

Report 1

프로그래밍 실습 1 복습

학번	2018212236
학부	전자정보통신공학
이름	김동주

제출일자	2018-09-09 (수정 전) 2018-09-16 (수정 후)
------	--

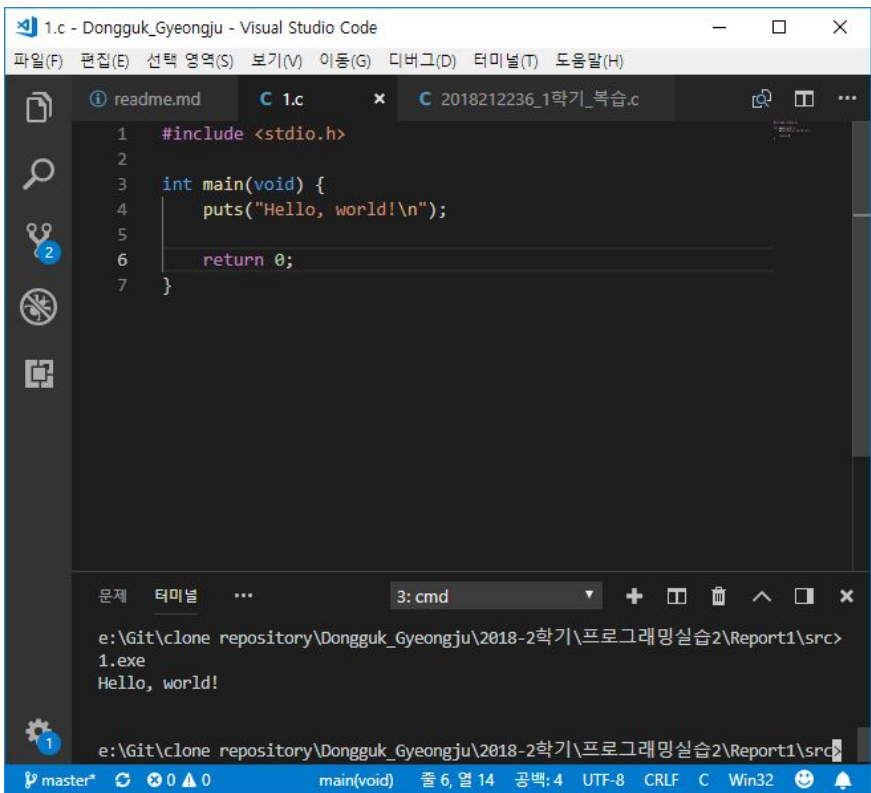
담당교수	반상우
------	-----

작업환경	Visual Studio Code
------	--------------------

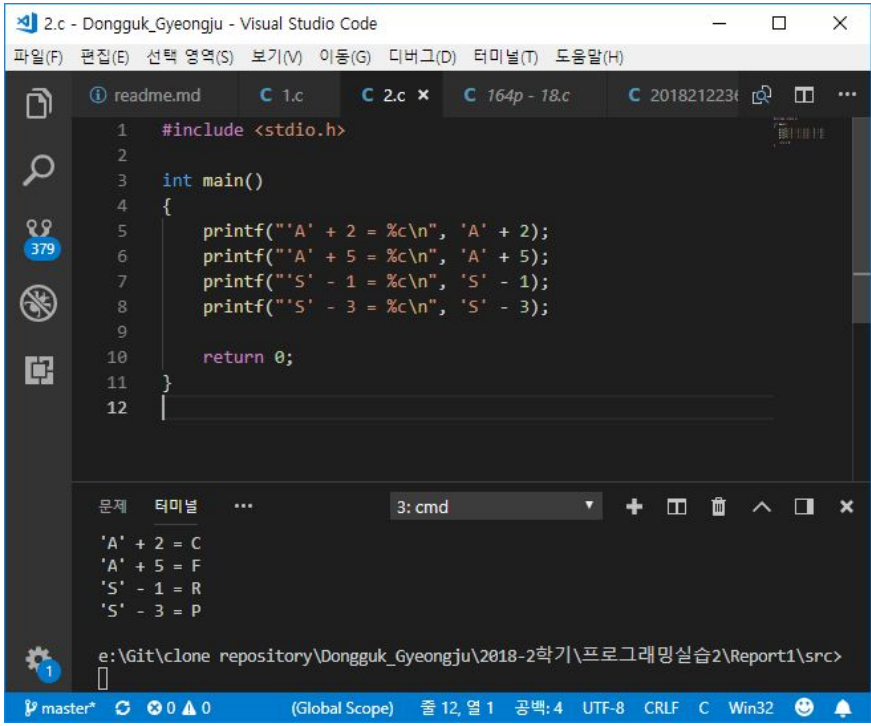
컴파일러	MinGW GCC
------	-----------

문제 내용	1학기 프로그래밍실습 1 내용 복습 - 각 단원별로 최소 1개의 프로그램 복습 - 총 10개 이상의 프로그램 복습 * 프로그램 선택은 자율 : eclass에 등록된 1학기 복습자료 또는 교재 내용 활용하여 각자에게 가장 도움이 될 수 있는 프로그램 선택.
--------------	---

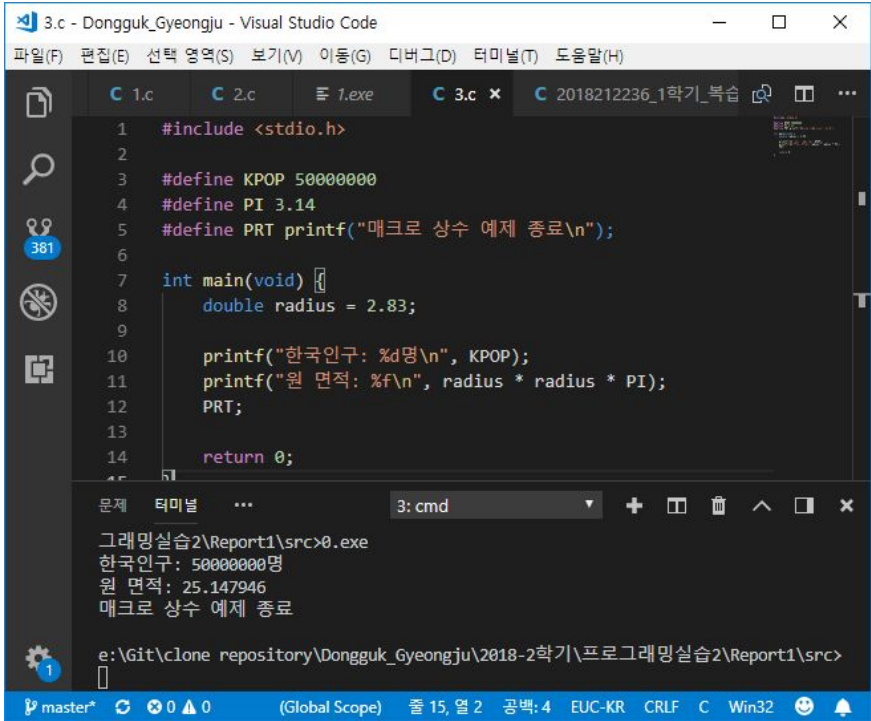
1. [Ch. 2] C프로그래밍 첫걸음

내용	Hello, world! - [p.74] putstring.c의 변형
코드	<pre>#include <stdio.h> int main(void) { puts("Hello, world!\n"); return 0; }</pre>
실행결과	 <p>The screenshot shows the Visual Studio Code interface. The editor window displays the C program code for '1.c'. The terminal window at the bottom shows the command prompt output: 'e:\Git\clone repository\Dongguk_Gyeongju\2018-2학기\프로그래밍실습2\Report1\src> 1.exe' followed by 'Hello, world!'.</p>
고찰	모든 프로그래머라면 한 번씩은 출력해 보았을 "Hello, world!" 를 출력하는 프로그램을 만들어보았다. 다시 새로 C언어를 시작하는 느낌이다.

