|  |
| --- |
| 4일차 C언어 심화 워크샵 |

* **답안 제출은 A4용지에 수기로 작성하여 제출하시오.**

**[6과. 함수 등가포인터]**

1. **f1, f2, f3함수에 대응하는 등가포인터 p, q, r을 설계하시오. 함수 등가 포인터를 활용하여 원래 함수 실행과 동일한 결과가 나오도록 설계하시오.**

|  |  |
| --- | --- |
| **char \* f1( char \*a )**  **{**  **return ++a;**  **}**  **int f2(int a, int b)**  **{**  **return a+b;**  **}**  **int \* f3(void)**  **{**  **static int a[]={1,2,3,4};**  **return a;**  **}** | **void main(void)**  **{**  **// p, q, r선언**  **// p, q, r에대응함수대입**  **char \*(\*p)(char \*)=f1;**  **int (\*q)(int, int)=f2;**  **int \*(\*r)(void)=f3;**  **printf("%s\n", f1("hello") );**  **printf("%d\n", f2(3, 4) );**  **printf("%d\n", f3()[2] );**  **// p, q, r로 위처럼 실행**  **printf("%s\n", p("hello") );**  **printf("%d\n", q(3, 4) );**  **printf("%d\n", r()[2] );**  **}** |

1. **전달받은 함수를 받아 함수를 실행하는 func1, func2, func3함수를 설계하시오.(210p)**

|  |  |  |
| --- | --- | --- |
| **char \* f1( char \*a )**  **{**  **return a;**  **}**  **int f2(int a, int b)**  **{**  **return a+b;**  **}**  **f3(void)**  **{**  **static int a[]={1,2,3,4};**  **return &a;**  **}** | **void func1( p )**  **{**  **printf("%s\n", p("hello") );**  **}**  **void func2( p )**  **{**  **printf("%d\n", p(3,4) );**  **}**  **void func3( p )**  **{**  **printf("%d\n", p() );**  **}**  **void main(void)**  **{**  **func1( f1 );**  **func2( &f2);**  **func3( f3 );**  **}** | **[출력화면]**  **ello**  **7**  **3** |

**🡺 void func1(char\* (\*p)(char \*) ), +1**

**void func2( int(\*p)(int, int))**

**int (\*f3(void)[4], (\*p())[2]**

1. **배열 fa를 완성하고, fa로 add,sub,mul함수를 호출하는 코드를 완성하라.(213p)**

|  |  |
| --- | --- |
| **int add(int a, int b)**  **{**  **return a+b;**  **}**  **int sub(int a, int b)**  **{**  **return a-b;**  **}**  **intmul(int a, int b)**  **{**  **return a\*b;**  **}** | **fa[3] = {add, sub, mul};**  **void main(void)**  **{**  **inti;**  **int a = 3, b = 4;**  **for(i=0; i<sizeof(fa)/sizeof(fa[0]); i++)**  **{**  **printf("%d\n", );**  **}**  **}** |

**🡺Int(\*fa[3] )(int,int) = {add, sub, mul};**

**printf("%d\n", fa[i](a,b) );**

1. **배열 fa와 fa로 f1, f2함수를 호출하는 코드를 완성하시오.(213p)**

|  |  |  |
| --- | --- | --- |
| **char \* f1( char \*a )**  **{**  **return ++a;**  **}**  **char \* f2( void )**  **{**  **static char a[ ]=”hello”;**  **return a;**  **}** | **fa[2] ={ f1, f2 };**  **void main(void)**  **{**  **printf(“%s\n”, fa[0](“hello”) );**  **printf(“%s\n”, fa[1] );**  **}** | **[출력화면]**  **ello**  **ello** |

**🡺char \*(\*fa[2])( ) ={f1, f2}; / fa[1]()**

1. **함수를 리턴하는 func1, func2함수와 main의 코드를 완성하시오. (214p)**

|  |  |  |
| --- | --- | --- |
| **void f1( void )**  **{**  **printf("f1\n");**  **}**  **func1(void)**  **{**  **return f1;**  **}**  **int f2( int a, int b)**  **{**  **return a + b;**  **}** | **func2(void)**  **{**  **return f2;**  **}**  **void main(void)**  **{**  **func1**  **printf("%d\n", func2 );**  **}** | **[출력화면]**  **f1**  **8** |

**🡺**

**void (\* func1(void)) (void)**

**int (\*func2(void))(int, int)**

**func1()();**

**printf("%d\n", func2()(3,5) );**

1. **typedef를 사용하여FP2, FP1를 완성하고, f1실행을 통해 9를 인쇄하라(226p)**

|  |  |
| --- | --- |
| **typedef FP2**  **typedef FP1**  **FP2 f2(void)**  **{**  **static int a[3][4] = {1,2,3,4,5,6,7,8,9,10,11,12};**  **return a;**  **}** | **FP1 f1(void)**  **{**  **return &f2;**  **}**  **void main(void)**  **{**  **printf("%d\n", f1( ) );**  **}** |

