🏠 C프로그래밍    **[Final-A:110]**

| | |
|---|---|
| **시작 일시** | 2018-12-18, 18:05:19 |
| **진행 상황** | 종료됨 |
| **완료 일시** | 2018-12-18, 18:57:33 |
| **소요시간** | 52 분 14 초 |
| **성적** | 최고 110.00점 중 **84.78**점 (**77**%) |

**정보**

The total exam time is 200 minutes.

- You should take the Final-A first.
  - During the Final-A, you cannot use any other program including the DEV-C++ except the PLMS browser itself.
- After finishing the Final-A, raise your hand to request the T.A for the password to start the Final-B. That is, you cannot start Final-B without finishing the Final-A.
  - You should use the DEV-C++ to solve the Final-B. Other programs are not allowed to use in default. But If it is necessary for you to use other IDEs or compiler programs, ask the T.A for permission.

The total points of the final exam is 250.

The following table shows points assigned to each question.

**Final-A (Short Answer Questions - 110 points,**

22 Questions, 5 points for each question

**Final-B (Essay(Programming) Questions - 140 points**

| Q # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| pts | 5 | 7 | 8 | 15 | 15 | 20 | 20 | 20 | 30 |

**정보**

To help you in managing your exam time, this page includes some previews of Final-B problems.

1. (5) The lower two source codes differ only in the srand() line. Describe/explain the role of srand(). You should also explain why time(0) is used in calling srand()
2. (7) When we execute the following program, the first value 123 of the CSV string is repeatedly extracted and the program does not terminate. Describe the reason and how to fix it to print out all values in the CSV string.
3. (8) In the source code below, we can get the same execution result from "stud_printx()" and "stud_print()" while they are clearly different functions in their argument type. Explain the differences between the 2 functions.
   - Compare the amount of memory copy when they are called.
   - Compare the possibility of unwanted side effect.
4. (15) Write a program that does the followings.
   - Find the minimum integer $n$ such that the sum of $1 + 2 + ... + n$ is no less than 1000. That is, $1 + 2 + ... + n \geq 1000$.
   - Print out "the sum" ($\geq 1000$) and $n$.
5. (15) A leap year is a calendar year containing one additional day added to keep the calendar year synchronized with the astronomical year. You can determine whether a year is a leap year or a common year using the following rules. Complete the isleapyear() function in the source code below which determines whether a year is a leap year (return true/1) or not (return false/0).
6. (20) The following program randomly generates 10 3-digit numbers and prints them out. All 3 digits of a number must be different. so 112, 887 are wrong values because of duplicate. The first digit can't be 0, so 98 or 34 are wrong values. Complete the function generate_target_number() by using generate_a_digit() function.
7. (20) strstr() is a C standard library function defined in <string.h>. It finds the first occurrence of the null-terminated byte string pointed to by substr in the null-terminated byte string pointed to by str. Write mystrstr() function that is equivalent to the "strstr()"
8. (20) Complete the my_atof() in the source code below that converts a string into a corresponding floating-point value.
   - You are not allowed to use any function in your my_atof().
   - Use "pointer based traversal" instead of "array index based traversal" in implementing my_atoi(). A little penalty will be imposed if you use "array index based traversal".
   - You can safely assume that the input string is terminated by '\0' (null character).
   - You can safely assume that the input string consists of only digits and no more than one ".". Ex) 123.23, 3465, 27.08, 0.5
9. (30) The following code sorts the array of STUD in points descending order. Complete the function find_max_stud() and sort_stud(). You are not allowed to add a new function nor to include a new header file.

**문제 1**
부분적으로 맞음
총 5.00 점에서
2.50 점 할당

For each type of programming error in the followings, determine whether it occurs in the source code below. If an error of the corresponding type occurs, select O, otherwise select X.