2. [Ch. 3] 자료형과 변수

<p>내용</p>	<p>다음 조건을 만족하는 프로그램을 작성하시오.</p> <ul style="list-style-type: none"> - 문자형 연산 'A' + 2 결과를 문자로 출력 - 문자형 연산 'A' + 5 결과를 문자로 출력 - 문자형 연산 'S' - 1 결과를 문자로 출력 - 문자형 연산 'S' - 3 결과를 문자로 출력 <p>[p.164] 18번</p>
<p>코드</p>	<pre>#include <stdio.h> int main() { printf("'A' + 2 = %c\n", 'A' + 2); printf("'A' + 5 = %c\n", 'A' + 5); printf("'S' - 1 = %c\n", 'S' - 1); printf("'S' - 3 = %c\n", 'S' - 3); return 0; }</pre>
<p>실행결과</p>	 <p>The screenshot shows the Visual Studio Code interface with the C program code in the editor. The terminal window at the bottom displays the output of the program: 'A' + 2 = C, 'A' + 5 = F, 'S' - 1 = R, and 'S' - 3 = P. The status bar at the bottom indicates the file is '2.c' and the encoding is UTF-8.</p>
<p>고찰</p>	<p>교과서 164쪽의 18번 문제의 복습을 하였다. 문자형 변수에도 정수의 덧뺄셈을 할 수 있음을 다시금 떠올렸다.</p>

