2019202050 이강현 챕터 25 과제

실습 1

#include <stdio.h>

char\*ReadUserName()//반환형이 char인 함수 ReadUserName선언

{

char name[30];//char형 1차원 배열 선언

printf("What's your name?");//프린트

gets(name);//입력값을 get에 입력

return name;//name값 반환

}

int main()

{

char\*name1;//char형 포인터 변수 선언

char\*name2;//char형 포인터 변수 선언

name1 = ReadUserName();//ReadUserName 함수 불러와 name1에 대입

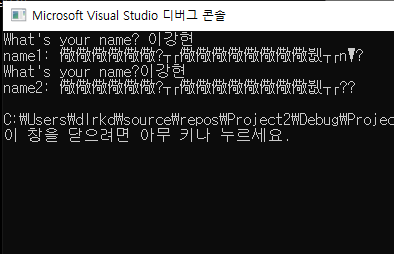
printf("name1: %s\n", name1);//프린트

name2 = ReadUserName();//ReadUserName 함수 불러와 name2에 대입

printf("name2: %s\n", name2);//프린트

return 0;//종료

}



한줄토의 : 주소만 남고 값은 없는 것

실습2

#include <stdio.h>

char name[30];//전역변수 char형 1차원 배열 선언

char \*ReadUserName()//반환형이 char인 포인터 배열 선언

{

printf("What's your name?");//프린트

gets(name);//name에 입력값 입력

return name;//name값 반환

}

int main()//메인함수 선언

{

char\*name1;//char형 포인터 변수 선언

char\*name2;//char형 포인터 변수 선언

name1 = ReadUserName();//ReadUserName함수 불러와 name1에 값 입력

printf("name1: %s \n", name1);//프린트

name2 = ReadUserName();//ReadUserName함수 불러와 name2에 값 입력

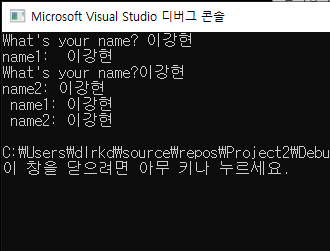
printf("name2: %s \n ", name2);//프린트

printf("name1: %s \n ", name1);//프린트

printf("name2: %s \n ", name2);//프린트

return 0;//종료

}



한줄토의 : 전역변수로 name을 선언하여 값이 소멸되지 않음

실습 3

#include <stdio.h>

int main()//메인함수 선언

{

int\*ptr1 = (int\*)malloc(sizeof(int));//포인터 변수 ptr1선언 후 malloc함수를 이용해 힙 영역에 메모리 할당

int\*ptr2 = (int\*)malloc(sizeof(int) \* 7);//포인터 변수 ptr2 선언 후 malloc함수를 이용해 힙 영역에 메모리 할당함

int i;//int형 변수 선언

\*ptr1 = 20;//ptr1 초기화

for (i = 0; i < 7; i++)//7번 반복하는 반복문

ptr2[i] = i + 1;//ptr[i]에 i+1값 넣음

printf("%d \n", \*ptr1);//프린트

for (i = 0; i < 7; i++)//7번 반복하는 반복문

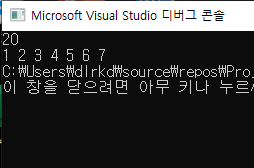
printf("%d ", ptr2[i]);//프린트

free(ptr1);//힙 영역에 할당된 메모리 공간 해제

free(ptr2);//힙 영역에 할당된 메모리 공간 해제

return 0;//종료

}



한줄토의 : malloc함수 사용법을 알게됨

실습 4

#include <stdio.h>

#include<stdlib.h>

char \*ReadUserName()//char형을 반환하는 포인터 함수 선언

{

char \*name = (char\*)malloc(sizeof(char) \* 30);//힙 영역에 char형 메모리 공간을 30만큼 할당

printf("What's your name?");//프린트

gets(name);//입력한 값을 name에 넣음

return name;//name 반환

}

int main()//메인함수 선언

{

char\*name1;//char형 포인터 배열 선언

char\*name2;//char형 포인터 배열 선언

name1 = ReadUserName();//ReadUserName함수를 불러온 뒤 반환한 값을 name1에 넣음

printf("name1: %s \n", name1);//프린트

name2 = ReadUserName();//ReadUserName함수를 불러온 뒤 반환한 값을 name2에 넣음

printf("name2: %s \n ", name2);//프린트

printf("name1: %s \n ", name1);//프린트

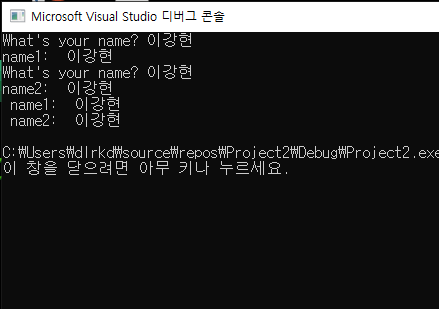
printf("name2: %s \n ", name2);//프린트

free(name1);//메모리 공간 해제

free(name2);//메모리 공간 해제

return 0;//종료

}



한줄토의 :메모리 동적 할당을 알게됨

실습 5

#include <stdio.h>

char\*ReadUserName()

