2019202050 챕터 19 과제

실습1

#include <stdio.h>

void SimpleAdder(int n1, int n2)//함수 선언

{

printf("%d + %d = %d \n", n1, n2, n1 + n2);//매개변수 서로 더하고 출력

}

void ShowString(char \* str)//함수 선언

{

printf("%s\n", str);//매개변수 출력

}

int main()

{

char \*str = "Function Pointer";//포인터 선언및 초기화

int num1 = 10, num2 = 20;//변수 선언 및 초기화

void(\*fptr1)(int, int) = SimpleAdder;//함수를 가리키는 포인터 선언

void(\*fptr2)(char \*) = ShowString;//함수를 가리키는 포인터 선언

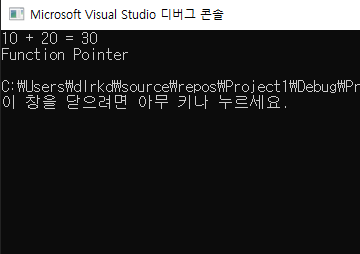
/\*함수 포인터 변수에 의한 호출\*/

fptr1(num1, num2);

fptr2(str);

return 0;

}



한줄 토의: 함수 포인터변수가 매개변수로 선언이 될 수 있다.

실습 2

#include <stdio.h>

int WhoIsFirst(int age1, int age2, int(\*cmp)(int n1, int n2))//함수선언

{

return cmp(age1, age2);//포인터로 매개변수 받기

}

int OlderFirst(int age1, int age2)//함수선언

{

/\*크기 비교연산\*/

if (age1 > age2)

return age1;

else if (age1 < age2)

return age2;

else

return 0;

}

int YoungerFirst(int age1, int age2)//함수선언

{

/\*크기 비교연산\*/

if (age1 < age2)

return age1;

else if (age1 > age2)

return age2;

else

return 0;

}

int main()

{

int age1 = 20;//변수 선언및 초기화

int age2 = 30;//변수 선언및 초기화

int first;//변수 선언

/\*출력문\*/

printf("입장순서 1 \n");

first = WhoIsFirst(age1, age2, OlderFirst);

printf("%d세와 %d세 중 %d세가 먼저 입장! \n\n", age1, age2, first);

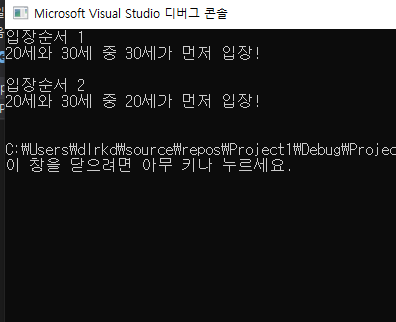
printf("입장순서 2 \n");

first = WhoIsFirst(age1, age2, YoungerFirst);

printf("%d세와 %d세 중 %d세가 먼저 입장! \n\n", age1, age2, first);

return 0;

}



한줄 토의: 함수 포인터를 활용하는 방법

실습2-2

#include <stdio.h>

int OlderFirst(int age1, int age2)//함수선언

{

/\*크기 비교연산\*/

if (age1 > age2)

return age1;

else if (age1 < age2)

return age2;

else

return 0;

}

int YoungerFirst(int age1, int age2)//함수선언

{

/\*크기 비교연산\*/

if (age1 < age2)

return age1;

else if (age1 > age2)

return age2;

else

return 0;

}

int main()

{

int age1 = 20;//변수 선언및 초기화

int age2 = 30;//변수 선언및 초기화

/\*출력문\*/

printf("입장순서 1 \n");

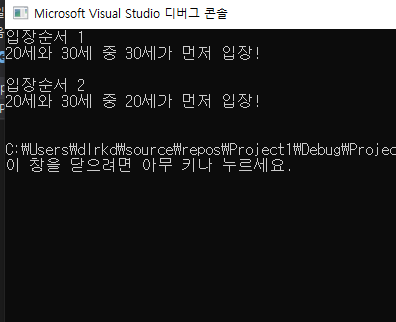
printf("%d세와 %d세 중 %d세가 먼저 입장! \n\n", age1, age2, OlderFirst(age1,age2));

printf("입장순서 2 \n");

printf("%d세와 %d세 중 %d세가 먼저 입장! \n\n", age1, age2, YoungerFirst(age1,age2));

return 0;