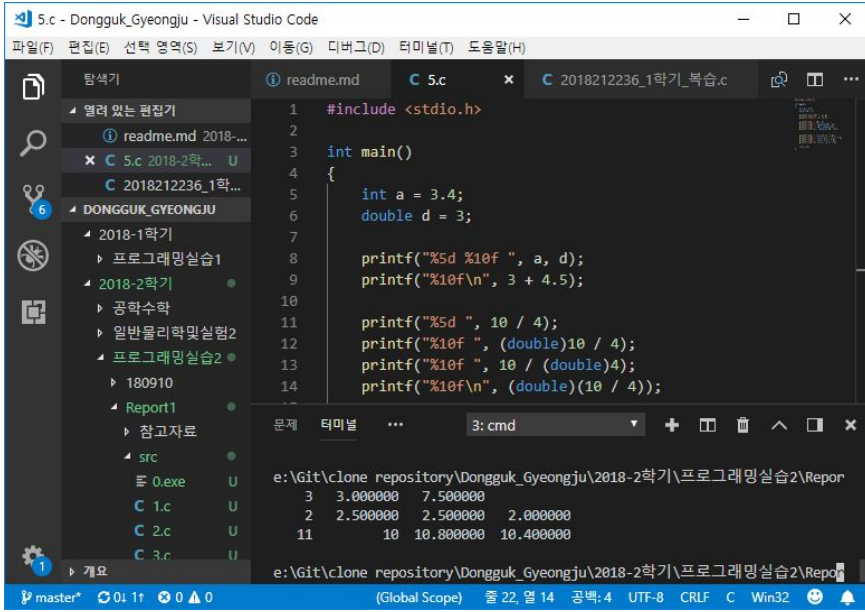
3. [Ch. 4] 전처리와 입출력

내용	[p.171] macro.c
코드	<pre> #include <stdio.h> #define KPOP 50000000 #define PI 3.14 #define PRT printf("매크로 상수 예제 종료\n"); int main(void) { double radius = 2.83; printf("한국인구: %d명\n", KPOP); printf("원 면적: %f\n", radius * radius * PI); PRT; return 0; } </pre>
실행결과	 <p>The screenshot shows the Visual Studio Code interface with the file '3.c' open. The code is as follows:</p> <pre> 1 #include <stdio.h> 2 3 #define KPOP 50000000 4 #define PI 3.14 5 #define PRT printf("매크로 상수 예제 종료\n"); 6 7 int main(void) { 8 double radius = 2.83; 9 10 printf("한국인구: %d명\n", KPOP); 11 printf("원 면적: %f\n", radius * radius * PI); 12 PRT; 13 14 return 0; 15 } </pre> <p>The terminal window at the bottom shows the output of the program:</p> <pre> 그래밍실습2\Report1\src>0.exe 한국인구: 50000000명 원 면적: 25.147946 매크로 상수 예제 종료 </pre>
고찰	<p>#define 전 처리기를 이용한 상수 사용은 1학기 활동을 통해 익숙해져 있었다. 하지만 함수를 전처리를 통해 선언하는 것에 익숙하지 않았기에, 5번 줄의</p>

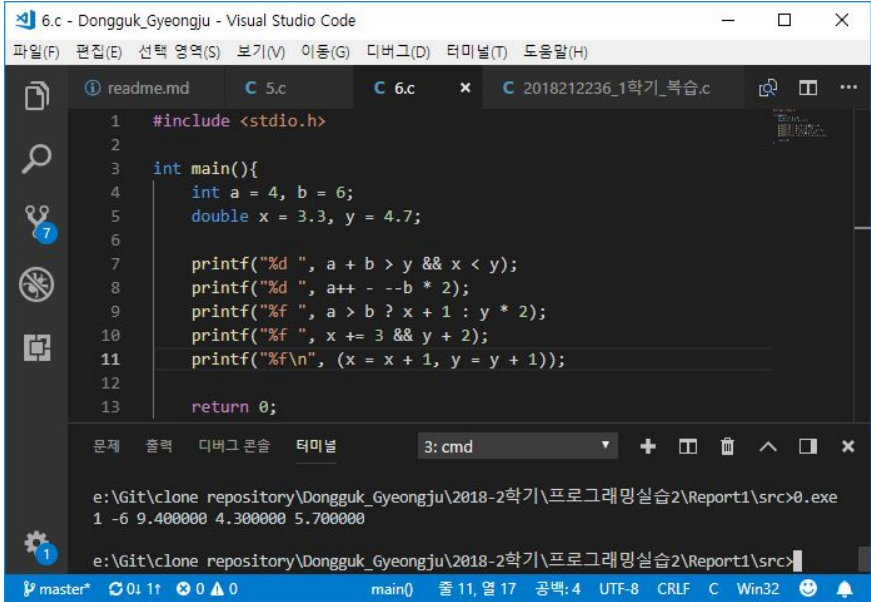
	<pre>puts(MESSAGE); return 0; }</pre>
실행결과	 <p>4.c - Dongguk_Gyeongju - Visual Studio Code</p> <pre>#include <stdio.h> #define MESSAGE "프로그래밍 언어의 학습은 일반 언어의 학습과 \ 같이 반복 학습이 중요하다" #define PI 3.14 #define VOLUME(r) (4 * PI * CUBE(r) / 3) #define SQUARE(x) ((x) * (x)) #define CUBE(x) (SQUARE(x) * (x)) #define MULT(x, y) ((x) * (y)) int main() { double radius = 2.32; 문제 출력 디버그 콘솔 터미널 3: cmd 반지름이 2.32인 구의 체적은 52.28 입니다. 실수 4.29의 제곱은 18.40 입니다. 실수 3.00의 제곱은 27.00 입니다. 실수 2.78와 실수 3.62의 곱은 10.06입니다. 프로그래밍 언어의 학습은 일반 언어의 학습과 같이 반복 학습이 중요하다 e:\Git\clone_repository\Dongguk_Gyeongju\2018-2학기\프로그래밍실습2\Report1\src\]</pre>
고찰	<p>앞선 4번 문제와 더불어 전처리를 이용하는 방법의 심화 문제이다.</p> <p>이번 문제에서는 매개변수를 사용하는 전처리 함수가 있으며, 이들이 선언 될 때에는 중괄호 블록 '{,}' 이 아닌 일반 괄호 '(,)'를 사용한다.</p> <p>잘 기억해두자.</p>

