

1. 자료형 범위 확인 및 오버플로우 테스터

1. 문제 설명

- char, int, float, double 등의 최대·최소 범위를 출력하고, 특정 값 이상으로 입력했을 때 오버플로우/언더플로우가 일어나는지 확인하는 프로그램을 작성하라.

2. 요구사항

- limits.h, float.h를 적절히 포함해 최대·최소값을 출력.
- 사용자에게 정수/실수를 각각 입력받고, 변수에 저장했을 때 어떻게 변하는지 출력.

[정수형 범위]

char : -128 ~ 127

int : -2147483648 ~ 2147483647

[실수형 범위]

float : 1.175494e-38 ~ 3.402823e+38

double: 2.225074e-308 ~ 1.797693e+308

정수를 입력하세요: 2147483647

입력한 값: 2147483647, 저장된 값: 2147483647

실수를 입력하세요: 9999999999

입력한 값: 9999999872.000000, 저장된 값:
9999999872.000000

1. 자료형 범위 확인 및 오버플로우 테스터

```
/* hw1.c */
#include <stdio.h>
#include <limits.h>
#include <float.h>

int main(void)
{
    printf("[ 정수형 범위 ]\n");
    printf("char : %d ~ %d\n", CHAR_MIN, CHAR_MAX);
    printf("int : %d ~ %d\n", INT_MIN, INT_MAX);

    printf("\n[ 실수형 범위 ]\n");
    printf("float : %e ~ %e\n", FLT_MIN, FLT_MAX);
    printf("double: %e ~ %e\n", DBL_MIN, DBL_MAX);

    // 오버플로우 실험
    int intValue;
    float floatValue;

    printf("\n정수를 입력하세요: ");
    scanf("%d", &intValue);
    printf("입력한 값: %d, 저장된 값: %d\n", intValue, intValue);

    printf("\n실수를 입력하세요: ");
    scanf("%f", &floatValue);
    printf("입력한 값: %f, 저장된 값: %f\n", floatValue, floatValue);

    return 0;
}
```

2. 형변환 실습 (정수/실수 계산기)

1. 문제 설명

- 정수 두 개와 실수 한 개를 입력받아, 여러 가지 연산을 수행한 뒤 출력한다.
- 연산 과정에서 정수형 계산, 실수형 계산을 혼합하여 자동 형변환, 강제 형변환을 각각 시도해본다.

2. 요구사항

- 예: $(\text{double})(a + b)$ vs. $(\text{double})a + (\text{double})b$ 결과 비교.
- 정수/실수로 나누기($(\text{double}) a / b$) 등 여러 케이스.

정수 2개와 실수 1개를 차례로 입력하세요: 10 3 2.5

- 정수 + 정수 = 13
- 정수 + 실수(자동 형변환) = 12.500000
- $(\text{double})(a+b) + c = 13.500000$
- a/b (자동) = 3.000000
- a/b (강제) = 3.333333

정수 2개와 실수 1개를 차례로 입력하세요: 4 2 7.7

- 정수 + 정수 = 6
- 정수 + 실수(자동 형변환) = 11.700000
- $(\text{double})(a+b) + c = 11.700000$
- a/b (자동) = 2.000000
- a/b (강제) = 2.000000

2. 형변환 실습 (정수/실수 계산기)

```
/* hw2.c */
#include <stdio.h>

int main(void)
{
    int a, b;
    double c;

    printf("정수 2개와 실수 1개를 차례로 입력하세요: ");
    scanf("%d %d %lf", &a, &b, &c);

    // 1) 정수형 계산
    int sumInt = a + b;
    printf("정수 + 정수 = %d\n", sumInt);

    // 2) 실수형 계산 (자동 형변환)
    double sumMix = a + c;
    printf("정수 + 실수(자동 형변환) = %lf\n", sumMix);

    // 3) 강제 형변환
    double sumForce = (double)(a + b) + c;
    printf("(double)(a+b) + c = %lf\n", sumForce);

    // 4) 나눗셈 비교
    double div1 = a / b;           // 정수 / 정수 -> 정수 결과 -> 자동 double 변환
    double div2 = (double)a / b;   // 강제 형변환 후 실수 나눗셈
    printf("a/b (자동) = %lf\n", div1);
    printf("a/b (강제) = %lf\n", div2);

    return 0;
}
```

3. `typedef` 활용 간단 프로젝트

1. 문제 설명

- `typedef`를 사용하여 새로운 자료형 이름을 정의한다(예: `typedef float SCORE;`).
- 학생의 성적을 입력받아 평균을 구하는 프로그램 작성
- 사용자가 입력하는 점수는 실수로 가능(예: 85.5, 90.0 등).

2. 요구사항

- SCORE라는 이름으로 `float`(또는 `double`)를 대체.
- 여러 과목 성적을 더하고, 평균을 계산.
- 평균을 적절히 형식 지정(소수점 이하 자리수)하여 출력.

국어, 영어, 수학 점수를 입력하세요 (예: 85.5 90.0 78.3):
85.5 90 78.3
총합: 253.80
평균: 84.60

국어, 영어, 수학 점수를 입력하세요 (예: 85.5 90.0 78.3):
95 100.0 88.7
총합: 283.70
평균: 94.57

3. typedef 활용 간단 프로젝트

```
/* hw3.c */
#include <stdio.h>

typedef float SCORE; // SCORE를 float의 별칭으로 정의

int main(void)
{
    SCORE korean, english, math;
    SCORE sum, average;

    printf("국어, 영어, 수학 점수를 입력하세요 (실수 가능): ");
    scanf("%f %f %f", &korean, &english, &math);

    sum = korean + english + math;
    average = sum / 3.0f;

    printf("총합: %.2f\n", sum);
    printf("평균: %.2f\n", average);

    return 0;
}
```