- Syntax Error : [ X ▼ ] ✓
- Logical Error : [ X ▼ ] ✗
- Runtime Error : [ X ▼ ] ✓
- No Error : [ O ▼ ] ✗

```c
#include <stdio.h>
int main(void)
{
    int x[3] = {1,2,3};
    int y[3] = {1,2,3};

    printf("the values of x, y is ");
    if( x == y )
        printf("equal.\n");
    else
        printf("not equal.\n");
    return 0;
}
```

**문제 2**

부분적으로 맞음

총 5.00 점에서
4.00 점 할당

For each printf() in the source code below, determine whether it is called or not.

```c
#include <stdio.h>
int main(void)
{
    int a=1,b=2,c=3;

    if (a+b==c)
        printf("a+b==c\n");    //   [called ▼]  ✓

    if (c=a+b)
        printf("c=a+b\n");     //   [called ▼]  ✓

    if (b-a++)
        printf("c=b-a++\n");   //   [called ▼]  ✓

    if (a && ~b)
        printf("a && ~b\n");   //   [called ▼]  ✓

    if (!a & c)
        printf("!a & c\n");    //   [called ▼]  ✗

    return 0;
}
```
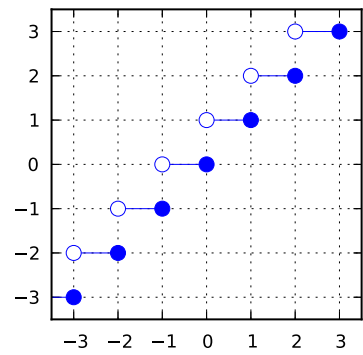
**문제 3**

정답

총 5.00 점에서
5.00 점 할당

Complete a program that outputs the result of each die throw (randomly with an equal probability of one of {1, 2, 3, 4, 5, 6})

double ceil(double arg) : Computes the smallest integer value not less than arg.



Choose a proper code for [1]  [1 ▼]  ✓

1. (int)(ceil(urand()*6))
2. ceil(urand()*6)-1.0
3. (int)(ceil(urand())*6)
4. (int)(urand()*6)

```c
#include <stdio.h>
double urand(void)
{
    return (double)rand()/(RAND_MAX+1);
}

int main(void)
{
    int i, diceval;

    srand(time(0));
    for (i=0; i<N_DICE; i=i+1) {
        diceval = [1];
        printf("%dth Dice Value : %d\n", i, diceval);
    }

    return 0;
}
```

**문제 4**

틀림

총 5.00 점에서
0.00 점 할당

Complete the source code below to get the following execution result.  [2 ▼]  ✗

|   | [1] | [2] | [3] |
|---|---|---|---|
| 1 | '0' | "abc" | "%c\n" |
| 2 | '\0' | "abc\0" | "%s\n" |
| 3 | 0 | "abc" | "%s\n" |
| 4 | '\0' | "abc0" | "%c\n" |

| Execution Result |
|---|
| abc |
| abc |

```c
#include <stdio.h>
int main(void)
{
    char str1[4] = { 'a', 'b', 'c', [1]};
    char str2[4] = [2];

    printf([3], str1 );
    printf([3], str2 );
    return 0;
}
```

**문제 5**
정답
총 5.00 점에서
5.00 점 할당

Recall the **"Selection Sort"** algorithm introduced in the lecture 11. This algorithm takes "n-1" steps if the number of items to sort is n.

We will apply this algorithm to sort the following sequence of integers in the ascending order.

Write the intermediate results how the sequence is changed after each step.

DO NOT include white space characters in your answer.

- (Initial state) 5,3,1,4,2
- (1) `1,3,5,4,2` ✓
- (2) `1,2,5,4,3` ✓
- (3) `1,2,3,4,5` ✓
- (4) 1,2,3,4,5

**문제 6**
부분적으로 맞음
총 5.00 점에서
3.33 점 할당

Complete the source code below.

