

8 주차 Report

문제 1) 다음의 프로그램에서 붉은 색으로 표시된 부분을 함수로 작성하여 프로그램을 수정하시오. 함수는 인수와 반환 값을 갖도록 설계하시오.

sol)

before

```
#include <stdio.h>

int main(void){

    int i, n, sum = 0;

    printf("정수를 입력하시오 : ");
    scanf("%d", &n);

    for(i = 0; i <= n; i++) // 함수로 묶을 부분
        sum += i;

    printf("0 부터 %d 까지의 합은 %d 입니다.\n");
    return 0;

}
```

after

```
#include <stdio.h>

int hap(int x, int y) { // 함수를 만들었다. 매개변수 x, y 를 정수형으로 선언

    int i; // for 문 돌릴 변수를 메인함수에서 땡겨와서 함수 내 지역변수선언

    for (i = 0; i <= x; i++) // 매개변수로 교체 n -> x
        y += i; // 매개변수로 교체 sum -> y

    return y; // 반환 될 때, y 값을 정수로 반환

}
```

```
}

int main(void) {

    int n, sum = 0;

    printf("정수를 입력하시오 : ");
    scanf_s("%d", &n);

    printf("0 부터 %d 까지의 합은 %d 입니다.\n", n, hap(n, sum));
    // 입력받은 n 과 sum 을 윗 함수의 매개변수로 전달 후 함수 돌리고 리턴받음
    return 0;
}
```

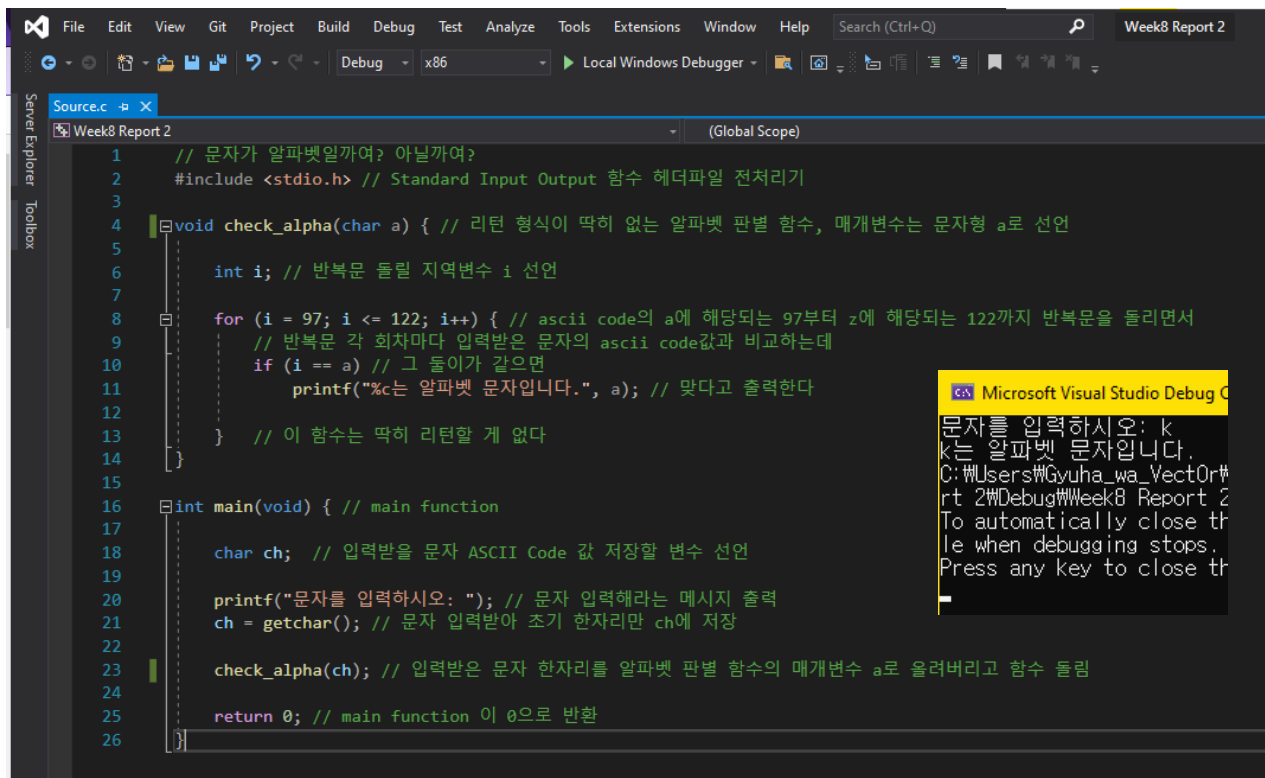
문제 2) 전달된 문자가 알파벳 문자인지 아닌지를 검사하는 함수 check_alpha()를 작성하고 이것을 호출하여서 사용자가 입력한 문자가 알파벳('a'에서 'z'까지)인지를 판단하여 출력하는 프로그램을 작성하라.

Hint : 문자를 입력받을 때는 ch = getchar(); 를 사용해도 좋다. scanf("%c", &ch); 를 사용해도 된다.

