문항 선택
1 2 3 4 5 6
전 문제씩 검토
검토 완료
강의실 홈
강의정보
• 교수계획표 (국문)
• 교수계획표 (영문)
성적/출석관리
• 동영상수누현황
• 스마트출석부
• 성적부
수강생 알림
• 쪽지 보내기
기타 관리
학습활동

시작 일시	2022-11-12 16:12
진행 상황	종료됨
완료 일시	2022-11-19 15:11
소요시간	6 일 22 시간
성적	최고 7.00점 중 5.92점 (85%)

문제 1

부분적으로 맞음
총 1.00 점에서 0.92 점 얻당
문제 표시

Consider the following source code and the execution result.
Discover the attributes of variables "x" and "px" and fill in the blanks.
For the Value attribute, enter the value of each variable in the "return" statement.

Identifier	x	px
Value	3.14 ✓	0x0061FF28 ✓
Type	double ✓	double * ✓
Memory Size (in bytes)	8 ✓	8 ✗
Scope	local ✓	local ✓
Lifetime	automatic ✓	automatic ✓
Address	0x 0061FF28 ✓	0x 0061FF24 ✓

Execution Result

```
ADDR:0061FF28
ADDR:0061FF24
```

```
#include <stdio.h>
int main(void)
{
    double x=3.14, *px = &x;

    printf("ADDR:%p\n", &x);
    printf("ADDR:%p\n", &px);

    return 0;
}
```

문제 2

정답
총 1.00 점에서 1.00 점 얻당
문제 표시

For the source code below, the output result of the printf() at the line // 0) is like the followings.

The output result of the printf() at the line // 0)

`0061FF2C, 0061FF24, 0061FF28`

The following table illustrates the memory spaces of variables n2, pn, and n1 whose addresses are 0x0061FF24, 0x0061FF28, and 0x0061FF2C respectively.
After the execution of the statements // 0), // 1), // 2), and // 3), fill in the blanks with the appropriate value stored in each variable.
If the value is for an address, write it in the Hexa format.

Address	0x0061FF24 - n2	0x0061FF28 - pn	0x0061FF2C - n1
// 0)	0 ✓	0x0061FF2C ✓	3 ✓
// 1)	3 ✓	0x0061FF2C ✓	3 ✓
// 2)	3 ✓	0x0061FF2C ✓	4 ✓
// 3)	3 ✓	0x0061FF2C ✓	7 ✓

Write the output result of the printf() at the line // 4).

The output result of the printf() at the line // 4)

7 ✓ , 3 ✓ , 7 ✓

```
#include <stdio.h>
int main(void)
{
    int n1=3, *pn = &n1;
    int n2=0;

    printf("%p, %p, %p\n", &n1, &n2, &pn);   // 0)
    n2 = *pn;                                // 1)
    *pn = n2 + 1;                            // 2)
    n1 = *pn + *(&n2);                       // 3)
    printf("%d, %d, %d\n",n1,n2,*pn);        // 4)
    return 0;
}
```

문제 3

정답
총 1.00 점에서 1.00 점 얻당
문제 표시

We want to build a program that gives the following result after the execution.
Complete the increase() procedure and its calling code in the main() function by filling in the blanks with the appropriate codes.

[1] = int* ✓

[2] = *x ✓

[3] = *x ✓

[4] = &a ✓

Execution Result Example 1	Execution Result Example 2
3 ++a == 4	7 ++a == 8

```
#include <stdio.h>
void increase([1] x)
{
    [2]=[3]+1;
}
int main(void)
{
    int a;

    scanf("%d", &a);
    increase([4]);
    printf("++a == %d\n", a);
    return 0;
}
```

문제 4

정답
총 1.50 점에서 1.50 점 얻당
문제 표시

Fill in the blanks in the following execution result of the given program.

Execution Result

a: 1, pa: 0061FF28
a: 2 ✓ , pa: 0061FF28 ✓
a: 2 ✓ , pa: 0061FF2C ✓
a: 4 ✓ , pa: 0061FF28 ✓

```
#include <stdio.h>

int main(void)
{
    int a = 1, *pa = &a;

    printf("a: %d, pa: %p\n", a, pa);
    ++*pa;
    printf("a: %d, pa: %p\n", a, pa);
    *pa++;
    printf("a: %d, pa: %p\n", a, pa);
    a += *--pa;
    printf("a: %d, pa: %p\n", a, pa);

    return 0;
}
```

문제 5

부분적으로 맞음
총 1.50 점에서 0.50 점 얻당
문제 표시

The following two source codes show two general methods for array traversal. That is, the left source code shows the index-based array traversal and the right one shows the pointer-based array traversal.
The execution results of the two codes should be the same.

Fill in the blanks of the right source code to complete the pointer-based array traversal.

[1] pstr=&str pstr+str ✗

[2] *pstr ✓

[3] *pstr++ ✗

```
#include <stdio.h>
int main(void)
{
    int i;                  // index
    int num=0, sum=0; // number
    char str[]="123,456,789";

    for(i=0; str[i] != '\0'; i++) {
        if (str[i] == ',') { // new num
            sum += num;
            num = 0;
        } else { // a digit
            num = num*10 + (str[i]-'0');
        }
    }
    sum += num;
    printf("Sum of all values in ");
    printf(" CSV(%s) : %d", str,sum);

    return 0;
}
```

```
#include <stdio.h>
int main(void)
{
    char *pstr;                 // pointer
    int num=0, sum=0;           // number
    char str[]="123,456,789";

    for( [1]; [2] != '\0'; [3] ) {
        if (*pstr == ',') { // new num
            sum += num;
            num = 0;
        } else { // a digit
            num = num*10 + (*pstr-'0');
        }
    }
    sum += num;
    printf("Sum of all values in ");
    printf(" CSV(%s) : %d", str,sum);

    return 0;
}
```

문제 6

정답
총 1.00 점에서 1.00 점 얻당
문제 표시

Fill in the blanks in the following execution result of the given program.

Execution Result

foo1(str1) = 9 ✓ , foo2(str1) = 8 ✓
foo1(str2) = 5 ✓ , foo2(str2) = 4 ✓

```
#include <stdio.h>

int foo1(char *pstr) {
    char *pcur = pstr;
    while (*pcur++)
        ;
    return (pcur-pstr);
}

int foo2(char *pstr) {
    char *pcur = pstr;
    while (*pcur)
        pcur++;
    return (pcur-pstr);
}

int main(void) {
    char str1[] = "12345678";
    char str2[] = "1234";

    printf("foo1(str1) = %d, foo2(str1) = %d\n", foo1(str1), foo2(str1));
    printf("foo1(str2) = %d, foo2(str2) = %d\n", foo1(str2), foo2(str2));

    return 0;
}
```

검토 완료