C프로그래밍 챕터14 과제 2019202050 이강현

실습1

#include <stdio.h>

void ShowArayElem(int \*param, int len)//배열의 주소값과 길이를 인자로 전달받는 함수 정의

{

int i;

for (i = 0; i < len; i++)//배열의 길이를 받아 그만큼 출력하는 반복문

printf("%d", param[i]);

printf("\n");

}

int main(void)

{

int arr1[3] = { 1,2,3 };//배열 선언및 초기화

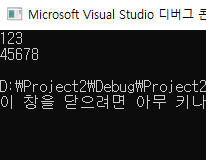
int arr2[5] = { 4,5,6,7,8 };//배열 선언및 초기화

ShowArayElem(arr1, sizeof(arr1) / sizeof(int));//배열의 바이트 크기를 정수형 바이트 크기로 나눠 배열의 길이 계산

ShowArayElem(arr2, sizeof(arr2) / sizeof(int));//배열의 바이트 크기를 정수형 바이트 크기로 나눠 배열의 길이 계산

return 0;

}



한줄 토의: 배열을 통째로 복사할 순 없기에 주소값을 전달한다.

실습2

#include <stdio.h>

void ShowArayElem(int \*param, int len)//배열의 주소값과 길이를 인자로 전달받는 함수 정의

{

int i;

for (i = 0; i < len; i++)//배열의 길이를 받아 그만큼 출력하는 반복문

printf("%d", param[i]);

printf("\n");

}

void AddArayElem(int \*param, int len,int add)//배열의 주소값, 길이,더할 값을 인자로 전달받는 함수 정의

{

int i;

for (i = 0; i < len; i++)//배열의 각 요소에 더할 값을 인자로 받아 더하는 반복문

param[i] += add;

}

int main(void)

{

int arr[3] = { 1,2,3 };//배열 선언및 초기화

AddArayElem(arr, sizeof(arr) / sizeof(int),1);//배열의 바이트 크기를 정수형 바이트 크기로 나눠 배열의 길이 계산

ShowArayElem(arr, sizeof(arr) / sizeof(int));

AddArayElem(arr, sizeof(arr) / sizeof(int),2);

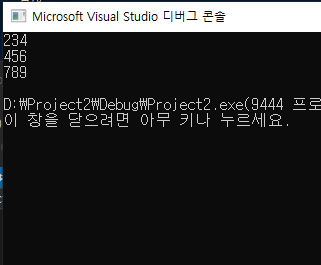
ShowArayElem(arr, sizeof(arr) / sizeof(int));

AddArayElem(arr, sizeof(arr) / sizeof(int),3);

ShowArayElem(arr, sizeof(arr) / sizeof(int));

return 0;

}



한줄 토의: 실습1의 문제를 함수 추가, 인자추가로 응용함

실습 3

#include <stdio.h>

void Swap(int n1, int n2)//인자 두개를 입력받아 서로 값을 바꾸는 함수 정의

{

int temp = n1;

n1 = n2;

n2 = temp;

printf("n1 n2: %d %d \n", n1, n2);

}

int main(void)

{

int num1 = 10;//변수 선언 및 초기화

int num2 = 20;//변수 선언 및 초기화

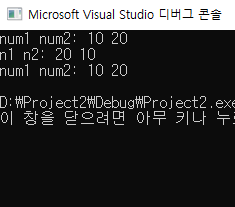
printf("num1 num2: %d %d \n", num1, num2);//출력문

Swap(num1, num2);//함수에 값 대입

printf("num1 num2: %d %d\n", num1, num2);//출력문

return 0;

}



한줄 토의: call-by-value는 함수에서 외부 변수 값의 변경이 불가하다.

실습4

#include <stdio.h>

void Swap(int \* ptr1, int \*ptr2)//인자 두개를 입력받아 서로 값을 바꾸는 함수 정의

{

int temp = \*ptr1;

\*ptr1 = \*ptr2;

\*ptr2 = temp;

}

int main(void)

{

int num1 = 10;//변수 선언 및 초기화

int num2 = 20;//변수 선언 및 초기화

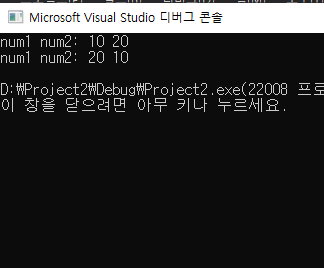
printf("num1 num2: %d %d \n", num1, num2);//출력문

Swap(&num1, &num2);//함수에 값 대입

printf("num1 num2: %d %d\n", num1, num2);//출력문

return 0;

}



한줄 토의: 인자를 포인터 변수로 받아 call-by-reference방식으로 함수내에서 외부 변수의 값 변경에 관여할 수 있다.

실습5

#include <stdio.h>

int main(void)

{

double PI = 3.1415;//double형 변수 선언 및 초기화

double rad;//double형 변수 선언

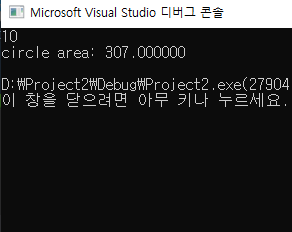
PI = 3.07;//실수로 잘못 삽입된 문장, 컴파일 시 발견 안됨

scanf("%lf", &rad);//반지름 입력받기

printf("circle area: %f \n", rad\*rad\*PI);//면적 출력

return 0;

}



한줄 토의: 실수로 삽입된 PI=3.07이라는 문장을 해결할 방법은 const선언을 통한 변수의 상수화!

실습 6

#include <stdio.h>

void ShowData(const int \* ptr)//ptr의 값 변경 불가

{

int \*rptr = ptr;//ptr을 가리키는 포인터 변수선언

printf("%d \n",\*rptr);

\*rptr = 20;//num값 변경

}

int main(void)

{

int num = 10;

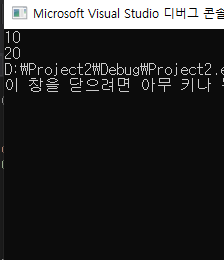
int \*ptr = &num;//num을 가리키는 포인터 변수 선언

ShowData(ptr);

printf("%d", num);//값이 변경되었는지 확인할 수 있는 출력문

return 0;

}



한줄 토의: ptr을 이용한 num의 값 변경은 const선언으로 해당 함수에서 불가능하지만 ptr을 가리키는 새로운 포인터 변수 rptr을 선언하여 바꿔줄 수 있기 때문에 ptr의 const선언은 의미없는 것

이 되어버렸다.

실습7

#include <stdio.h>

int SquareByValue(int num)//call-by-value방식

{

num = num \* num;

return num;

}

void SquareByReference(int \*ptr)//call-by-reference방식

{

\*ptr = (\*ptr)\*(\*ptr);

}

int main(void)

{

int num;

int \*ptr = &num;//num의 주소값을 갖는 포인터 변수 ptr

scanf("%d", &num);//값 입력받기

printf("정수 입력:%d \n", num);

printf("%d \n", SquareByValue(num));

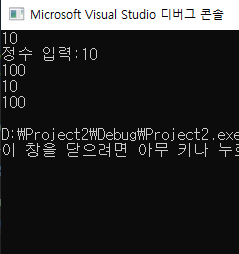
printf("%d \n", num);//call-by-value방식이므로 num값이 초기에 입력받은 그대로

SquareByReference(ptr);

printf("%d \n", num);//call-by-value방식이므로 num값이 함수내에서 변경됨

return 0;

}



한줄 토의: call-by-value와 call-by-reference방식의 차이를 알 수 있다.