- 문제 분석 & 동작 설명

입력받은 문자가 알파벳인지 아닌지를 판별하기 위해서는 if 절을 이용하면 될 것이다. 그대신 함수를 작성해라 했으니 기본적으로는 if 절을 새로 만든 함수에다 넣고, 메인함수에서 불러오도록 하면 되겠구나. 입력 받은 문자를 알파벳 판별 함수로 들어가면서 문자형 매개변수로 들어가게 한다. a 부터 z 까지 해당되는 수인 97 부터 122 까지 for 반복문을 돌려 각 턴에 해당되는 값이 입력받은 문자의 ascii 코드값과 일치하는지 다른지 if 문으로 판별할 수 있도록 하자. 일치하면 맞다고 출력할 것이고, 다르면 종료하도록 하면 되겠다.

- 프로그램 소스 (주석 포함)



```
1 // 문자가 알파벳일까요? 아닐까요?
2 #include <stdio.h> // Standard Input Output 함수 헤더파일 전처리기
3
4 void check_alpha(char a) { // 리턴 형식이 딱히 없는 알파벳 판별 함수, 매개변수는 문자형 a로 선언
5
6     int i; // 반복문 돌릴 지역변수 i 선언
7
8     for (i = 97; i <= 122; i++) { // ascii code의 a에 해당되는 97부터 z에 해당되는 122까지 반복문을 돌리면서
9         // 반복문 각 회차마다 입력받은 문자의 ascii code값과 비교하는데
10         if (i == a) // 그 둘이 같으면
11             printf("%c는 알파벳 문자입니다.", a); // 맞다고 출력한다
12     } // 이 함수는 딱히 리턴할 게 없다
13
14 }
15
16 int main(void) { // main function
17
18     char ch; // 입력받을 문자 ASCII Code 값 저장할 변수 선언
19
20     printf("문자를 입력하십시오: "); // 문자 입력해라는 메시지 출력
21     ch = getchar(); // 문자 입력받아 초기 한자리만 ch에 저장
22
23     check_alpha(ch); // 입력받은 문자 한자리를 알파벳 판별 함수의 매개변수 a로 올려버리고 함수 돌림
24
25     return 0; // main function 이 0으로 반환
26 }
```

Microsoft Visual Studio Debug Console Output:

```
문자를 입력하십시오: k
k는 알파벳 문자입니다.
C:\Users\Wgyuha_wa_VectOr\rt 2\Debug\Week8 Report 2
To automatically close the console when debugging stops,
Press any key to close the console.
```

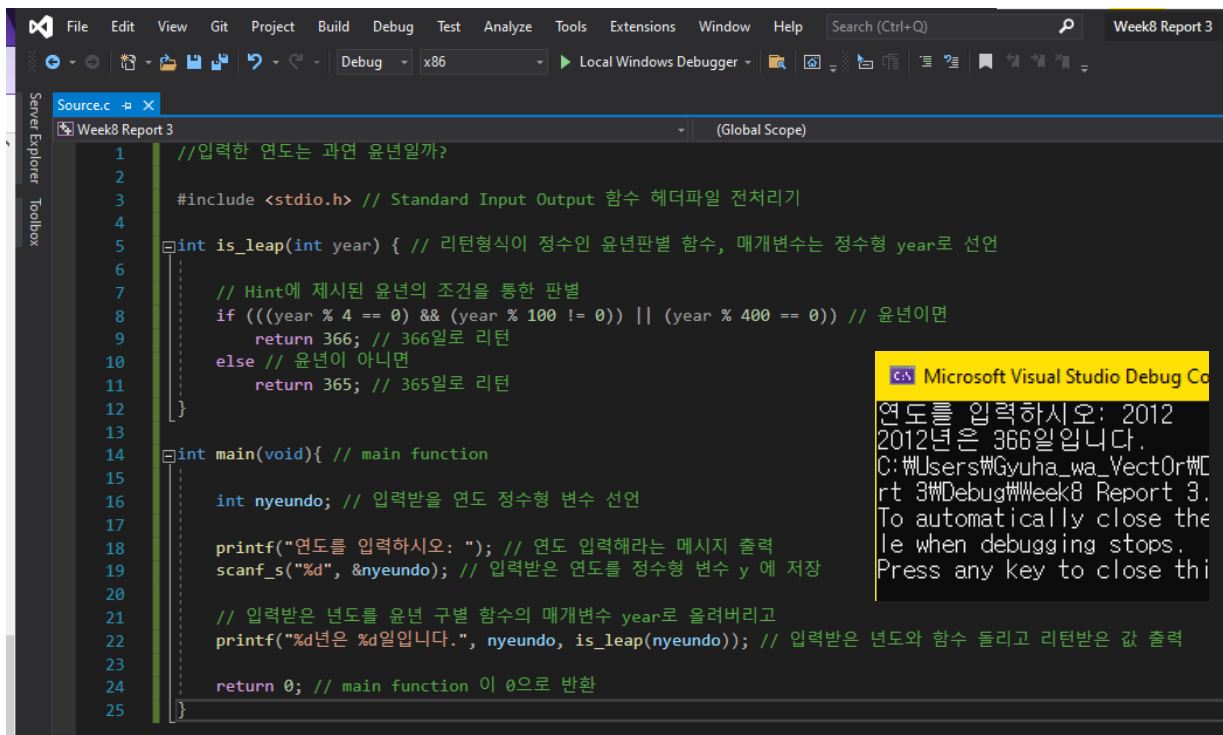
문제 3) 우리는 앞에서 윤년을 구하는 알고리즘을 학습하였다. 이것을 함수 `is_leap(int year)` 함수로 작성하고 이 함수를 사용하여 사용자가 입력한 연도가 윤년인지를 출력하는 프로그램을 작성하라.

