

# 기프 24 족보

#시험준비

1. int 자료형 변수는 최대값을 넘으면 자동으로 음수로 변환된다. (참/거짓)
  1. 참
2. printf() 함수는 항상 화면에 출력을 마친 뒤 개행을 포함한다. (참/거짓)
  1. 거짓
3. #define으로 정의된 상수는 프로그램 실행 중 값을 변경할 수 있다.
  1. 거짓
4. int 와 float 자료형의 산술 연산에서, int 가 자동으로 float 로 변환되므로 float 타입으로 결과가 나온다. (참/거짓)
5. 다음 코드에서 x++과 ++x의 결과는 동일하다. (참/거짓)
  2. 거짓 // 5 7

```
int x= 5;
printf("%d %d", x++, ++x); 5, 7
```

6. #define으로 정의된 상수는 프로그램 실행 중에 그 값을 변경할 수 없지만, const로 정의된 변수는 필요할 때 값을 변경할 수 있다. (참/거짓)
  1. 거짓
7. 다음 코드에서 max 변수는 프로그램 실행 도중 오버플로우가 발생하여 0으로 순환될 수 있다. (참/거짓)
  1. 참

```
unsigned int max = UINT_MAX;
max += 1;
```

8. \*\*int 변수 x 및 result 에 대해, 논리연산 및 비교 연산을 이용하여 x가 5이하이거나 10 이상일 때, result =1 , 그렇지 않을 때는 result = 0이 되도록 하는 명령어를 쓰시오. ( 단 if-else 금지 )\*\*

1. result = x >= 10 || x <= 5 ? 1 : 0;
2. result2 = (x >= 10) || (x <= 5);

9. \*\*다음 코드에서 논리적 오류를 수정하여 1부터 5까지의 숫자를 출력하도록 하시오\*\*  
\*\* (나머지 부분은 수정하지 않고 while의 조건문 만을 수정하여 1부터 5까지 출력 되도록 하시오.)\*\*

\*\*A. 제한 조건 : 출력 결과는 "1 2 3 4 5" 가 나와야한다.\*\*

```
```c
int x= 0
while ( )
printf("%d",x);
```

```

10. 한 계산기 시스템은 사용자로부터 두 개의 숫자를 입력받고, 다음 연산을 수행한다. 아래에 해당 연산을 처리하는 코드에서, **\*\*자료형 변환 오류와 연산자 우선순위 문제\*\***로 인해 예상치 못한 결과가 출력되고 있다. 이 코드를 분석하고 문제를 찾아 올바른 겨로가를 만들 수 있도록 코드를 수정하시오.

1. 출력 예시 :

합 : 16.200

곱 : 46

나눗셈 결과 : 3.37838

- A. 두 숫자의 합을 구하고, 소수점 이하 3자리 까지 출력
- B. 두 숫자의 곱을 구한 뒤, 그 결과를 정수로 출력
- C. 첫 번째 숫자에 두번째 숫자를 나누고, 그 결과를 소수점 이하 5자리까지 출력

```
```c
```

```
#include <stdio.h>
int main(){
    float num1, num2;
    int result1;
    float result2;
    printf("첫 번째 숫자를 입력하세요 : ");
    scanf("%f", &num1);
    printf("두 번째 숫자를 입력하세요 : ");
    scanf("%f", &num2);

    result1 = num1 + num2;
    //result2 = num1 + num2;
    printf("합 : %.3f\n", result1);
    //printf("합 : %.3f\n", result2);

    result2 = num1 * num2;
    printf("곱 : %d\n", result2);
    //printf("곱 : %d\n", (int)result2);

    result1 = num1 / num2;
    //result2 = num1/num2;
```

```

    printf("나눗셈 결과 : %.5f\n", result1);
    //printf("나눗셈 결과 : %.5f\n", result2);

    return 0;