{

char\*name = (char\*)calloc(30, sizeof(char));//char형 포인터 변수 선언 후 calloc함수를 이용해 힙 영역에 30크기의 메모리 할당

printf("What's your name?");//프린트

gets(name);//입력한 값을 name에 넣음

return name;//name 반환

}

int main()//메인함수 선언

{

char\*name1;//char형 포인터 배열 선언

char\*name2;//char형 포인터 배열 선언

name1 = ReadUserName();//ReadUserName함수를 불러온 뒤 반환한 값을 name1에 넣음

printf("name1: %s \n", name1);//프린트

name2 = ReadUserName();//ReadUserName함수를 불러온 뒤 반환한 값을 name2에 넣음

printf("name2: %s \n ", name2);//프린트

printf("name1: %s \n ", name1);//프린트

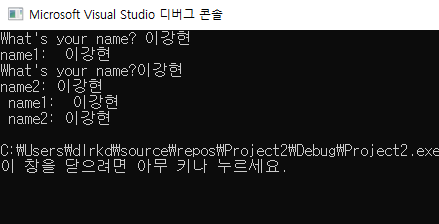
printf("name2: %s \n ", name2);//프린트

free(name1);//메모리 공간 해제

free(name2);//메모리 공간 해제

return 0;//종료

}



한줄토의 : malloc 함수와 calloc 함수의 차이를 알게됨

실습 6

#include <stdio.h>

int main()//메인함수 선언

{

int a,i;//int형 변수 선언

char \* str;//char형 포인터 변수 선언

printf("문자열의 최대길이 입력 : ");//프린트

scanf("%d", &a);//a에 정수 대입

getchar();//\n 문자 제거

str = (char \*)malloc(sizeof(char)\*(a + 1));//str에 char형의 a+1의 메모리 할당 공간을 선언

printf("문자열 입력: ");//프린트

fgets(str,a+1,stdin);//크기가 a+1인 입력값을 str에 넣음

str[strlen(str) - 1] = 0;//\n문자 제거

i = strlen(str);//문자열 크기 정의

for (int j =i; j>0; j--) {//i-1번 만큼 반복하는 반복문 선언

if (str[j] == ' ') {//만약 str[j]가 공백이라면

printf("%s ", &str[j + 1]);//프린트

str[j] = 0;//0으로 초기화

}

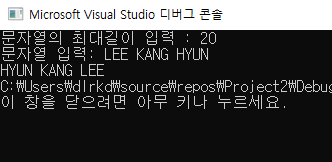
}

printf("%s", &str[0]);//프린트

free(str);//메모리 할당 공간 해제

return 0;//종료

}



한줄토의 : 메모리를 할당하는 법을 배움

실습 7

#include <stdio.h>

int main()//메인함수 선언

{

int arr = 5;//초기값 설정

int \* num = (int \*)malloc(sizeof(int) \* arr);//힙 영역에 할당

int i = 0;//배열 인덱스

puts("정수를 입력하세요 : ");

while (1) {//무한반복문 설정

scanf("%d", &num[i]);//입력

if (num[i] == -1)//만약 -1이면 종료

break;

if (arr == i + 1) {//만약 arr이랑 i+1이 크기가 같으면

arr += 3;//arr에 +3을 더해줌

num = (int \*)realloc(num, sizeof(int) \* arr);//realloc을 이용해 +3추가

}

i++;//i++

}

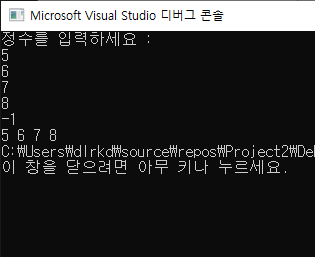
for (int j = 0; j < i; j++)//i가 더해진 값 만큼 출력

printf("%d ", num[j]);

free(num);//메모리 할당 공간 해제

return 0;//종료

}



한줄토의 : realloc함수를 이용하는 법을 알게됨