Hint : 윤년은 4 의 배수이지만 100 의 배수는 제외하고 400 의 배수는 무조건 추가하면 구할 수 있다. 윤년이면 366 일이고 평년이면 365 일 이 된다.

- 문제 분석 & 동작 설명

윤년을 구하는 알고리즘은 앞서 힌트에 자세히 이야기해주고 있다. 따라서 이 힌트를 윤년을 판별해주는 if 의 조건문에 넣어주면 될듯하다. 이러한 판별을 `is_leap(int year)` 함수로 작성해라 했으므로 if 문을 저 새로운 함수로 넣어주면 되겠다. 입력받은 연도를 바탕으로 앞서 언급한 새 함수에 넣어 돌려주고 윤년이면 366, 아니면 365 로 리턴받은 값을 출력하면 끝!

- 프로그램 소스 (주석 포함)



```
1 //입력한 연도는 과연 윤년일까?
2
3 #include <stdio.h> // Standard Input Output 함수 헤더파일 전처리기
4
5 int is_leap(int year) { // 리턴형식이 정수인 윤년판별 함수, 매개변수는 정수형 year로 선언
6
7     // Hint에 제시된 윤년의 조건을 통한 판별
8     if (((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0)) // 윤년이면
9         return 366; // 366일로 리턴
10    else // 윤년이 아니면
11        return 365; // 365일로 리턴
12 }
13
14 int main(void){ // main function
15
16     int nyeundo; // 입력받은 연도 정수형 변수 선언
17
18     printf("연도를 입력하시오: "); // 연도 입력해라는 메시지 출력
19     scanf_s("%d", &nyeundo); // 입력받은 연도를 정수형 변수 y 에 저장
20
21     // 입력받은 연도를 윤년 구별 함수의 매개변수 year로 올려버리고
22     printf("%d년은 %d일입니다.", nyeundo, is_leap(nyeundo)); // 입력받은 연도와 함수 돌리고 리턴받은 값 출력
23
24     return 0; // main function 이 0으로 반환
25 }
```

Microsoft Visual Studio Debug Console Output:

```
연도를 입력하시오: 2012
2012년은 366일입니다.
C:\Users\Gyuha_wa_VectO\rt 3\Debug\Week8 Report 3.
To automatically close the
le when debugging stops.
Press any key to close thi
```

문제 4) 다음과 같은 간단한 기능을 하는 함수들을 작성하고, 사용자로부터 임의의 값을 입력받은 후에 작성한 함수들을 테스트하여 보자.

- (a) 주어진 정수가 짝수이면 1 을 반환하고 홀수이면 0 을 반환하는 함수 `int even(int n)`
- (b) 주어진 정수의 절대값을 구하는 함수 `int absolute(int n)`
- (c) 주어진 정수가 음수이면 -1 을, 양수이면 1 을, 0 이면 0 을 반환하는 함수 `int sign(int n)`

Hint : 함수 원형을 먼저 정의한 후에 함수를 정의하도록 하자. 함수들을 구현할 때, 조건 연산자 `?:`를 사용하여도 된다.

