

2023년 12월 3일 일요일



가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

본 연습문제는 가천대학교 컴퓨터공학전공의 'C언어' 기말고사를 대비하기 위함으로 제작되었습니다.

모든 문제는 화, 수요일의 오상엽 교수님 수업 기준으로 제작되었습니다.

본 연습문제의 저작권은 임규연 (lky473736)에게 있습니다.

* 질문 및 오류 제보 : lky473736@icloud.com

'problemsolving' GitHub : <https://github.com/lky473736/problemsolving/tree/main>

본 문서의 구성은 아래와 같습니다.

1p : 개괄

2p : FAQ

3p ~ 12p : 연습 10제

13p ~ : Q1 ~ Q5 해설

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

FAQ)

안녕하세요, 저는 컴퓨터공학전공의 임규연입니다.
중간고사에 이어서 연습문제를 제작하였습니다. 본 문서를 열람해주셔서 감사합니다.

이번에는 얼마 남지 않은 기말고사를 실전적으로 연습할 수 있도록 적절한 난이도의 손코딩 문제 5문항, 소스코드를 설명하는 문항 5문항을 만들었습니다.

시험을 연습해보는 일종의 '실전 모의고사'라고 생각하시고, 본 파일을 출력하여 풀어봄을 추천합니다.

아래는 저에게 자주 문의를 주시는 질문들에 대한 답변입니다.

Q1. GitHub에 있는 코드를 복붙해도 되나요?

A1. GitHub에 Repositories에 들어가보시면 대부분이 'MIT Licence'라고 붙여져 있을 겁니다. 그건 저작권법에 보호받는 코드들이여서, 상업적으로 이용하시면 안됩니다. 상업적으로 이용만 안하신다면 모든 코드들을 마음대로 복붙 및 활용하셔도 좋습니다.

Q2. 본 연습문제 이외에도 연습하고 싶은데 어떻게 해야 하나요?

A2. 본 연습문제 풀이를 완료하셨다면 코드업의 'C언어 기초 100제' (<https://codeup.kr/problemsetsol.php?psid=23>)로 직접 연습하시는 것을 추천드립니다.

문제 오류 및 문제풀이에 관한 질문은 언제든지 이메일로 보내주세요.
감사합니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q1) 구구단 구현

while문과 배열을 사용하여 구구단을 출력하는 프로그램을 만드시오.

예시)

1 * 1 == 1

1 * 2 == 2

...

3 * 1 == 3

3 * 2 == 6

...

9 * 8 == 72

9 * 9 == 81

(프로그램 종료)

* 문제의 해설은 13p에 있습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q2) 조건 연산자를 제어문으로 구현

아래 조건 연산자를 해석하고, 아래와 동일한 역할을 하는 **if-else문**으로 구현하시오.

```
max = (num1 > num2) ? ((num1 > num3) ? num1 : num3) : ((num2 > num3) ? num2 : num3);
```

* 문제의 해설은 13p에 있습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q3) 팩토리얼 함수 구현

정수형 변수 n 를 입력으로 받고, $n!$ 을 출력하는 함수 `fact()`가 포함된 프로그램을 구현하시오.

(예시)

3 입력

>>> 6

(프로그램 종료)

* 문제의 해설은 13p에 있습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q4) 포인터를 사용하여 배열의 요소를 역으로 출력 구현

크기 5의 배열의 각 요소를 입력 받고, 포인터를 사용하여 그 배열의 값을 역으로 출력하는 프로그램을 구현하시오.

(예시)

```
arr[0]? =  
>>> 3 입력
```

```
arr[1]? =  
>>> 4 입력
```

```
arr[2]? =  
>>> 5 입력
```

```
arr[3]? =  
>>> 6 입력
```

```
arr[4]? =  
>>> 10 입력
```

```
arr[4] = 10  
arr[3] = 6  
arr[2] = 5  
arr[1] = 4  
arr[0] = 3
```

(프로그램 종료)

* 문제의 해설은 13p에 있습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q5) 특정 문자열에 있는 특정 문자의 갯수 출력 구현

길이 10짜리 특정 문자열과 특정 문자 1개를 입력받고, 문자열 안에 특정 문자가 몇 개 있는지 출력하는 프로그램을 구현하시오. (단, 특정 문자열에서 특정 문자를 탐색 시 **포인터**를 사용하시오.)

(예시)

```
string / character? =  
>>> apple / p 입력
```

2

(프로그램 종료)

* 문제의 해설은 13p에 있습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q6) 개행 문자를 처리

아래 코드의 실행 결과를 작성하고, 아래 코드가 무한적으로 반복될 수 있도록 고치시오.

```
#include <stdio.h>

int main()
{
    char c;

    while ((c = getchar()) != '\n') // EOF
    {
        putchar(c);
    }

    return 0;
}
```

* Q6~Q10의 해설은 제공하지 않습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q7) 배열과 포인터

아래 코드의 실행 결과를 작성하고, 아래 함수 p를 포인터를 사용하여 다시 작성하시오.

```
#include <stdio.h>

char p(char k[2][4]);

void main()
{
    char k[2][4] = {"abc", "def"};
    p(k);
}

char p(char k[2][4])
{
    int i, j;
    for (i = 0; i < 2; i++)
    {
        for (j = 0; j < 4; j++)
        {
            printf ("%c", k[i][j]);
        }

        printf ("\n");
    }
}
```

* Q6~Q10의 해설은 제공하지 않습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q8) 복잡한 증감 연산자와 기억 클래스

아래 프로그램의 출력 결과를 작성하시오.

```
#include <stdio.h>

void p();

int main()
{
    int a = 1, b = 2, c = 3;

    if (a++ + 1 == b)
    {
        printf ("%d %d %d\n", a, b, c);
    }

    else
    {
        printf ("%d %d\n", a, b);
    }

    p();
    p();

    printf ("%d\n", ++a);

    return 0;
}

static int a = 10;

void p()
{
    auto int a = 1;
    printf ("%d\n", a);

    static char c = 'a';
    printf ("%c\n", c);
    c++;
}
```

* Q6~Q10의 해설은 제공하지 않습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q9) 개방 서브루틴 (프리프로세서)

개방 서브루틴과 프로그램 번역 과정을 설명하시오.