5. [Ch. 5] 연산자

내용	[p.239] typecast.c
코드	<pre>#include <stdio.h> int main() { int a = 3.4; double d = 3; printf("%5d %10f ", a, d); }</pre>

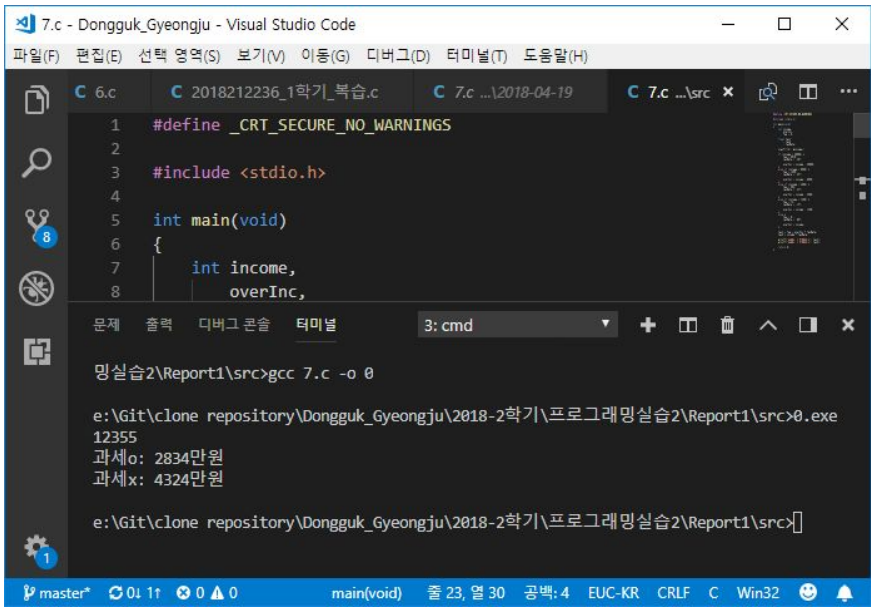
	<pre> printf("%10f\n", 3 + 4.5); printf("%5d ", 10 / 4); printf("%10f ", (double)10 / 4); printf("%10f ", 10 / (double)4); printf("%10f\n", (double)(10 / 4)); printf("%5d ", (int)(3.4 + 7.8)); printf("%10d ", (int) 3.4 + (int) 7.8); printf("%10f ", (int) 3.4 + 7.8); printf("%10f\n", 3.4 + (int) 7.8); return 0; } </pre>
<p>실행결과</p>	 <p>The screenshot shows the Visual Studio Code interface with a C file named '5.c' open. The code contains several printf statements that demonstrate the effects of different data types and casting. The terminal window at the bottom shows the output of the program, which includes integer and floating-point values formatted as specified in the code.</p>
<p>고찰</p>	<p>계산을 함에 있어서 매우 중요한 형변환에 대하여 복습하였다. 형 변환 순서에 따라 값이 다르게 출력되는 것을 보면서 코드를 작성할 때 형변환 사용 여부, 순서를 매우 신중하게 결정해야 한다는 것을 느꼈다.</p>

6. [Ch. 5] 연산자 (2)