- 문제 분석 & 동작 설명

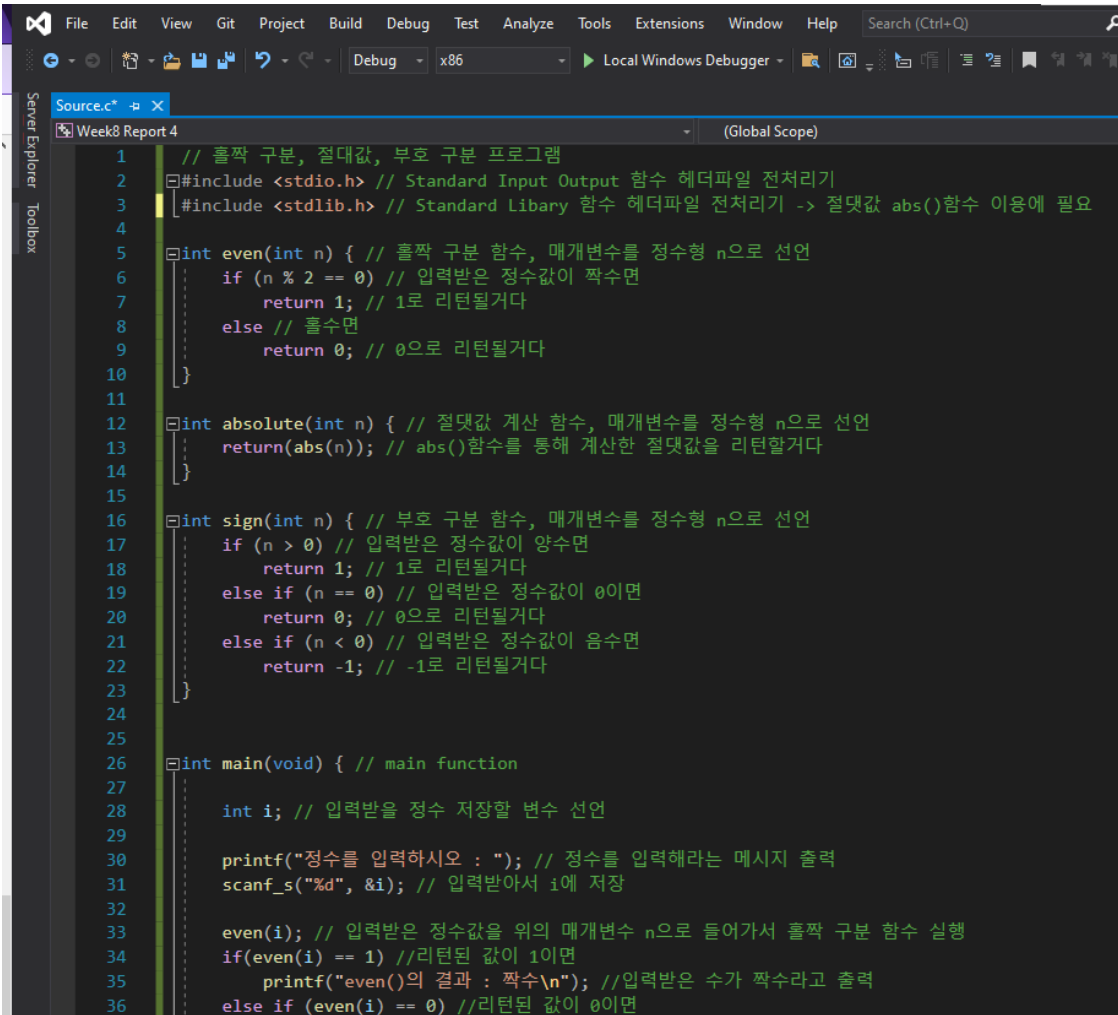
사용자로부터 어떠한 입력 값을 받아 홀짝, 절댓값, 부호를 구분해야 한다. 문제에 제시된 대로 각각의 function 을 만들면 되는데, 우선 홀짝 함수를 보면 짝수는 1 로 리턴, 홀수는 0 으로 리턴 값 자체가 지정되어있다. 따라서 우리는 입력받은 수가 짝수나 홀수나 판별해주면 되는데 이는 if 절을 입력받은 수가 2 로 나누어떨어지면 짝수고 아니면 홀수라는 조건으로 이용하여 함수를 만들 수 있을 것이다.

입력한 수의 절댓값을 구하는 함수를 보면, 입력받은 수가 음수, 양수인지 if 문으로 따져서 음수이면 -1 을 곱해서 양수로 바꾸어 리턴하고, 양수이면 그대로 리턴하는 함수를 만들 수 있을 것이다. 아 물론 `stdlib.h` 에 포함되어있는 `abs()` 함수를 넣으면 굳이 if 문이 필요없을거다! 내 스타일상... 이번에는 `abs()`를 써볼까 싶다.

마지막으로 부호 구분 함수를 보면 양수는 1 으로, 0 은 0 으로, 음수는 -1 으로 리턴 값이 지정되어 있다. 따라서 우리는 입력받은 수가 양수냐, 0 이냐, 음수냐 판별해주면 되는데 이는 if 절을 통해 그 조건들을 이용하여 함수를 만들 수 있을 것이다.

