

## 실습 5

학과 : 전자공학과

학번 : 2023104322

이름 : 현시온

- 과제는 pdf로 변환하여 제출(과제 문서 첫 줄에 학과/학번/이름 포함)
- 과제는 순서대로 작성하며, 문제와 설명을 모두 포함(형식이 맞지 않으면 감점)
- 프로그램을 작성하는 문제는 소스코드와 실행 결과를 모두 text로 붙여넣기(그림으로 포함하지 말 것)하고 코드 설명 및 결과에 대한 설명을 포함해야 함
- 문의 사항은 이메일(nize@khu.ac.kr) 또는 오픈 카톡방을 이용

1. 아래 코드를 반복문을 이용하여 다시 작성하라.

```
std::cout << 2 << '\n';
std::cout << 4 << '\n';
std::cout << 6 << '\n';
std::cout << 8 << '\n';
std::cout << 10 << '\n';
std::cout << 12 << '\n';
std::cout << 14 << '\n';
std::cout << 16 << '\n';
```

다시 작성한 코드:

```
for (int count = 2; count <= 16; count += 2) {
//2가 할당된 정수형 변수 count에 중괄호 내의 코드를 처리할 때마다 2만큼 증가시키는 과정을
count가 16 이하일 때 까지만 반복한다.
    std::cout << count << '\n';
//중괄호 내의 코드는 변수 count를 출력 및 줄 바꿈 처리의 역할을 한다.
}
```

2. 아래 코드의 동작을 설명하라.

```
#include <iostream>

int main() {
    int a = 0; //정수형 변수 a 선언 및 0으로 초기화
    while (a < 20) { //a가 20보다 작은 값인 한 중괄호 안에 내용을 반복한다
        int b = 0; //정수형 변수 b 선언 및 0으로 초기화
        while (b < 10) { //b가 10보다 작은 값인 한 중괄호 안에 내용을 반복한다
            if ((a + b) % 2 == 0) //만약 a와 b의 합이 짝수라면
                std::cout << "*"; //문자열 *을 출력한다
            b++; //b에 1을 더한다
        }
        std::cout << '\n'; //줄바꿈
        a++; //a에 1을 더한다
    }
}
```

설명:

출력 결과:

3. 아래 코드를 `continue`를 사용하지 않는 코드로 다시 작성하라.

```
#include <iostream>

int main() {

    int x = 100, y;

    while (x > 0) {

        std::cin >> y;

        if (y == 25) {

            x--;

            continue;

        }

        std::cin >> x;

    }

}
```

```

        std::cout << "x = " << x << '\n';
    }
}

```

다시 작성한 코드:

```

#include <iostream>
int main() {
    int x = 100, y;
    while (x > 0) {
        std::cin >> y;
        if (y == 25) {
            x--;
        }
        else {
            std::cin >> x;
            std::cout << "x= " << x << '\n';
        }
    }
}

```

//continue 문법은 남은 코드를 실행하지 않고 반복문의 내용을 다시 실행하는 의미를 품고 있다. 그러므로 continue 밑 코드를 else로 묶어 처리하는 것을 통해 같은 역할을 하는 코드를 작성할 수 있다.

4. double형 자료를 20개 입력 받아서, 합, 평균, 최댓값과 최솟값을 출력하는 프로그램을 작성하라.

```

#include <iostream>
#include <limits>

int main() {

    double x = 0, sum = 0, mean = 0,
    max = std::numeric_limits<double>::min(),
    min = std::numeric_limits<double>::max();
    //입력 받은 값을 할당할 변수 x, 합을 할당할 변수 sum, 평균을 할당할 변수 mean을 선언하고, 대수
    //비교를 통한 최대 최소 결정을 위해 double형의 최솟값을 변수 max에, 최댓값을 변수 min에 할당한다.
    for (int count = 1; count <= 5; count++) {
        //10이 할당된 정수형 변수 count에 중괄호 내의 코드를 처리할 때마다 1만큼 증가시키는 과정을
        //count가 5 이하일 때 까지만 반복한다.
        std::cin >> x;
        sum = sum + x;
        mean = sum / count;
        if (x > max) {
            max = x;
        }
        if (x < min) {
            min = x;
        }
    }
    //중괄호 내의 코드는 x를 입력 받은 다음, sum에 그 x를 더하고, count는 입력 받은 자료의 수이므로
    //평균 = 총합/자료수를 의미하는 mean은 sum/count의 결과값을 할당하는 역할을 한다. 또한 가장 작은
    //값이 할당된 max와 입력 받은 값 x를 대수 비교하여 x가 max보다 크다면 그 x값을 max에 할당하고,
    //가장 큰 값이 할당된 min과 입력 받은 값 x를 대수 비교하여 x가 min보다 작다면 그 x값을 min에

```

```
std::cout << "sum is " << sum << '\n';
std::cout << "mean is " << mean << '\n';
std::cout << "max is " << max << '\n';
std::cout << "min is " << min << '\n';
}
// 반복문을 전부 처리하고, 연산 된 변수들을 출력한다.
```

```
int a = 0;
while (a < 100);
    std::cout << a++;
std::cout << '\n';
```

출력하는 두 문장이 while 문의 종괄호로 묶이지 않았으므로 while 문에서 조건 하에서 이루어지는 작업은 null space, 즉 아무 작업도 하지 않는다. 그러므로 조건을 벗어날 변화가 이루어지지 않으므로 아무 작업도 하지 않는 상태를 무한히 반복한다.