}

```

11. 다음 프로그램에서 가)에 들어가야 하는 함수 선언과 프로그램 실행결과를 모두 작성하시오

```

#include <stdio.h>
가)
int main(void){
    int x= 5, y=5;
    operator();
    operator();
    operator();

    printf("x = %d, y =%d\n", x, y);

    return 0;
}

void operator(void){

    static int x= 0;
    int y = 0;

    x+=1;
    y+=1;

    printf("x : %d, y : %d \n", x, y);
}

```

가 ) void operator(void);

12. 다음 중략된 프로그램의 switch 문을 if문으로 변경하시오.

아래는 점수(score)에 따라 학점을 출력하는 프로그램의 일부이다. 프로그램의 중략된 부분을 표시했으며, switch문을 if문으로 변경하여 프로그램을 완성하시오.

A. 요구사항

1. switch 문을 if 문으로 변경하시오.
2. 점수 범위에 따른 조건을 if문을 사용하여 처리하시오.  
(95점 이상: A+, 90점이상:A, 85 점 이상: B+, 80점 이상 B )

```

switch(score){
    case 100:
    case 99:
    case 98:
    case 97:
    case 96:
    case 95:
        printf("A+학점.\n");
        break;

//중략된 부분 :case문을 사용하여 다른 학점을 출력하는 부분이 있습니다.

}

```

```

if(score<=100 && score>=95){printf("A+학점");}
else if(score>=90){printf("A학점");}
else if(score>=85) {printf("B+학점");}
else if(score>=80) {printf("B학점");}

```

13. 아래 프로그램에서 ---- 안에 부분을 함수의 호출을 이요하는 한 줄의 명령어로 수정하고 사용된 함수의 정의를 쓰시오.

```

#include <stdio.h>
int main(void){
    int i,n,sum=0;
    printf("정수를 입력하시오: ");
    scanf("%d", &n);
    ----
    for(int i=0;i<=n;i++)
        sum += i;
    ----
    printf("합은 %d입니다. \n", sum);

    return 0;
}

```

변경한 부분 : sum = number\_sum(n);

```

int number_sum(int n){
    if(n<=0) return 0;
    else return n + number_sum(n-1);
}

```

재귀함수

14. 아래 프로그램에서 ---- 안에 부분을 작성하여 실행 결과가 다음과 같도록 하시오.  
정수를 입력하시오: 5

```
#include <stdio.h>
int main(){
    int num;
    printf("정수를 입력하시오 : ");
    scanf("%d", &num);
    -----
    //실행결과
    *****
    ****
    ***
    **
    *
    -----
    return 0;
}
```