}



한줄 토의: 함수포인터를 쓰지 않을 경우 함수를 가리킬 수 없는 단점

실습 3

#include <stdio.h>

void SoSimpleFunc()

{

printf("I'm so simple");

}

int main()

{

int num = 20;

void\* ptr;//형이 없는 void포인터

ptr = &num;//변수 num의 주소 값 저장

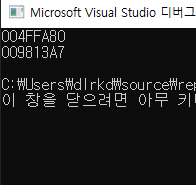
printf("%p \n", ptr);

ptr = SoSimpleFunc; //함수 SoSimpleFunc의 주소 값 저장

printf("%p \n", ptr);

return 0;

}



한줄 토의: 형 정보가 존재하지 않기에 주소값의 저장이 가능하다.

실습4

#include <stdio.h>

int main(int argc, char \*argv[])//이중포인터 사용

{

int i = 0;

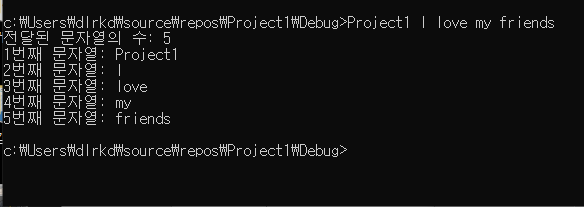
printf("전달된 문자열의 수: %d \n", argc);

for (i = 0; i < argc; i++)//전달된 문자열 갯수만큼 출력하는 반복문

printf("%d번째 문자열: %s \n", i + 1, argv[i]);

return 0;

}



한줄토의: main함수의 인자 전달에 관한 학습

char \* argv[] 관련 예제

#include <stdio.h>

/\*문자열을 인자로 받아 출력하는 함수\*/

void ShowAllString(int argc, char \*argv[])

{

int i;

for (i = 0; i < argc; i++)

printf("%s \n", argv[i]);

}

int main()

{

char \* str[3] = {//char형 포인터 배열

"C Programming",

"C++ Programming",

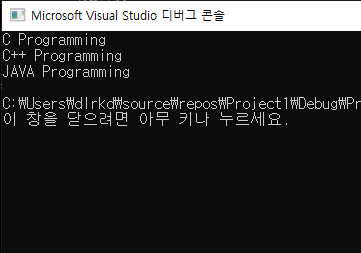
"JAVA Programming"

};

ShowAllString(3, str);//함수 실행

return 0;

}



한줄 토의: char \* argv[]를 통해 문자열을 출력해보는 예제

실습5

#include <stdio.h>

/\*문자열을 인자로 받아 출력하는 함수\*/

int main(int argc, char \*argv[])

{

int i = 0;

printf("전달된 문자열의 수:%d \n", argc);

while (argv[i] != NULL)//문자열이 없을 때까지 인자전달 반복

{

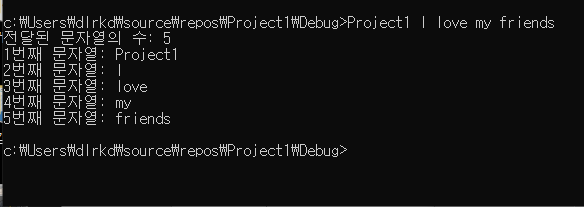
printf("%d번째 문자열:%s \n", i + 1, argv[i]);

i++;

}

return 0;

}



한줄 토의: 문자열의 개수의 측정과 그 구분을 위해 널 문자의 사용을 알게 되었고 main함수의 인자전달에 대해 알게 되었다.

실습6

#include <stdio.h>

#include <stdlib.h>//rand 함수가 정의된 헤더파일선언

int main()

{

int i;

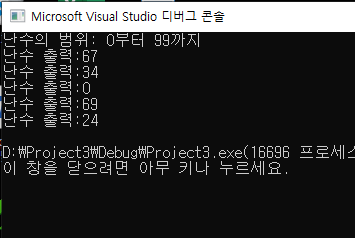
printf("난수의 범위: 0부터 %d까지 \n", (rand()%100));//rand()%n은 0~n-1까지의 난수범위

for (i = 0; i < 5; i++)

printf("난수 출력:%d \n", (rand()%100));

return 0;

}



한줄토의: rand,함수를 이용한 난수의 출력