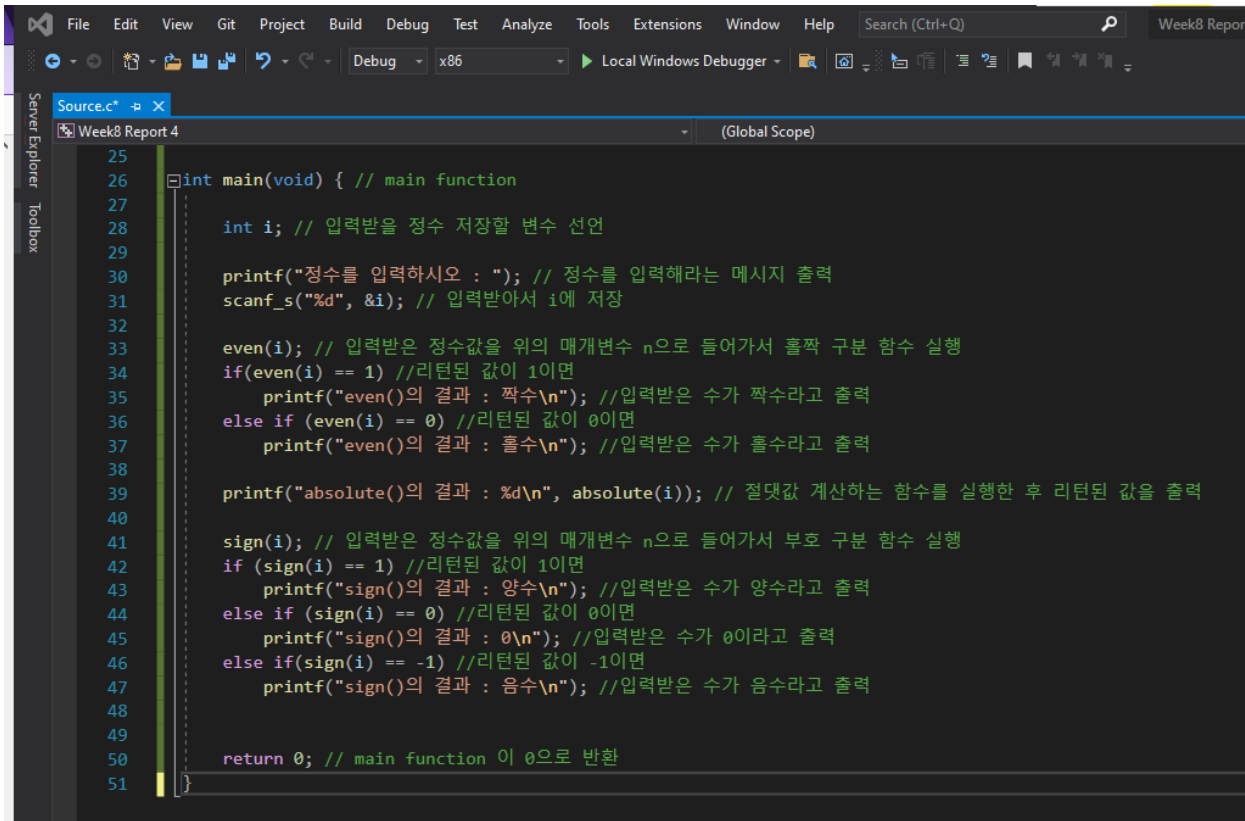
이렇게 입력받은 수에 대한 3 가지 함수에 대하여 출력을 해줘야 하는데, `even()`이랑 `sign()`은 각 함수 내에서 if 절로 판별했던 것처럼, 리턴 값을 if 절로 판별해주어 홀짝, 부호 판단을 해주고, `absolute()`는 그대로 출력해주면 되겠다.

- 프로그램 소스 (주석 포함)
(Line 1 ~ Line 36)

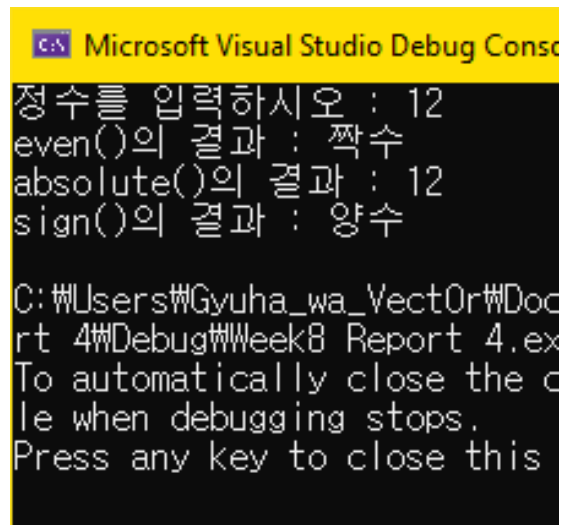


```
1 // 홀짝 구분, 절대값, 부호 구분 프로그램
2 #include <stdio.h> // Standard Input Output 함수 헤더파일 전처리기
3 #include <stdlib.h> // Standard Library 함수 헤더파일 전처리기 -> 절댓값 abs()함수 이용에 필요
4
5 int even(int n) { // 홀짝 구분 함수, 매개변수를 정수형 n으로 선언
6     if (n % 2 == 0) // 입력받은 정수값이 짝수면
7         return 1; // 1로 리턴될거다
8     else // 홀수면
9         return 0; // 0으로 리턴될거다
10 }
11
12 int absolute(int n) { // 절댓값 계산 함수, 매개변수를 정수형 n으로 선언
13     return(abs(n)); // abs()함수를 통해 계산한 절댓값을 리턴할거다
14 }
15
16 int sign(int n) { // 부호 구분 함수, 매개변수를 정수형 n으로 선언
17     if (n > 0) // 입력받은 정수값이 양수면
18         return 1; // 1로 리턴될거다
19     else if (n == 0) // 입력받은 정수값이 0이면
20         return 0; // 0으로 리턴될거다
21     else if (n < 0) // 입력받은 정수값이 음수면
22         return -1; // -1로 리턴될거다
23 }
24
25
26 int main(void) { // main function
27
28     int i; // 입력받을 정수 저장할 변수 선언
29
30     printf("정수를 입력하시오 : "); // 정수를 입력해라는 메시지 출력
31     scanf_s("%d", &i); // 입력받아서 i에 저장
32
33     even(i); // 입력받은 정수값을 위의 매개변수 n으로 들어가서 홀짝 구분 함수 실행
34     if(even(i) == 1) //리턴된 값이 1이면
35         printf("even()의 결과 : 짝수\n"); //입력받은 수가 짝수라고 출력
36     else if (even(i) == 0) //리턴된 값이 0이면
```