We want that the program copies all values in the array na[100] to the memory block pointed by p_nacopy.

```c
#include <stdio.h>
#include <stdlib.h>
#include <assert.h>

int main(void)
{
        int na[100], i;
        int *p_nacopy = NULL;

        for (i=0; i<100; i++)
                na[i] = i;

        p_nacopy = malloc( sizeof(int)*100 ✓ );  //[1] - choose a proper one

        assert ✓ (p_nacopy);    //[2] - write a proper macro/function name
        /* The macro/function is used to checks if its argument (which must have scalar type) compares equal to zero. If it does, it outputs implementation-specific diagnostic information on the standard error output and calls abort(). */

        /* for block
        for (i=0; i<100; i++)
            p_nacopy[i] = na[i];
         */
        // In the next statement, we want to do the same work with
        // the above "for block" if it is not commented.
        strcpy ✗ (p_nacopy, na, [1]); //[3] - write a proper function name

        free(p_nacopy);

        return 0;
}
```

**문제 7**
부분적으로 맞음
총 5.00 점에서
3.00 점 할당

For the following variables and memory blocks used in the source code below, classify its lifetime.

| Variable | Declared Line | Lifetime |
|---|---|---|
| sum1 | line 5 | static ✓ |
| count | line 8 | static ✓ |
| memory block pointed by pns_copy | line 16 | Automatic ✗ |
| pns_copy | line 16 | Dynamic ✗ |
| sum2 | line 17 | Automatic ✓ |

```c
1    #include <stdio.h>
2    #include <stdlib.h>
3    #include <assert.h>
4
5    int sum1 = 0;
6
7    int add(int num) {
8        static int count = 0;
9        count ++;
10       sum1 += num;
11       return count;
12   }
13
14   int main(void) {
15       int num[] = {1, 2, 3, 4, 5};
16       int *pns_copy = malloc(sizeof(num));
17       int i, sum2;
18
19       assert(pns_copy);
20       for (i=0; i<5; i++) {
21           pns_copy[i] = num[i];
22           sum2 = add(num[i]);
23       }
24       printf("sum1 is %d\n", sum1);
25       printf("sum2 is %d\n", sum2);
26
27       free(pns_copy);
28
29       return 0;
30   }
```

**문제 8**

정답

총 5.00 점에서
5.00 점 할당

Complete the source code below to get the following execution result.

DO NOT include inessential white space characters.

| Execution Result Example 1 | Execution Result Example 2 |
|---|---|
| 1<br>++a == 2 | 7<br>++a == 8 |

```c
#include <stdio.h>
void increase( int*  ✓   x)
{
    (*x)  ✓ ++;
}
int main(void)
{
    int a;

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

**문제 9**

정답

총 5.00 점에서
5.00 점 할당

What is the output of the following program ?

| Execution Result |
|---|
| (a, b)-(3,7), (a,b)-( 3 ✓ , 7 ✓ ), (a,b)-( 3 ✓ , 7 ✓ ), (a,b)-( 7 ✓ , 3 ✓ ) |

```c
#include <stdio.h>
void swap1(int a, int b)
{
    int t = a;
    a = b;
    b = t;
}
void swap2(int *a, int *b)
{
    int *t = a;
    a = b;
    b = t;
}
void swap3(int *a, int *b)
{
    int t = *a;
    *a = *b;
    *b = t;
}

int main(void)
{
    int a=3, b=7, t;

    printf("(a,b)-(%d,%d), ",a,b);

    swap1(a,b);
    printf("(a,b)-(%d,%d), ",a,b);

    swap2(&a,&b);
    printf("(a,b)-(%d,%d), ",a,b);

    swap3(&a,&b);
    printf("(a,b)-(%d,%d)",a,b);

    return 0;
}
```

**문제 10**

정답

총 5.00 점에서
5.00 점 할당

Complete the source code below to get the following execution result.
DO NOT include inessential white space characters in your answer.

**Execution Result**