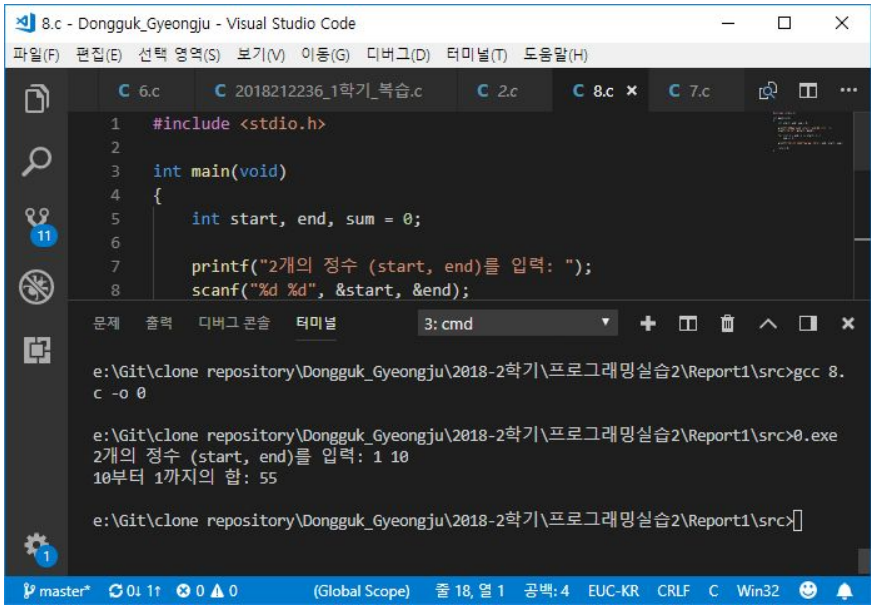
내용	[p.245] priority.c
코드	<pre> #include <stdio.h> int main(){ int a = 4, b = 6; double x = 3.3, y = 4.7; printf("%d ", a + b > y && x < y); printf("%d ", a++ - --b * 2); printf("%f ", a > b ? x + 1 : y * 2); printf("%f ", x += 3 && y + 2); printf("%f\n", (x = x + 1, y = y + 1)); return 0; } </pre>
실행결과	 <p>The screenshot shows the Visual Studio Code interface with the file '6.c' open. The code is the same as in the previous block. The terminal output is as follows:</p> <pre> e:\Git\clone repository\Dongguk_Gyeongju\2018-2학기\프로그래밍실습2\Report1\src>0.exe 1 -6 9.400000 4.300000 5.700000 </pre>
고찰	<p>꼼꼼한 비교를 통하여</p> <ul style="list-style-type: none"> - 산술 > 관계 > 논리 - 단항 > 곱셈 > 뺄셈 - 산술 > 관계 > 조건 - 산술 > 논리 > 대입 - 괄호 > 산술 > 대입 > 쉼표 <p>의 순서로 실행 됨을 확인하였다.</p>

7. [Ch. 6] 조건

내용	세울 계산
코드	<pre>#define _CRT_SECURE_NO_WARNINGS #include <stdio.h> int main(void) { int income, overInc, fee = 0; float tax1 , tax2 , taxRate; scanf("%d", &income); if (income > 30000) { fee = 9010; taxRate = .38f; overInc = income - 30000; } else if (income > 8800) { fee = 1590; taxRate = .35f; overInc = income - 8800; } else if (income > 4600) { fee = 582; taxRate = .24f; overInc = income - 4600; } else if (income > 1200) {</pre>