(Line 25 ~ Line 51)



```
25
26 int main(void) { // main function
27
28     int i; // 입력받은 정수 저장할 변수 선언
29
30     printf("정수를 입력하십시오 : "); // 정수를 입력해라는 메시지 출력
31     scanf_s("%d", &i); // 입력받아서 i에 저장
32
33     even(i); // 입력받은 정수값을 위의 매개변수 n으로 들어가서 홀짝 구분 함수 실행
34     if(even(i) == 1) //리턴된 값이 1이면
35         printf("even()의 결과 : 짝수\n"); //입력받은 수가 짝수라고 출력
36     else if (even(i) == 0) //리턴된 값이 0이면
37         printf("even()의 결과 : 홀수\n"); //입력받은 수가 홀수라고 출력
38
39     printf("absolute()의 결과 : %d\n", absolute(i)); // 절댓값 계산하는 함수를 실행한 후 리턴된 값을 출력
40
41     sign(i); // 입력받은 정수값을 위의 매개변수 n으로 들어가서 부호 구분 함수 실행
42     if (sign(i) == 1) //리턴된 값이 1이면
43         printf("sign()의 결과 : 양수\n"); //입력받은 수가 양수라고 출력
44     else if (sign(i) == 0) //리턴된 값이 0이면
45         printf("sign()의 결과 : 0\n"); //입력받은 수가 0이라고 출력
46     else if(sign(i) == -1) //리턴된 값이 -1이면
47         printf("sign()의 결과 : 음수\n"); //입력받은 수가 음수라고 출력
48
49
50     return 0; // main function 이 0으로 반환
51 }
```



```
C:\Users\Gyuha_wa_VectOr\Documents\4\Debug\Week8 Report 4.exe
정수를 입력하십시오 : 12
even()의 결과 : 짝수
absolute()의 결과 : 12
sign()의 결과 : 양수

C:\Users\Gyuha_wa_VectOr\Documents\4\Debug\Week8 Report 4.exe
To automatically close the console when debugging stops,
Press any key to close this window.
```

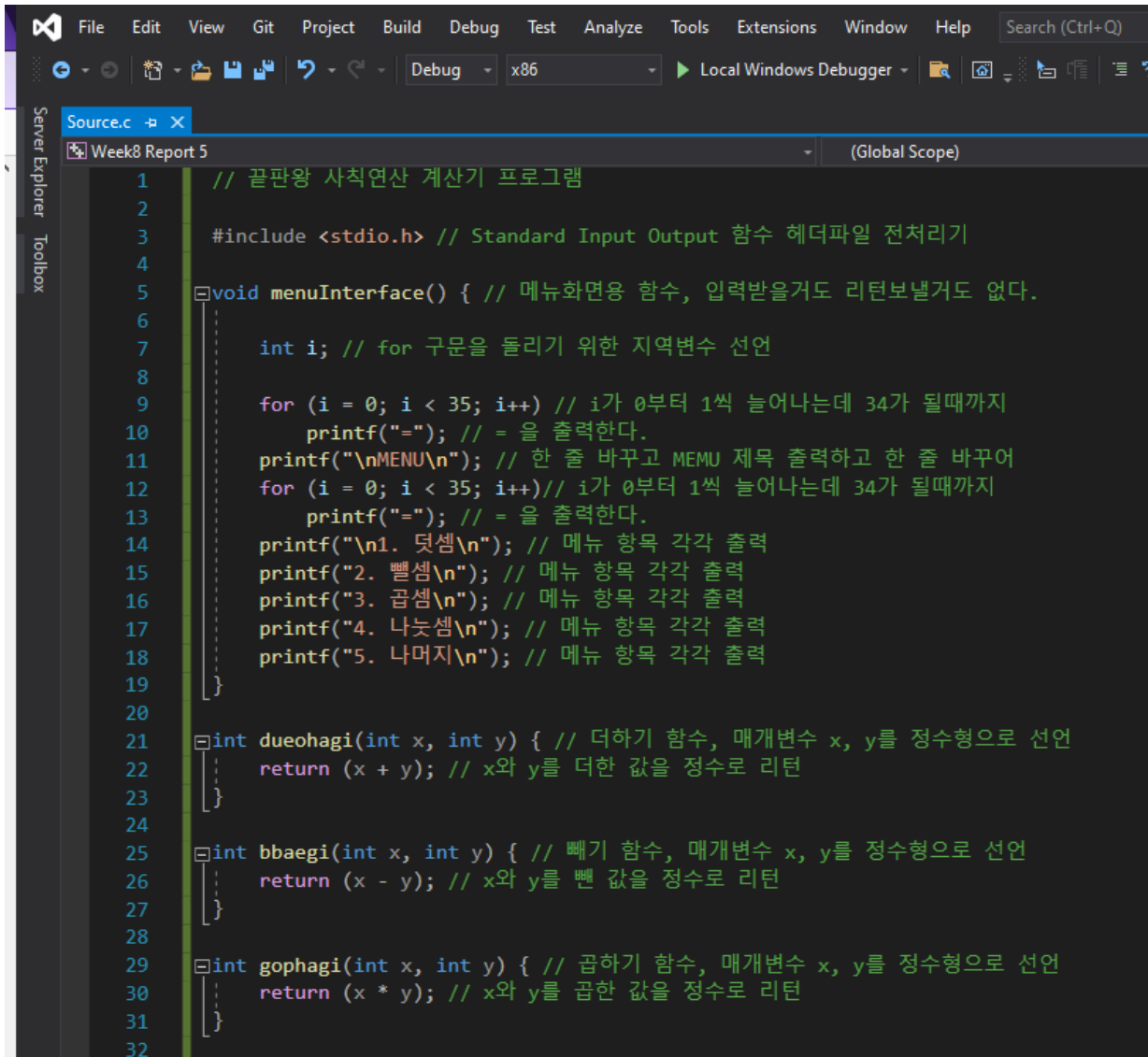
문제 5) 사용자로부터 2 개의 숫자를 받아서 덧셈, 뺄셈, 곱셈, 나눗셈, 나머지 연산의 결과를 계산해주는 프로그램을 작성해보자. 다음과 같은 메뉴를 화면에 표시한다. 프로그램 작성시에 최대한 함수를 많이 사용해보자.

