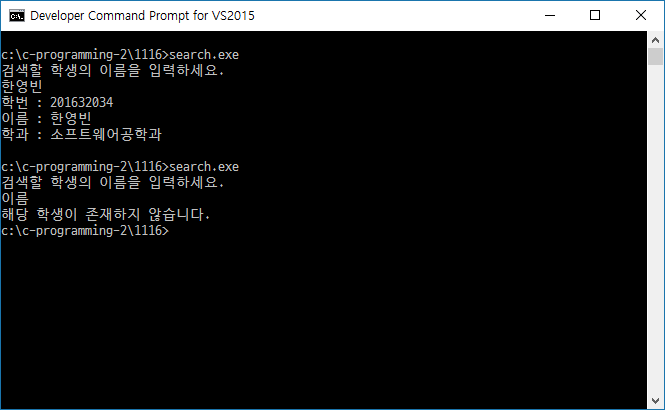
// 일치하는 경우가 없음 - 그냥 종료

fputs("해당 학생이 존재하지 않습니다.", stdout);

return 0;

}

실행 결과 – result.png



해결하기 어려웠던 부분, 과제를 하면서 알게 된 부분에 대한 설명(50자 이상)

항상 인코딩 문제와 잠깐 씨름 하는 것 말고는 해결하기 어려웠던 점은 크게 없었던 것 같다. 문자열 비교 할 때 잠깐 막히기는 한 것 같다. 구조체도 생각보다 별로 어렵지는 않은 것 같다(아직은 그렇다.)

1. 지도해 준 사람의 이름을 명기할 것, 이름이 없는데 유사한 코드이면 복사로 간주 [↑](#footnote-ref-1)