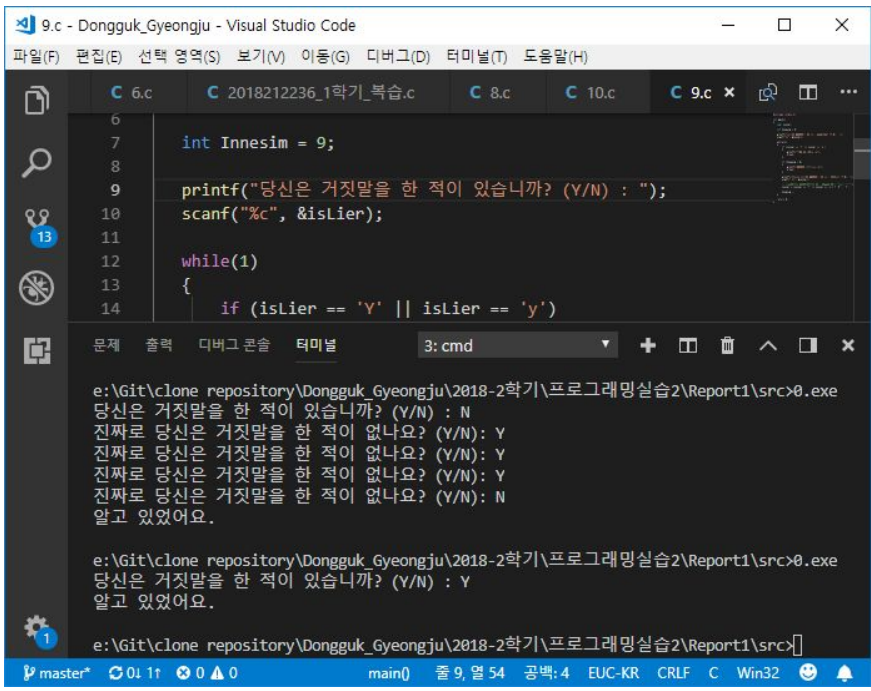
	<pre> fee = 72; taxRate = .15f; overInc = income - 1200; } else { fee = 0; taxRate = .06f; overInc = income; } tax1 = fee + overInc * taxRate; tax2 = income * taxRate; printf("과세o: %.0f만원\n", tax1); printf("과세x: %.0f만원\n", tax2); return 0; } </pre>
실행결과	 <p>The screenshot shows the Visual Studio Code interface with a C program being executed. The code defines a main function that calculates taxes based on income. The terminal output shows the execution of gcc 7.c -o 0, followed by the program output: 과세o: 2834만원 and 과세x: 4324만원.</p>
고찰	<p>소득 비용에 따른 서로 다른 세율을 계산하는 예제를 통해 if, else 조건문에 익숙해졌다.</p>

8. [Ch. 7] 반복

내용	start 부터 end 까지의 합
코드	<pre> #include <stdio.h> int main(void) { int start, end, sum = 0; printf("2개의 정수 (start, end)를 입력: "); scanf("%d %d", &start, &end); for (int i = end; i >= start; i--) sum += i; printf("%d부터 %d까지의 합: %d\n", end, start, sum); return 0; } </pre>
실행결과	 <p>The screenshot shows the Visual Studio Code interface with the C program open. The terminal window at the bottom shows the following output:</p> <pre> e:\Git\clone repository\Dongguk_Gyeongju\2018-2학기\프로그래밍실습2\Report1\src>gcc 8. c -o 0 e:\Git\clone repository\Dongguk_Gyeongju\2018-2학기\프로그래밍실습2\Report1\src>0.exe 2개의 정수 (start, end)를 입력: 1 10 10부터 1까지의 합: 55 e:\Git\clone repository\Dongguk_Gyeongju\2018-2학기\프로그래밍실습2\Report1\src> </pre>
고찰	<p>for문을 활용하여 입력받은 start부터 end까지의 합을 구하는 코드를 작성하였다.</p> <p>반복 연산자 중에서는 역시 for문이 가장 깔끔하고 이쁘다고 생각하였다.</p>