```
Sum of all values in CSV[123,456,789] : 1368
```

```c
#include <stdio.h>
int main(void)
{
    char *pcsv;
    int num=0, sum=0;
    char csv[]="123,456,789";

    for( pcsv= csv ✓ ; *pcsv ✓ ; pcsv++)    {
        if (',' == *pcsv) {
            sum += num;
            num = 0;
        } else {
            num = num*10 + *pcsv-'0' ▼ ✓ ;
        }
    }
    sum += num;

    printf("Sum of all values in ");
    printf(" CSV[%s] : %d", csv,sum);

    return 0;
}
```

**문제 11**

부분적으로 맞음

총 5.00 점에서
3.75 점 할당

We want to complete the sum() function in the source code below. Choose all proper argument declarations in the followings for the blank [1].

- double p[4]     X ▼ ✗
- double * p      O ▼ ✓
- double (*p)[3]  X ▼ ✓
- double * p[]    X ▼ ✓

```c
#include <stdio.h>
double sum([1])
{
    return p[0] + p[1] + p[2];
}

int main(void)
{
    double a[]={1.2, 2.3, 3.4};

    printf("sum == %f\n", sum(a) );
    return 0;
}
```

**문제 12**

정답

총 5.00 점에서
5.00 점 할당

We want to get the following execution result by completing the source code below.
Choose all proper expressions for the blank [1].

- &*(na+1)   O ▼ ✓
- &na+1      X ▼ ✓
- &na[1]     O ▼ ✓
- na+1       O ▼ ✓

**Execution Result**

```
Enter an integer : 5
Enter an integer : 3
5
3
```

```c
#include <stdio.h>
int main(void)
{
    int na[2];

    printf("Enter an integer : ");
    scanf("%d", na    );
    printf("Enter an integer : ");
    scanf("%d", [1]);

    printf("%d\n", na[0] );
    printf("%d\n", na[1] );
}
```

**문제 13**

부분적으로 맞음

총 5.00 점에서
2.50 점 할당

What is the output of the following program?

| Execution Result |
|---|
| 4 ✗ |
| 0 ✓ |

```c
#include <stdio.h>
int foo(char *s, char *t)
{
    char *p1, *p2;

    for (p1 = s; *p1; p1++) {
        for (p2 = t; *p2; p2++)
            if (*p1 == *p2) break;
        if (*p2 == '\0') break;
    }
    return p1 - s;
}

int main(void)
{
    printf("%d\n", foo("abcd","babc"));
    printf("%d\n", foo("abcd","bcd"));

    return 0;
}
```

**문제 14**

부분적으로 맞음

총 5.00 점에서
4.09 점 할당

What is the output of the following program ?

| Execution Result |
|---|
| a ✓ |
| b ✓ |
| c ✓ |
| a = abc ✓ |
| a ✓ |
| b ✓ |
| c ✓ |
| a = bcd ✗ |

```c
#include <stdio.h>
int main(void)
{
    char a[] = "abc";
    char *p;
    int i;

    p = a;
    for (i = 0; i <3; ++i)
        printf("%c\n", *p++);
    printf("a = %s\n", a);

    p = a;
    for (i = 0; i < 3; ++i)
        printf("%c\n", (*p)++);
    printf("a = %s\n", a);

    return 0;
}
```

**문제 15**

부분적으로 맞음

총 5.00 점에서
3.75 점 할당

What is the output of the following program ?

| **Execution Result** |
|---|

```
n[0] = 1, p0.x = 1
n[0] = [ 2 ] ✓, p0.x = [ 1 ] ✓
n[0] = [ 2 ] ✗, p0.x = [ 2 ] ✓
```