```
for(int i=5;i>0;i--){
    for(int j=0; j<i;j++){
        printf("*");
    }
    printf("\n");
}
```

[코딩 문제]

( ❤ )

1. `#define`과 `const`를 모두 사용하여 다음과 같은 상수를 정의하고, 사용자로부터 두 개의 정수를 입력받아 다음 연산을 수행한 후 결과를 출력하는 프로그램을 작성하시오.
  1. `#define`을 사용하여 AMX를 100으로 정의하시오.
  2. `const`를 사용하여 MIN을 0으로 정의하시오.
  3. 입력받은 두 정수 중 하나가 MAX보다 크면 MAX, 하나가 MIN보다 작으면 MIN으로 값을 조정하시오.
  4. 두 정수의 합과 곱을 출력하시오.

- 
1. 세 개의 상수를 정의하고, 각각 다른 자료형을 사용하는 변수를 이용하여 복합연산을 수행하는 프로그램을 작성하시오. 이 프로그램은 다음 요구사항을 충족해야 한다.

## 1. 상수 정의

1. PI (실수형 상수) : 3.141592
2. MULTIPLIER (정수형 상수) : 5
3. OFFSET (정수형 상수) : 100

## 2. 입력 : 하나의 정수와 하나의 실수를 입력 받는다.

1. 입력예시 :

10  
2.5

## 3. 출력 : 다음 수식을 기반으로 계산된 결과를 출력하시오:

$$(\text{정수 입력} + \text{MULTIPLIER}) / (\text{실수 입력} + \text{PI}) - \text{OFFSET}$$

출력 예시 :

결과 : -92.204

## 4. 제한 조건 :

1. 결과는 소수점 이하 3자리 까지 출력해야 한다.
2. 입력 받은 정수와 실수 중, 필요시 연산 과정에서 다른 자료형으로 변환하여 사용 한다.
3. 연산 과정에서 발생하는 정수와 실수 간 자료형 변환을 명시적으로 처리해야 한다.

---

1. 문제 요구사항: 한 은행은 대출 심사 시스템을 운영하고 있다. 고객이 대출을 받기 위해서는 여러 조건을 충족해야 한다. 은행은 다음 조건을 바탕으로 대출 승인 여부를 결정한다.

고객의 연 소득, 신용 등급, 대출 금액, 이미 받은 대출 금액, 나이를 입력 받아 아래의 5개 조건을 모두 만족하는지 여부를 판단하는 프로그램을 작성하시오. 대출이 승인되면 "대출 승인"을 거부되면 "대출 거부"를 출력하시오

1. 고객의 연 소득이 30,000,000원 이상이여야 한다.
2. 고객의 신용 등급은 1등급에서 10등급 사이여야 한다.
3. 대출 금액은 고객의 연 소득의 50% 이내여야 한다.
4. 총 대출 금액은 대출 금액과 이미 받은 대출 금액을 합산하여 산정된다.
5. 고객의 나이는 20세 이상 60 세 이하이어야 한다.

입력:

- 사용자로부터 입력 받아야 할 숫자에 굵은 글씨로 표기함
- 고객의 연 소득 (정수)
- 고객의 신용 등급 (1~10 사이의 정수)
- 대출 금액(정수)
- 이미 받은 대출 금액(정수)
- 고객의 나이(정수)
- ...

```
연 소득을 입력하세요 : **35000000**  
신용 등급을 입력하세요 (1~10): **3**  
대출 금액을 입력하세요: **15000000**  
이미 받은 대출 금액을 입력하세요: **5000000**  
나이를 입력하세요: **35**
```

출력:

출력예시:

**대출 승인**

1. 대출 조건을 모두 만족하면 "대출 승인"
  2. 조건을 만족하지 않으면 "대출 거부"
- 

4. 1부터 100 사이의 랜덤 숫자 두개와 사칙 연산(더하기, 빼기, 곱하기, 나누기) 및 나머지 연산 % 중 하나를 랜덤으로 선택하여 문제를 10개 출제하는 프로그램을 작성하시오. 각 문제의 사칙연산 및 나머지 연산은 매번 랜덤하게 선택되고, 사용자가 계산한 결과를 입력 받아 정답과 비교해맞았는지 틀렸는지를 출력하도록 하시오.(컴퓨터의 랜덤 피연산자와 오퍼랜드의 설정은 아래 랜덤 생성 코드를 참고하세요.)

```
#include <time.h>  
#include <stdlib.h> // 랜덤 넘버 생성 함수를 사용하기 위한 라이브러리  
  
scrand((unsigned) time (NULL));  
  
rand() % 100 + 1; // 1~100 사이의 랜덤 값 반환
```

<실행결과 예시> 사용자입력

$50 * 60 = \underline{3000}$

맞았습니다.

$43 / 21 = \underline{2}$

맞았습니다.

$100 \% 4 = \underline{0}$

맞았습니다.

$19 - 5 = \underline{14}$

맞았습니다.

$33 * 2 = \underline{66}$

맞았습니다.

$49 / 2 = \underline{24}$

맞았습니다.

$77 \% 3 = \underline{2}$

맞았습니다.

$15 + 12 = \underline{27}$

맞았습니다.

$9 - 3 = \underline{6}$

맞았습니다.

$40 * 7 = \underline{280}$

맞았습니다.

-----