기초컴퓨터프로그래밍 10분반 25\_이규하 (2021039057)

6주차 Report

문제 1) 다음 중 틀린 문장을 지적하고 올바르게 수정하라.

1. 조건식의 값이 1일 때만 참으로 간주된다.

→ 조건식의 값이 0이 아니면 다 참이다.

1. 반복 루프가 중첩되어 있는 경우, break는 하나의 반복 루프만 벗어 날 수 있다.
2. do…while 문에서 조건식의 값이 거짓이면 한 번도 수행되지 않는다.

→ 거짓이라 해도 그 안에 문장은 한번은 반드시 실행이 된다.

1. for 문에서 초기식, 조건식, 증감식이 전부 비어 있으면 안 된다.

→ 셋 다 비워있어도 상관없다. 단 그때는 무한루프를 돌게 될 것이다.

1. for문안에 다른 for 문이 들어갈 수 있다.

문제2) 다음은 무한 반복을 구현한 소스이다. 올바르게 구현된 것을 모두 골라보자.

1. while(1) {}
2. for(1) {}
3. for( ; 1 ; ) {}
4. for( ; ; ; ) {}

문제3) 다음의 프로그램을 실행시키면 “Hello World!”는 몇 번이나 출력되는가?

// 답 : 6번

#include <stdio.h>

int main(void)

{

    int x;

    for (x = 0; x < 10; x++){

// x가 0부터 9가 될 때까지 아래의 조건을 반복해서 돌릴 것이다.

// x가 0부터 5까지는 아래 if 조건이랑은 상관없으니 출력만 반복해서 돌릴거다.

        if (x > 5) // 만약에 x가 0부터 8까지는

            continue; // 아래 출력문을 내보내지 않고, for 문단의 처음으로 되돌아간다.

        if (x > 8) // 만약에 x가 9일때는

            break; // 아래 출력문을 내보내지 않고, for 문단을 탈출한다.

        printf("Hello World! \n"); // 출력문이다.

    }

    return 0;

}

문제4) 다음의 프로그램에서 생성되는 출력 결과는 무엇인가?

(a)

/\* 출력 결과 :

0

3

6

9

\*/

#include <stdio.h>

int main(void)

{

    int i = 0; // while 조건에 들어갈 i를 0으로 초기화 시켰다.

    while (i < 10){ // i 가 9가 될 때까지 아래 문장을 돌릴 것이다.

        printf("%d\n", i); // i 값을 출력을 해주자

        i += 3; // i는 반복문 돌 때마다 3씩 늘어난다.

    }

    return 0;

}

(b)

/\* 출력 결과 :

0

3

6

9

\*/

#include <stdio.h>

int main(void)

{

    int i = 0; // while 조건에 들어갈 i를 0으로 초기화 시켰다.

    do // 아래의 문장을 진행해라

    {

        printf("%d\n", i); // i 값을 출력해주라

        i += 3; // i 값은 3씩 더해질 것이다

    } while (i < 10); // i가 9가 될 때까지 위의 두문장을 반복하거라

    return 0;

}

(a), (b) 차이점 : (a)는 while 문에 있는 i의 조건을 먼저 판단하여 반복문 내 실행문을 돌리게 되지만, (b)는 do와 while 사이에 있는 실행문을 먼저 진행 한 후 while에 있는 i의 조건을 따져 반복문으로 돌릴지 그대로 탈출할지 결정하는 것에서 차이가 있다.

(c)

/\* 출력 결과 :

0

2

4

6

8

\*/

#include <stdio.h>

int main(void)

{

    int i;

    for(i = 0; i < 10; i += 2){ // 정수로 선언된 i에다 0으로 초기화했다.

    //반복문 돌릴때마다 i에2씩 더해질 것이고 i가 9가 될 때까지 아래의 출력문장을 반복 할 것이다.

        printf("%d\n", i); // i 값을 출력하라

    }

    return 0;

}

(d)

/\* 출력 결과 :

10

9

8

7

6

5

4

3

2

1

0

\*/

#include <stdio.h>

int main(void)

{

    int i;

    for (i = 10; i >= 0; i--) { // 정수로 선언된 i에다 10으로 초기화 했다.