\*  
\* \*  
\* \* \*  
\* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \* \*  
\* \* \* \* \* \* \*  
\* \* \* \* \* \* \* \*  
\* \* \* \* \* \* \* \*  
\* \* \* \* \* \* \* \*  
\* \* \* \* \* \*  
\* \* \* \* \*  
\* \* \*

```
#include <iostream>

int main() {
    for (int a = 1; a <= 6; a++) {
        //1이 할당된 정수형 변수 a에 중괄호 내의 코드를 처리할 때마다 1만큼 증가시키는 과정을 a가 5
        이하일 때 까지만 반복한다.
        int b = 1;
        while (b <= a) {
            std::cout << "★";
            b++;
        }
    }
}
```

```

    }
    std::cout << 'Wn';
//새로운 정수형 변수 a 선언 및 1을 할당한 다음, b가 a보다 작거나 같은 한, *을 출력하고 b를 1
증가시키는 과정을 반복한다. 그런 다음 줄 바꿈을 실행한다. 따라서 해당 코드에 의해서 *이 6개인
줄까지 출력할 수 있다.
    if (a == 6) {
        for (int c = 1; c <= 5; c++) {
            int d = 5;
            while (d >= c) {
                std::cout << "*";
                d--;
            }
            std::cout << 'Wn';
        }
    }
}
//*이 6인 줄까지 처리했다면 a는 6이 할당되어 있을 것이므로 해당 if문이 처리될 조건을 만족한다.
해당 내용은 위 코드의 반대 과정으로 처리하여 *이 5인 줄부터 *이 1인 줄까지 출력해낼 수 있다.

```

7. 오일러의 수(Euler's number, e)는 자연로그의 밑(base of natural logarithms)이며, 아래와 같이 근사화할 수 있다.

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots$$

위 수식을 이용하여 e의 근사값을 계산하여 출력하는 프로그램을 작성하라. (단, 더해주는 항의 값이 0.00001보다 작은 값이 나타날 때까지 계산)

작성한 코드:

```

#include <iostream>

int main() {
    double euler = 1, facto = 1, count = 1;
//자연상수를 표현할 변수 euler에 1을 할당, 팩토리얼을 표현할 변수 facto에 1을 할당, 팩토리얼이
자연수의 곱이므로 그 자연수를 표현할 변수 count에 1을 할당한다.
    while ((1 / facto) > 0.00001) {
//더해주는 항의 값이 팩토리얼의 역수이므로 1/facto라고 볼 수 있고, 1/facto가 0.00001보다
작아지는 순간 while문 조건에 벗어나므로 euler에 1/facto를 더하는 것을 멈춘다.
        facto = facto * count;
        euler = euler + (1 / facto);
        count++;
    }
//변수 facto는 팩토리얼을 의미하므로 반복 횟수를 의미하는 count를 곱하여 표현하고, 변수
euler에는 팩토리얼의 역수 1/facto를 더하여 표현한 다음 count에 1을 더하는 과정을 while문 조건을
만족하는 한 반복한다.
    std::cout << euler;
}
//해당 과정을 통해 도출된 자연상수를 의미하는 euler 값을 출력한다.

```

8. for 문을 이용하여 아래와 같이 출력되는 프로그램을 작성하시오.

6, 8, 10, 12, ..., 60

작성한 코드:

```
#include <iostream>
int main() {
    for (int a = 6; a <= 58; a += 2) {
        std::cout << a << ", ";
    }
    std::cout << 60;
}
```

//60이 할당된 정수형 변수 a에 중괄호 내의 코드를 처리할 때마다 2만큼 증가시키는 과정을 a가 58 이하일 때 까지만 반복한다. 중괄호 내의 코드는 a 값과 문자열 ', ' 출력의 역할을 한다. 마지막 60은 쉼표 처리의 예외를 위해 평범하게 직접 출력했다.

9. for 문을 이용하여 아래와 같이 출력되는 프로그램을 작성하시오.

7, 9, 11, 13, ..., 67

작성한 코드:

```
#include <iostream>
int main() {
    for (int a = 7; a <= 65; a += 2) {
        std::cout << a << ", ";
    }
    std::cout << 67;
}
```

//70이 할당된 정수형 변수 a에 중괄호 내의 코드를 처리할 때마다 2만큼 증가시키는 과정을 a가 65 이하일 때 까지만 반복한다. 중괄호 내의 코드는 a 값과 문자열 ', ' 출력의 역할을 한다. 마지막 67은 쉼표 처리의 예외를 위해 평범하게 직접 출력했다.

10. for 문을 이용하여 아래와 같이 출력되는 프로그램을 작성하라.

1, 4, 7, 10, ... 의 수열에서 50개의 출력

작성한 코드:

```
#include <iostream>
int main() {
    for (int a = 1; a < 50; a++) {
        std::cout << 1 + 3*a << ", ";
    }
    std::cout << 151;
}
```

//수열에서 50개까지만 출력하기에 항의 구별을 위한 정수형 변수 a를 채택한다. 따라서 해당 수열의 일반항은  $1+3*n$  라고 볼 수 있다. 10이 할당된 정수형 변수 a에 중괄호 내의 코드를 처리할 때마다 1만큼 증가시키는 과정을 a가 50보다 작을 때 까지만 반복한다. 중괄호 내의 코드는 제 a항을 의미하는  $1 + 3*a$  연산의 결과값과 문자열 ', ' 출력의 역할을 한다. 마지막 151은 쉼표 처리의 예외를 위해 평범하게 직접 출력했다.