Hint : 메뉴를 화면에 표시하는 함수도 만들어서 사용해보자.

- 문제 분석 & 동작 설명

솔직히 이쯤되면 계산기 사칙연산 + 나머지연산 각각을 함수로 만들면 되겠구나... 라는 생각이 들수 밖에 없을 것이다. 리턴값이 각 연산한 값으로 하여 각각의 연산 함수를 만들고, 사용자로부터 입력받은 연산 메뉴 번호에 따라 연산 함수를 골라 쓰도록 하자. 연산 메뉴 번호를 판별하기 위해서는 if 절이 빠질 수는 없을 것이다. 문제에서 함수를 많이 만들어보자 했으므로 메뉴 디자인도 리턴값 관련없는 함수로 따로 빼내어 메인함수를 좀 깨끗하게 만들어보자. 그 MENU 글 위, 아래로 있는 긴 짝대기는 반복문을 이용하여 하나의 글자를 길게 출력하도록 하자. 단, 이때는 반복문을 돌릴 지역변수가 필요할 것이다. 이 프로그램은 실행결과에서 알 수 있듯이, 한번만 하고 끝내는 프로그램이 아니다. 연산을 다 하고 사용자가 원하면 y 을 입력해 다시 메뉴 선택하도록 하게 해야되는데, 이는 do_while 문을 이용하면 될 것이다. 어쨌든 한번은 다 돌려야한다. 그리고 사용자가 마지막으로 입력한것이 조건이 맞으면 반복하고, 그렇지 않으면 프로그램을 종료해야 한다. 이러한 상황에는 do_while 문 만한 것이 없을 것이다.

- 프로그램 소스 (주석 포함)
(Line 1 ~ Line 32)



```
1 // 끝판왕 사칙연산 계산기 프로그램
2
3 #include <stdio.h> // Standard Input Output 함수 헤더파일 전처리기
4
5 void menuInterface() { // 메뉴화면용 함수, 입력받을거도 리턴보낼거도 없다.
6
7     int i; // for 구문을 돌리기 위한 지역변수 선언
8
9     for (i = 0; i < 35; i++) // i가 0부터 1씩 늘어나는데 34가 될때까지
10         printf("="); // = 을 출력한다.
11     printf("\nMENU\n"); // 한 줄 바꾸고 MENU 제목 출력하고 한 줄 바꾸어
12     for (i = 0; i < 35; i++) // i가 0부터 1씩 늘어나는데 34가 될때까지
13         printf("="); // = 을 출력한다.
14     printf("\n1. 덧셈\n"); // 메뉴 항목 각각 출력
15     printf("2. 뺄셈\n"); // 메뉴 항목 각각 출력
16     printf("3. 곱셈\n"); // 메뉴 항목 각각 출력
17     printf("4. 나눗셈\n"); // 메뉴 항목 각각 출력
18     printf("5. 나머지\n"); // 메뉴 항목 각각 출력
19 }
20
21 int dueohagi(int x, int y) { // 더하기 함수, 매개변수 x, y를 정수형으로 선언
22     return (x + y); // x와 y를 더한 값을 정수로 리턴
23 }
24
25 int bbaegi(int x, int y) { // 빼기 함수, 매개변수 x, y를 정수형으로 선언
26     return (x - y); // x와 y를 뺀 값을 정수로 리턴
27 }
28
29 int gophagi(int x, int y) { // 곱하기 함수, 매개변수 x, y를 정수형으로 선언
30     return (x * y); // x와 y를 곱한 값을 정수로 리턴
31 }
32 }
```