```c
#include <stdio.h>
typedef struct st_point {
        int x;
        int y;
} tpoint;

void array_foo1(int a[]) {
        a[0] = a[0] + 1;
}
void array_foo2(int *a) {
        a[0] = a[0] + 1;
}

void struct_foo1(tpoint p) {
        p.x = p.x + 1;
}
void struct_foo2(tpoint *pp) {
        pp->x = pp->x + 1;
}

int main(void)
{
        int n[] = {1,1};
        tpoint p0 = {1,1};

        printf("n[0] = %d, p0.x = %d\n", n[0], p0.x);

        array_foo1(n);
        struct_foo1(p0);

        printf("n[0] = %d, p0.x = %d\n", n[0], p0.x);

        array_foo2(n);
        struct_foo2(&p0);

        printf("n[0] = %d, p0.x = %d\n", n[0], p0.x);

        return 0;
}
```

**문제 16**

정답

총 5.00 점에서
5.00 점 할당

Complete the source code below to get the following execution result.

| **Execution Result** |
|---|

```
[1:Choi] = 9.900000
[2:Park] = 0.100000
[3:Kim] = 5.000000
[1:Choi] = 9.900000
[2:Park] = 0.100000
[3:Kim] = 5.000000
```

Choose a proper one from the followings [ 1 ▼] ✓

| | **[1]** | **[2]** | **[3]** | **[4]** |
|---|---|---|---|---|
| 1 | i<3 | &pnuecs[i++] | ++ps | pnuecs+3 |
| 2 | i<3 | &pnuecs[++i] | ps++ | pnuecs+3 |
| 3 | i<3 | &pnuecs[i++] | ps++ | pnuecs+3 |
| 4 | i<=3 | &pnuecs[++i] | ++ps | pnuecs+4 |

```c
#include <stdio.h>
typedef struct student {
    int id;
    char *pname;
    double points;
} STUD;

void stud_print(STUD *ps) {
    printf("[%d:%s] = %lf\n",
    ps->id, ps->pname, ps->points);
}

int main(void)
{
    STUD pnuecs[3]={ {1, "Choi", 9.9},
        {2, "Park", 0.1}, {3, "Kim", 5.0 } };
    STUD *ps = pnuecs;
    int i=0;

    while ([1])
        stud_print([2]);
    do {
        stud_print(ps);
    } while ([3] < [4]);

    return 0;
}
```

**문제 17**

부분적으로 맞음

총 5.00 점에서
2.50 점 할당

By executing the following program, we got the following execution result. Besides, we successfully create 2 new files, "values.txt" and "values.bin".
Write the size of "values.txt" and "values.bin".

- size of "values.txt" : 41 ✗ bytes

- size of "values.bin" : 40 ✓ bytes

Note that we use **"f_open" instead of "fopen"** because PLMS does not allow to use "fopen" in source code due to its security policy. Thank you for your understandings.

| Execution Result |
|---|
| sizeof(float) = 4<br>1.20,2.30,3.40,4.50,5.60,1000.20,2000.30,3000.40,4000.50,5000.60, |

```c
#include <stdio.h>
int main(void)
{
    FILE *fptxt, *fpbin;

    float vals[10] = {1.2, 2.3, 3.4, 4.5, 5.6,
                1000.2, 2000.3, 3000.4,
                4000.5, 5000.6};
    int i;

    fptxt = f_open("values.txt","w");
    fpbin = f_open("values.bin","wb");

    printf("sizeof(float) = %d\n", sizeof(float));
    for (i=0; i<10; i++) {
        printf("%.2f,",vals[i]);
        fprintf(fptxt,"%.2f,",vals[i]);
    }
    fclose(fptxt);

    fwrite(vals, sizeof(float), 10, fpbin);
    fclose(fpbin);

    return 0;
}
```

**문제 18**

정답

총 5.00 점에서
5.00 점 할당

What is the output of the following program? Fill in each blank with an appropriate value.

| Execution Result |
|---|
| ADDR:0x0062FEA8<br>ADDR:0x0062FEA4<br>Size 4 |

pn-1: 0x0062FEA0 ✓

*pn-1: -1 ✓

(*(&nx))++: 36 ✓

*(pn+1)+1: 38 ✓