**🡺**

**typedef int (\* FP2)[4];**

**typedef FP2 (\*FP1)( void);**

**f1()()[2][0]**

1. **int a[4][3]={1,5,3,4,2,6,7,11,9,10,8,12}; 배열에서 1열을 기준으로 행단위 오름차순 정렬하시오.**

**qsort함수를 이용하여 콜백함수를 작성하여 정렬하시오.**

**#include<stdio.h>**

**#include<stdlib.h>**

**int func(const void \*e1, const void \*e2)**

**{**

**const int \*p=e1, \*q=e2;**

**if( p[1] > q[1] ) return 1;**

**else if( p[1] < q[1] ) return -1;**

**return 0;**

**}**

**int main()**

**{**

**int a[4][3]={1,5,3,4,2,6,7,11,9,10,8,12};**

**int i,j;**

**qsort(a, 4, sizeof(a[0]), func);**

**for(i=0;i<4;i++)**

**{**

**for(j=0;j<3;j++)**

**{**

**printf("%d ", a[i][j]);**

**}**

**printf("\n");**

**}**

**}**

**[ 6.2과 typedef ]**

1. **typedef을 완성하시오.**

**(1) typedef ;**

**int \* p -> IP p**

**(2) typedef ;**

**int \* func() -> IP func();**

**(3) typedef ;**

**double a[10][25]; -> DAA a[10]**

**(1) int \*IP**

**(2) int \*IP**

**(3) double DAA[25]**

1. **int \*(\*(\*(\*x)())[10])();와 같은 함수의 선언을 다음과 같이 typedef을 사용하여 나눠 표현을 하였을 때 각 typedef을 완성하시오.**

**typedef ; //함수 포인터 FA1**

**typedef ; // FA1 10배열 포인터 FA2**

**FA2 (\*x)();**

**🡺 int \*(\*FA1)(), FA1 (\*FA2)[10]**

1. **union uni \*(\*x)[2]; 변수가 있을 때 다음 함수의 리턴 타입이 UPTR이 되도록 typedef를 완성하시오.**

**typedef ;**

**UPTR f(void)**

**{**

**return \*(x+1);**

**}**

**🡺 typedef union uni \*\*UPTR;**

1. **FP1에 대한 typedef를 완성하고 출력결과를 쓰시오.**

|  |  |
| --- | --- |
| **typedef FP1**  **FP1 func(void)**  **{**  **static int a[2][3] = {1,2,3,4,5,6};**  **return &a;**  **}** | **void main(void)**  **{**  **printf("%d", (\*(func()+1))[-3][4] );**  **}** |

**🡺 typedef int (\*FP1)[2][3]; 출력결과는 2**

1. **FP2에 대한 typedef를 완성하고 출력결과를 쓰시오.**

|  |  |
| --- | --- |
| **typedef FP2**  **FP2 func(void)**  **{**  **static char \*a[3] = {"good", "well", "nice"};**  **return &a;**  **}** | **void main(void)**  **{**  **printf("%s", func()[1][-2]+1);**  **}** |

**🡺 typedef char \*(\*FP2)[3]; 출력결과는 ell**

1. **FPTR에 대한 typedef를 완성하고 출력결과를 쓰시오.**

|  |  |
| --- | --- |
| **typedef FPTR**  **int \*f1(void)**  **{**  **static int a[4] = {1,2,3,4};**  **return a;**  **}**  **int \*f2(void)**  **{**  **static int a[4] = {10,20,30,40};**  **return a;**  **}**  **FPTR fa[2]= {f1, f2};**  **int f4(void)**  **{**  **return 1;**  **}** | **FPTR (\*func1(void))**  **{**  **return fa;**  **}**  **FPTR func2(int (\*p)(void))**  **{**  **return func1()[p()];**  **}**  **void main(void)**  **{**  **printf("%d\n",func2(f4)( )[3] );**  **}** |

**🡺 typedef int \*(\*FPTR)(); 출력결과: 40**

**[ 7.1과 포인터 활용 정리]**

1. **문자열 교환 코드를 Swap()함수로 작성하시오.**

|  |  |
| --- | --- |
| **void Swap( ap , bp )**  **{ // temp의 선언**  **temp ;**  **// 문자열 교환코드 작성**  **;**  **;**  **;**  **}** | **void main(void)**  **{**  **char \*a = "hi";**  **char \*b = "hello";**  **printf(" Before : %s, %s ", a, b );**  **Swap( a , b );**  **printf(" After : %s, %s ", a, b );**  **}** |