    // 반복문 돌릴 때 마다 i에 1씩 빼질 것이고, i가 0이 될 때까지 아래의 출력 문장을  반복 할 것이다.

        printf("%d\n", i); // i 값을 출력하라

    }

        return 0;

}

(e)

// 출력 결과 : \*\*\*\*\*\*\*\*\*

#include <stdio.h>

int main(void)

{

    int x, y;

    for (x = 0; x < 3; x++) // 정수로 선언된 x에다 0으로 초기화 했다

    // 반복문 돌릴 때마다 x에 1씩 더해질 것이고, x가 2가 될 때까지 아래의 반복문을 반복할 것이다.

        for(y = 2; y >= 0; y--) // 정수로 선언된 y에다 2로 초기화 했다.

    // 반복문 돌릴 때마다 y에 1씩 뺴질 것이고, y가 0이 될 때까지 아래의 출력문을 반복할 것이다.

            printf("\*"); // \*을 출력하라

    // 결국 x가 0, 1, 2일때 내부 반복문에서는 각각 \*\*\*로 출력이 될 것이고

    // 이를 이으면 \*\*\*\*\*\*\*\*\* 가 된다.

        return 0;

}

(f)

// 출력 결과 : 54321

#include <stdio.h>

int main(void)

{

    int i = 5; // for 조건에 들어갈 i를 5로 초기화 시켰다.

    for ( ; i; i--)

// for에서 초기식은 없고, 초건식은 only i 뿐이고,

증감식은 i가 1씩 줄어드는 것이다.

     // 따라서 아래 문장을 반복할 때 마다 i는 1씩 빼질 것이다.

        printf("%d", i); // i를 출력하라

    return 0;

}

문제5) 동일한 결과를 생성하도록 for 루프는 while 루프로, while 루프는 for 루프로 변환하라.

(a)

before

#include <stdio.h>

int main(void)

{

    int i;

    for (i = 10 ; i >= 0; i--)

    {

        printf("%d\n", i);

    }

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    int i = 10;

    while (i >= 0)

    {

        printf("%d\n", i);

        i--;

    }

    return 0;

}

(b)

before

#include <stdio.h>

int main(void)

{

    int i = 10;

    while (i >= 0)

    {

        printf("%d\n", i);

        i -= 3;

    }

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    int i;

    for (i = 10 ; i >= 0; i -= 3)

    {

        printf("%d\n", i);

    }

    return 0;

}

문제 6) 다음의 코드에서 잘못된 점이 있으면 지적하고 올바르게 수정하라. 논리적인 오류도 포함된다.

(a) 반복문을 돌리는데 i값의 변화가 없으니 반복문은 무한 루프에 빠지게 된다. 따라서 반복문 내에 i++을 추가하였다. 그리고 i 값을 출력해야 하므로 정수인 i를 받는 %d를 이용하여 i값을 출력해줬다.

before

#include <stdio.h>

int main(void)

{

    int i = 0;

    while(i < 10)

    {

        printf("i의 값\n", i);

    }

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    int i = 0;

    while(i < 10)

    {

        printf("i의 값 : %d\n", i);

        i++

    }

    return 0;

}

(b) while 줄 뒤에는 ; 가 붙지 않는다. 그리고 i 값을 출력해야 하므로 정수인 i를 받는 %d를 이용하여 i값을 출력해줬다.

before

#include <stdio.h>

int main(void)

{

    int i = 0;

    while(i++ < 10) ;

    {

        printf("i의 값\n", i);

    }

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    int i = 0;

    while (i++ < 10)

    {

        printf("i의 값 : %d\n", i);

    }

    return 0;

}

(c) for의 증감식이 i—이었다면 조건식에 따라 밑도 끝도 없이 1씩 빼면서 무한루프에 빠지게 된다. 따라서 i++로 바꿔줘야 한다.

before

#include <stdio.h>

int main(void)

{

    int i;

    for(i = 0; i < 10; i--)

        printf("i = %d\n", i);

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    int i;

    for (i = 0; i < 10; i++)

        printf("i = %d\n", i);

    return 0;