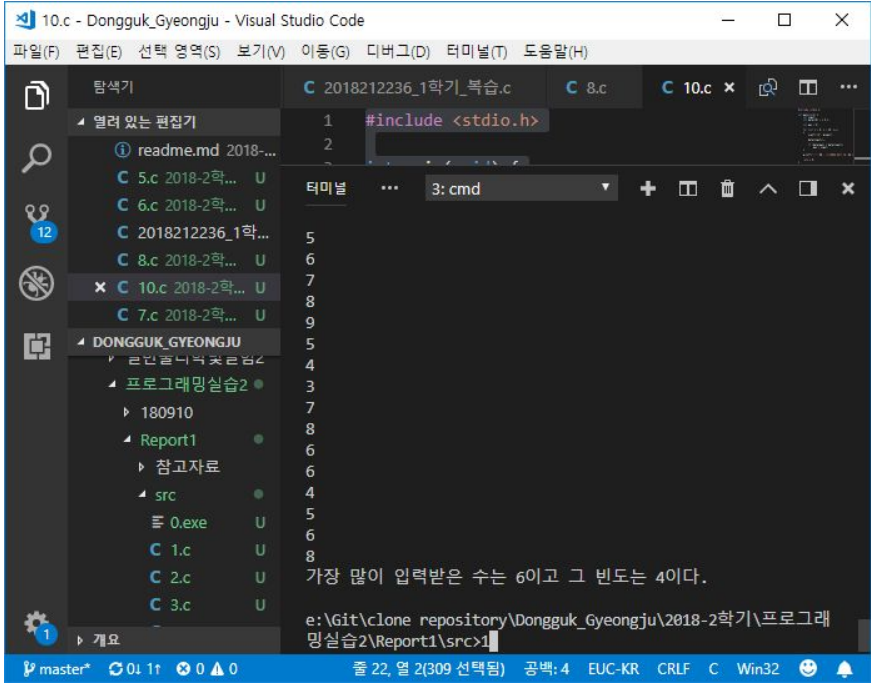
9. [Ch. 7] 반복 (2)

내용	나를 속이는 거짓말
코드	<pre>#include <stdio.h> int main() { char isLier; int Innesim = 9; printf("당신은 거짓말을 한 적이 있습니까? (Y/N)\n"); scanf("%c", &isLier); while(1) { if (isLier == 'Y' isLier == 'y') { printf("알고 있었어요.\n"); break; } if (Innesim < 0) { printf("거짓말 치지마요.\n"); break; } printf("진짜로 당신은 거짓말을 한 적이 없나요? (Y/N) : "); scanf(" %c", &isLier); // 부정형으로 질문하였으니 한국 문법에 맞게 대답을 // 역전 시켜준다. isLier = (isLier == 'Y' isLier == 'y') ? 'N' : 'Y'; } }</pre>

	<pre>Innesim--; } return 0; }</pre>
실행결과	 <p>The screenshot shows a Visual Studio Code window with a C file named '2018212236_1학기_복습.c'. The code defines a variable 'Innesim' and a loop that asks the user if they have lied. The terminal output shows the program running and the user entering 'N' and 'Y' multiple times.</p>
고찰	<p>반복문을 통해 원하는 입력을 받을 때 가지 질문을 반복하는 재밌는 프로그램을 만들었다.</p> <p>삼항 연산자 및 조건문, 반복문에 익숙해 졌음을 느꼈다.</p>

10. [Ch. 8] 배열

내용	빈도 출력
코드	<pre>#include <stdio.h> int main(void) { int input; int data[10] = { 0 }; int max = 0; for (int i = 0; i < 20; i++)</pre>

	<pre> { scanf("%d", &input); data[input]++; if (data[max] < data[input]) max = input; } printf("가장 많이 입력받은 수는 %d이고 그 빈도는 %d이다.\n", max, data[max]); return 0; } </pre>
실행결과	
고찰	<p>배열을 선언하고 값을 0으로 초기화하여, 증감연산자를 통해 빈도수를 계산하는 프로그램이었다.</p> <p>입력받을 수 있는 최대 숫자에 해당하는 크기만큼 선언되어야 하고, 사용되지 않는 원소가 있을 수 있다는 단점을 해결 할 수 있으면 좋겠다고 생각하였다.</p>