```c
#include <stdio.h>
int main(void)
{
    int nx=36;
    int ny=0;
    int* pn = &ny;

    printf("ADDR:%p\n", &nx);
    printf("ADDR:%p\n", &ny);
    printf("Size %d\n", sizeof(pn));

    printf("pn-1: %p\n", pn-1);
    printf("*pn-1: %d\n", *pn-1);
    printf("(*(&nx))++: %d\n", (*(&nx))++);
    printf("*(pn+1)+1: %d\n", *(pn+1)+1);

    return 0;
}
```

**문제 19**

부분적으로 맞음

총 5.00 점에서
2.50 점 할당

Complete the source code below to get the following execution result.

In the first blank, declare p which is an integer pointer array of size 3.

In the second blank, initialize p properly.

DO NOT include inessential white space characters.

| Execution Result |
|---|
| 11<br>33<br>22 |

```
#include <stdio.h>
int main(void)
{
    int x=11, y=22, z=33, i;

    p[3]  ✗  =  {&x,&z,&y}  ✓ ;

    for (i=0; i<3; i++)
        printf("%d\n", *p[i]);

    return 0;
}
```

**문제 20**

정답

총 5.00 점에서
5.00 점 할당

What is the output of the following program ?

| Execution Result |
|---|
| 11 ✓ , 17 ✓ , 23 ✓ , 27 ✓ , 37 ✓ |

```
#include <stdio.h>
int main(void)
{
    int na[] = {37, 23, 11, 27, 17};
    int nlen = sizeof(na)/sizeof(int);
    int i,j,t;

    for (i=nlen-1; i > 0; i--) {
        for (j=0; j<i; j++)
            if (na[j] > na[j+1]) {
                t=na[j];
                na[j] = na[j+1];
                na[j+1] = t;
            }
    }
    printf("%d, %d, %d, %d, %d\n",
        na[0], na[1], na[2], na[3], na[4]);
    return 0;
}
```

**문제 21**

정답

총 5.00 점에서
5.00 점 할당

Complete the source code below to get the following execution result.

DO NOT include inessential white space characters.

| Execution Result |
|---|
| 6 5 4 9 |

```
#include <stdio.h>
#define NSIZE 3
int main(void)
{
        int A[NSIZE][NSIZE]={ {1,2,3},
                              {4,5,6},
                              {7,8,9} };
        int temp;
        int *ptr = A, *curptr;

        for(curptr=ptr; curptr < ptr+9; curptr+=3) {
                temp = *curptr;
                *curptr = *(curptr+2);
                *(curptr+2) = temp;
        }

        for(curptr= ptr+3 ✓ ; curptr < ptr+7 ✓ ; curptr++)
            printf("%d ", *curptr);
        printf("\n");

        return 0;
}
```

**문제 22**

부분적으로 맞음

총 5.00 점에서
2.86 점 할당

For each blue code line in the source code below, answer whether it raises a syntax error.

```
#include <stdio.h>
typedef struct student {
    int id;
    char *pname;
    double points;
} STUD;

int main(void)
{
        STUD pnuecs[] = {{1, "Choi", 9.9},
                         {2, "Park", 0.1},
                         {3, "Kim", 5.0 }};
        STUD *p_std, s1, stdarray[3];
        int i;

        i = (pnuecs+1).id;      //   [OK ▼]  ✗

        i = &(pnuecs[1])->id;   //   [OK ▼]  ✗

        s1 = pnuecs[0];         //   [OK ▼]  ✓
        s1 = *(pnuecs+2);       //   [OK ▼]  ✓

        p_std = &pnuecs;        //   [Error ▼]  ✗
        p_std = pnuecs[1];      //   [Error ▼]  ✓

        stdarray = pnuecs;      //   [Error ▼]  ✓

        return 0;
}
```