(Line 33 ~ Line 65)

```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl-Q) Week8 Report 5
Debug x86 Local Windows Debugger
Source.c Week8 Report 5 (Global Scope) main(void)
33 int nanugi(int x, int y) { // 나누기 함수, 매개변수 x, y를 정수형으로 선언
34     return (x / y); // x와 y를 나눈 값을 정수로 리턴
35 }
36
37 int trash(int x, int y) { // 나머지 함수, 매개변수 x, y를 정수형으로 선언
38     return (x % y); // x와 y를 나누고 난 나머지 값을 정수로 리턴
39 }
40
41
42 int main(void) { // main function
43
44     int menuNumber, a, b; // 입력받을 메뉴의 숫자와, 연산할 두개의 수 각각을 저장할 정수형 변수 선언
45     char yes; // do_while 문에서 초기 흐름 끝내고 반복 탈건지 말건지 구분하는 while절 용 변수 선언
46
47     menuInterface(); // 메뉴화면용 함수를 불러옴
48
49     do // do_while문 시작
50     {
51         printf("원하는 메뉴를 선택하십시오 (1 ~ 5) : "); // 원하는 연산 메뉴의 숫자를 입력해라는 메시지 출력
52         scanf_s("%d", &menuNumber); // 입력받은 메뉴의 숫자를 입력받아 menuNumber에 저장
53         printf("숫자 2개를 입력하십시오 : "); // 연산할 두 수를 입력해라는 메시지 출력
54         scanf_s("%d %d", &a, &b); // 띄어쓰기 단위로 입력받은 두 수를 a와 b에 각각 저장
55
56         if (menuNumber == 1) // 만약 입력받은 메뉴의 숫자가 1이라면
57             printf("연산 결과 : %d\n", dueohagi(a, b)); // 입력받은 두 수를 더하기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
58         else if (menuNumber == 2) // 만약 입력받은 메뉴의 숫자가 2이라면
59             printf("연산 결과 : %d\n", bbaegi(a, b)); // 입력받은 두 수를 빼기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
60         else if (menuNumber == 3) // 만약 입력받은 메뉴의 숫자가 3이라면
61             printf("연산 결과 : %d\n", gophagi(a, b)); // 입력받은 두 수를 곱하기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
62         else if (menuNumber == 4) // 만약 입력받은 메뉴의 숫자가 4이라면
63             printf("연산 결과 : %d\n", nanugi(a, b)); // 입력받은 두 수를 나누기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
64         else if (menuNumber == 5) // 만약 입력받은 메뉴의 숫자가 5이라면
65             printf("연산 결과 : %d\n", trash(a, b)); // 입력받은 두 수를 나머지 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
```

(Line 49 ~ Line 74)

```
File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl-Q) Week8 Report 5
Debug x86 Local Windows Debugger
Source.c Week8 Report 5 (Global Scope) main(void)
48
49 do // do_while문 시작
50 {
51     printf("원하는 메뉴를 선택하십시오 (1 ~ 5) : "); // 원하는 연산 메뉴의 숫자를 입력해라는 메시지 출력
52     scanf_s("%d", &menuNumber); // 입력받은 메뉴의 숫자를 입력받아 menuNumber에 저장
53     printf("숫자 2개를 입력하십시오 : "); // 연산할 두 수를 입력해라는 메시지 출력
54     scanf_s("%d %d", &a, &b); // 띄어쓰기 단위로 입력받은 두 수를 a와 b에 각각 저장
55
56     if (menuNumber == 1) // 만약 입력받은 메뉴의 숫자가 1이라면
57         printf("연산 결과 : %d\n", dueohagi(a, b)); // 입력받은 두 수를 더하기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
58     else if (menuNumber == 2) // 만약 입력받은 메뉴의 숫자가 2이라면
59         printf("연산 결과 : %d\n", bbaegi(a, b)); // 입력받은 두 수를 빼기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
60     else if (menuNumber == 3) // 만약 입력받은 메뉴의 숫자가 3이라면
61         printf("연산 결과 : %d\n", gophagi(a, b)); // 입력받은 두 수를 곱하기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
62     else if (menuNumber == 4) // 만약 입력받은 메뉴의 숫자가 4이라면
63         printf("연산 결과 : %d\n", nanugi(a, b)); // 입력받은 두 수를 나누기 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
64     else if (menuNumber == 5) // 만약 입력받은 메뉴의 숫자가 5이라면
65         printf("연산 결과 : %d\n", trash(a, b)); // 입력받은 두 수를 나머지 함수의 매개변수 a, b에 올려 함수를 돌려 나온 정수값 리턴받아 출력
66
67     printf("계속하려면 y를 누르세요 : "); // 연산이 끝난 후 프로그램을 계속 실행건지 물어보는 메시지 출력
68     scanf_s("%c", &yes); // 사용자로부터 문자 입력 받아 yes에 저장
69
70 } while (yes == 'y'); // 만약 사용자가 y라 입력하면 do의 첫부분으로 되 돌아 간 다.
71
72
73 return 0; // main function 이 0으로 반환
74 }
```

```
Microsoft Visual Studio Debug Console

=====
MENU
=====
1. 덧셈
2. 뺄셈
3. 곱셈
4. 나눗셈
5. 나머지
원하는 메뉴를 선택하시오 (1 ~ 5) : 1
숫자 2개를 입력하시오 : 10 20
연산 결과 : 30
계속하려면 y를 누르세요 : y
원하는 메뉴를 선택하시오 (1 ~ 5) : 2
숫자 2개를 입력하시오 : 10 20
연산 결과 : -10
계속하려면 y를 누르세요 : n

C:\Users\Gyuha_wa_Vect0r\Documents\학교\4. 대학교\충북대학교\1학년\기초컴퓨터프로그래밍 10분반\8주차\과제\5번\Week8 Report 5\Debug\Week8 Report 5.exe (process 24708) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```