}

(d) 관계 연산자 ‘ != ’ 는 두 연산자의 값이 다른 것인지 확인하는 연산자이다. 그러나 실수는 소수점 아래의 수도 있으므로 정확히 1.0이 될 수는 없을 것이다. 이리 되면 밑도 끝도 없이 0.1씩 더해져 무한루프에 빠지게 될 것이다. 따라서 x < 1.0 으로 바꿔줘야 한다.

before

#include <stdio.h>

int main(void)

{

    float i;

    for (x = 0.1; x != 1.0; x += 0.1)

        printf("%f\n", i);

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    float i;

    for (x = 0.1; x < 1.0; x += 0.1)

        printf("%f\n", i);

    return 0;

}

(e) 정수인 변수 i가 특정한 값으로 초기화 되지 않았다. 따라서 나는 for의 초기문에서 i를 0으로 초기화 해주었다.

before

#include <stdio.h>

int main(void)

{

    int i;

    for ( ; i < 10; i++)

        printf("i = %d\n", i);

    return 0;

}

after

#include <stdio.h>

int main(void)

{

    int i;

    for (i = 0; i < 10; i++)

        printf("i = %d\n", i);

    return 0;

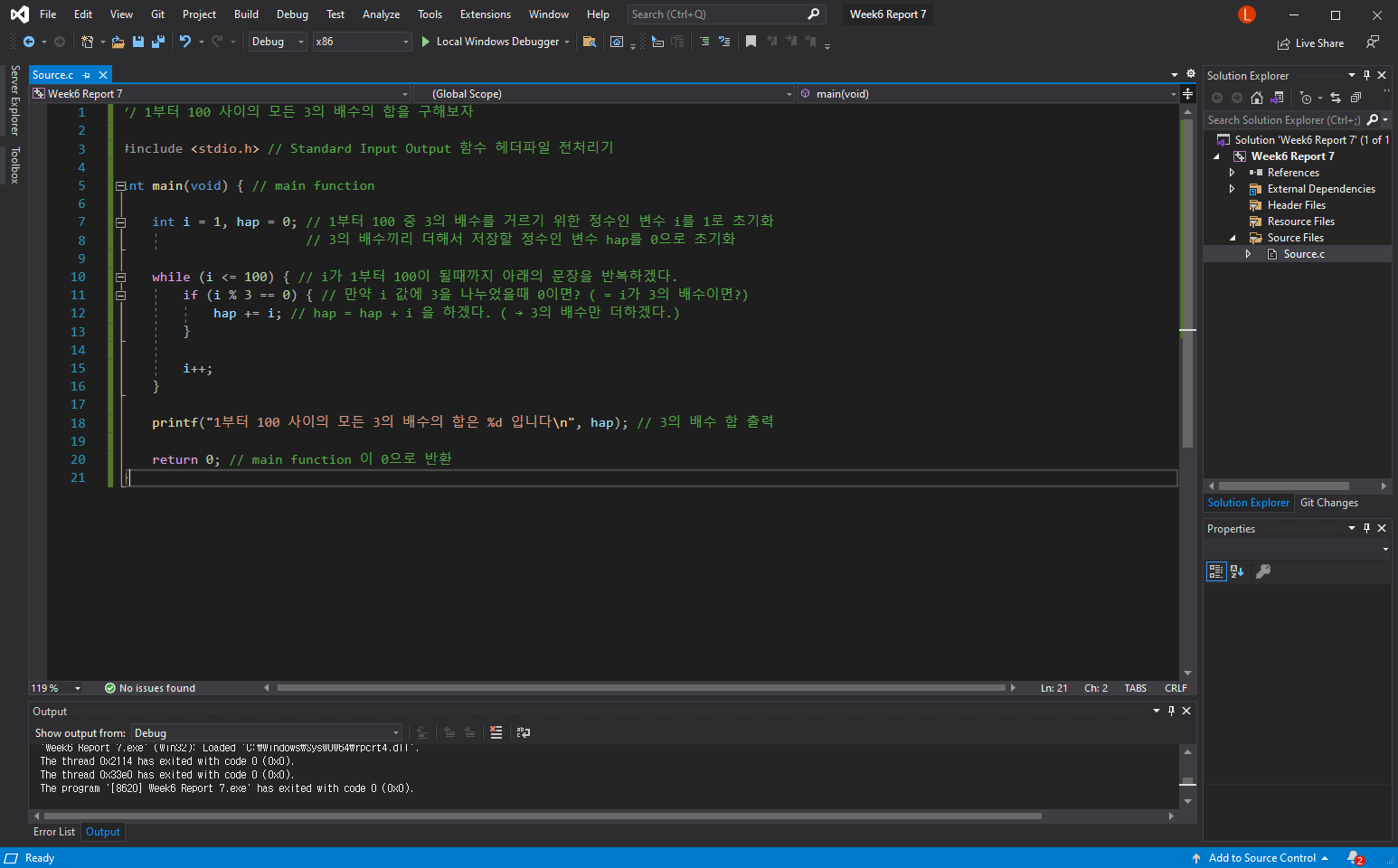
}

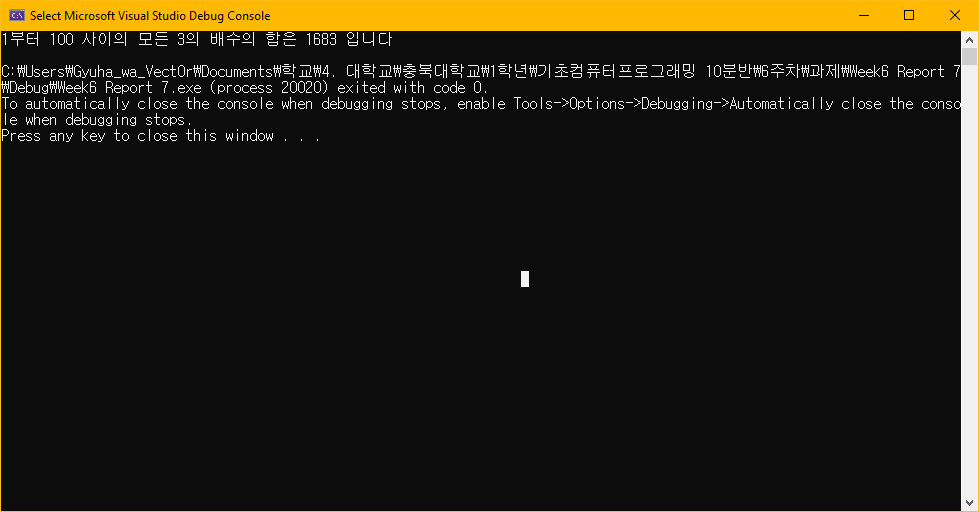
문제 7) 1부터 100 사이의 모든 3의 배수의 합을 계산하여 출력하는 프로그램을 반복 구조를 사용하여 작성하라

Hint : 3의 배수의 합은 i % 3 == 0의 조건으로 검사할 수 있다.

* 문제 분석 & 동작 설명

: 조건이 어떻게 되든 1부터 100까지의 합을 구하라 했으므로 반복문을 이용하면 간편하다. while, for, do\_while 중 하나를 이용하면 된다. while 문을 이용하여 반복문을 돌려보자. 3의 배수는 어느 수를 3으로 나누었을 때 나머지가 0이 되면 3의 배수이다. 따라서 if와 나머지 연산자인 %을 이용하여 i % 3 = 0이라는 조건으로 3의 배수를 판별해주자. 판별해 준 3의 배수 끼리를 더해서 출력하면 되겠구나.

* 프로그램 소스 (주석 포함)



문제 8) 중첩 반복문을 사용하여서 다음과 같이 출력하는 프로그램을 작성하여 보자.

* 문제 분석 & 동작 설명

: 중첩 반복문을 사용해라는 것은 반복문 안에 반복문을 이용해라는 것이다. 바깥 반복문에서 한번 돌리는 동안 내부 반복문에서 그 반복문의 조건을 다 수행하는 식으로 흘러가야 한다.

* 프로그램 소스 (주석 포함)