**🡺 void Swap(char \*\*ap, char \*\*bp)**

**🡺 char \*temp;**

**🡺 temp=\*ap; \*ap=\*bp; \*bp=temp;**

**🡺 Swap(&a, &b);**

1. **배열 a의 문자열을 오름차순으로 정렬하는 Sort함수를 구현하시오.**

**(추가문제=> qsort함수를 이용하여 콜백함수 설계하여 정렬하시오)**

|  |  |
| --- | --- |
| **void Sort( p )**  **{**  **// 오름차순으로 정렬**  **}** | **void main()**  **{**  **int i;**  **char \*a[5] = {"kind", "cool", "good", "well", "best"};**  **Sort( a );**  **for(i=0; i<5; i++) printf("%s\n", a[i]);**  **}** |

**🡺** **void Sort( char \*\*p )**

**{**

**int i,j;**

**char \*temp;**

**for(i=0;i<5-1;i++)**

**{**

**for(j=0;j<5-1-i;j++)**

**{**

**if( strcmp(p[j], p[j+1]) ==1 )**

**{**

**temp=p[j+1];**

**p[j+1]=p[j];**

**p[j]=temp;**

**}**

**}**

**}**

**}**

**[ 7.2과 힙과 동적배열 ]**

1. **a와 같은 형태의 다차원 배열을 만들고자 할 때 포인터 변수 p의 선언과 동적 할당 함수의 인자 내용을 작성하시오.(266p)**

|  |  |
| --- | --- |
| **[1] int a[5][3];**  **[2] short a[2][7];**  **[3] float a[3][6][7];**  **[4] char (\*a[7]) [6][3];** | **[1] int (\* p) [3];**  **p = (int (\*)[3])malloc( 5 \* sizeof(int[3]) );**  **[2] short (\*p)[7];**  **p = (short(\*)[7])malloc( 2 \* sizeof(short[7] );**  **[3] float (\*p)[6][7];**  **p = (float (\*)[6][7])malloc( 3 \* sizeof(float [6][7]) );**  **[4] char (\*\*p)[6][3] ;**  **p = (char (\*\*)[6][3])malloc(7\*sizeof(char (\*)[6][3]));** |

1. **다음의 코드를 완성시켜 보자. (266p)**

|  |
| --- |
| **float add(int a, int b)**  **{**  **return (float)(a+b);**  **}**  **void main(void)**  **{**  **p ; // [1] p변수 선언**  **p = ; // [2] 동적할당**  **p[2] = add;**  **printf("%f\n", p[2](3,4) ) ; // [3]p의 2번 요소의 함수 호출**  **}** |

**[1] p변수는 int 데이터 2개를 parameter로 하고 float 데이터를 리턴하는 함수의 주소를 저장하는 동적할당 배열의 주소를 저장할 변수이다. p변수를 선언하시오.**

**🡺float (\*\*p)(int,int) ;**

**[2] p변수에 위에서 설명한 함수의 주소를 4개 저장할 수 있는 배열의 공간을 동적할당 받아 저장하는 코드를 작성하시오.**

**🡺 p = (float (\*\*)(int,int))malloc(4 \* sizeof(float (\*)(int,int)));**

**[3] p변수 2번 요소에 저장된 주소의 함수를 호출하는 코드를 작성하시오. (함수 호출 시 3,4 라는 인자를 전달하시오.)**

**🡺p[2](3,4)**

**[ 7.3과 메모리직접 액세스 ]**

1. **다음의 목적을 달성할 수 있는 코드를 완성하시오.(267p)**
2. **0x1000 번지로부터 2바이트 메모리 공간에 0xabcd 값을 저장시키시오.**

* **\*(unsigned short\*)0x1000 = 0xabcd; or \*(short\*)0x1000 = 0xabcd;**

1. **0x2000 번지로부터 4바이트 메모리 공간에 10.23f 값을 저장시키시오.**

|  |  |
| --- | --- |
| * **\*(float\*)0x2000 = 10.23f;** |  |

1. **0x3000번지에 저장된 4바이트 값의 10번 비트를 1로 바꾸어 저장시키시오.**

**(해당 주소는 H/W주변장치 임)**

* **\*(volatile unsigned int\*)0x3000 |= (1<<10);**

**[4] 0x4000 번지에 저장된 4바이트 값의 7번 비트를 0으로 바꾸어 저장시키시오.**

**(해당 주소는 H/W주변장치 임)**

* **\*(volatile unsigned int\*)0x4000 &= ~(1<<7);**