* Q6~Q10의 해설은 제공하지 않습니다.

2023년 12월 3일 일요일

가천대학교 컴퓨터공학전공 'C언어' 기말고사 대비 연습 10제

Q10) 구조체와 공용체

아래 프로그램을 설명하시오.

```
#include <stdio.h>

struct list{
    char *nam;
    char *tel;
};

struct list m;

void inp();

int main()
{
    inp();
    return 0;
}

void inp()
{
    char name1[10], tel1[10];

    scanf ("%10s %10s", name1, tel1);

    m.nam = name1;
    m.tel = tel1;

    printf ("%s %s\n", m.nam, m.tel);
}
```

* Q6~Q10의 해설은 제공하지 않습니다.

(Solution 1 ~ 5)

1. 구구단 구현

반복문의 이중 중첩 및 2차원 배열을 이용하면 손쉽게 구할 수 있습니다.

arr[9][9]를 선언하여,

row 1에서는 1단,

row 2에서는 2단,

...

예를 들어, 아래와 같이 구현할 수 있습니다.

```
#include <stdio.h>
```

```
int main()
{
    int gugudan[9][9];

    for (int i = 0; i < 9; i++)
    {
        for (int j = 0; j < 9; j++)
        {
            gugudan[i][j] = (i + 1) * (j + 1);
        }
    }

    for (int i = 0; i < 9; i++)
    {
        for (int j = 0; j < 9; j++)
        {
            printf ("%d * %d == %d\n", i + 1, j + 1, gugudan[i][j]);
        }
        printf ("=====\n");
    }
}
```

2. 조건 연산자를 제어문으로 구현

```
max = (num1 > num2) ? ((num1 > num3) ? num1 : num3) : ((num2 > num3) ? num2 : num3);
```

위 조건 연산자를 이용한 문장을 해석하자면 아래와 같습니다.

```
num1 > num2이면 -> num1 > num3이면 -> max = num1
                  아니면              -> max = num3
```

```
아니면          -> num2 > num3이면 -> max = num2
                  아니면              -> max = num3
```

(‘아니면’은 위의 질문의 부정이 됩니다. 예를 들어

```
num2 > num3이면
아니면
```

여기서의 ‘아니면’의 의미는 ‘num2 > num3이 아니면’ 입니다.)

세 수 num1, num2, num3을 입력 받고, 서로의 대소관계를 비교하여 최댓값을 찾는 문장이 되겠습니다. 따라서 위의 해석과 동일히 제어문을 작성하면 됩니다.

예를 들어, 아래와 같이 구현할 수 있습니다.

```
#include <stdio.h>

void main()
{
    int num1, num2, num3, max;

    scanf ("%d %d %d", &num1, &num2, &num3);

    if (num1 > num2)
    {
        if (num1 > num3)
        {
            max = num1;
        }

        else
        {
            max = num3;
        }
    }
}
```

(다음 장에 계속)

```
else
{
    if (num2 > num3)
    {
        max = num2;
    }

    else
    {
        max = num3;
    }
}

printf ("max = %d", max);
}
```

3. 팩토리얼 함수 구현

함수의 재귀 (recursion-call) 호출을 사용하면 쉽게 구현할 수 있습니다.
(static 변수를 활용하여도 구현 가능합니다.)

예를 들어, 아래와 같이 구현할 수 있습니다.

```
#include <stdio.h>

int fact(int n);

int main()
{
    auto int n;

    scanf ("%d", &n);

    printf ("factorial n : %d", fact(n));
}

int fact(int n)
{
    if (n == 1)
    {
        return 1;
    }

    else
    {
        return n * fact(n - 1); // recursion
    }
}
```


4. 포인터를 사용하여 배열의 요소를 역으로 출력 구현

일단 크기 5의 배열의 component를 각각 입력받아 저장한 후, 포인터의 주소값을 배열의 마지막 component에 대응되는 주소로 하여 1씩 감소하면서 출력하면 됩니다.

표현이 어렵다면 아래 코드를 참고해주세요. 감사합니다.

```
#include <stdio.h>

int main()
{
    int arr[5];
    int component;

    for (int i = 0; i < 5; i++)
    {
        scanf ("%d", &component);
        arr[i] = component;
    }

    int *point; // 포인터형 변수 startpoint 선언
    point = arr + 4; // 마지막 주소에 대응

    for (int i = 0; i < 5; i++)
    {
        printf ("arr[%d] == %d\n", 4 - i, *(point - i));
    }
}
```

5. 특정 문자열에 있는 특정 문자의 갯수 출력 구현

특정 문자열을 문자열 배열로 받아,
이산수학에서 배운 특정 문자를 찾는 linear search algorithm을 적용하면 됩니다.

예를 들어, 아래와 같이 구현할 수 있습니다.

```
#include <stdio.h>

int main()
{
    char string[10]; // 입력 받는 문자열
    char finding; // 찾아야 하는 문자
    int counting = 0; // 찾아야 하는 문자의 갯수

    gets(string);

    scanf ("%c", &finding);

    for (int i = 0; i < 10; i++)
    {
        if (finding == *(string + i))
        {
            counting += 1;
        }
    }

    printf ("%d", counting);
}
```

(수고하셨습니다.)