**진도 보고서**

**IT공학과 박미현**

# 목차

**(1) 자신이 도움을 준 사람의 명단과 도움의 내용, 시점**

**(2) 자신에게 도움을 베푼 사람 명단과 받은 도움의 내용, 시점**

**(3) 과제 수행을 위해 참고한 문헌이나 인터넷주소: 그 참고 내용과 범위**

**(4) 투자한 순수 집중 시간 및 과제 결과로 본, 본인이 원하는 과제 점수**

**(5) 과제 수행 과정을 상세히 기록한 시작~종료 날자 별 데일리 리포트**

(1) 자신이 도움을 준 사람의 명단과 도움의 내용, 시점

\* 도움을 준 사람의 명단: **불특정 소수**

\* 시점: **상시**

카카오톡 오픈채팅에는 해당 수업에 대한 질문 사항 같은 것들을 공유하는 그룹채팅방이 존재한다. 이 단톡방에는 약 50명 정도가 있는데 아마 기초 프로그래밍 02반을 수강하는 학생이라면 대부분 참여하고 있을 것이다. 이 그룹채팅 방에는 포인터에 대한 질문이나 과제 문제에 대한 질문들 등등이 주기적으로 올라오곤 한다. 나는 상시적으로 단톡방에 올라오는 질문에 대한 질문에 가끔 친절한 답변을 해준 바 있다. 실질적으로 불특정 학우들에게 도움이 됐을지는 모르겠지만, 나로 인해 도움이 되었었음 한다.

나에게 물어보면 열정적으로 대답해줄 수 있는데, 사실 나에게 도움을 청한 사람이 아예 없었다. 해당 수업에서 나의 친구는 오직 '이은정' 학우 뿐이다. 은정 학우도 알아서 잘 하는 사람이기 때문에 나에게 딱히 큰 도움을 구하지 않았던 것 같다. 나에게 질문을 좀 해줬으면 좋겠다.

(2) 자신에게 도움을 베푼 사람 명단과 받은 도움의 내용, 시점

\* 내게 도움을 베푼 사람: **이은정**

\* 시점: **11/26(월)**

이 시점 당시 나는 3번 문제만 건너뛰고 1, 2, 4번 문제를 끝내거나 마무리하고 있던 상태였다. 3번을 건너뛴 이유는 바로, 한 부분에서 또 분명히 막힐 것이라는 사실을 잘 알고있었기 때문이다. 나는 5점짜리 워밍업 과제를 하면서 이미 우선순위 계산기를 구현해본 경험이 있다(한 부분을 실패해서 결국은 제출 못했지만). 그래서 나의 걸리는 부분이 어디인지, 어디 부분을 좀 더 다뤄서 완성해야 할지 잘 알고 있다. 일단은 그 부분에 대해서는 나중에 곰곰히 생각해보자라는 마음으로 3번을 미뤄둔 상태였다.

3번에서 단 한 부분, 내가 해결하지 못했던 부분은 바로 '두 자리 이상의 피연산자로 계산하기' 부분이었다. 사용자로부터 수식으로 된 문자열을 입력 받고, 그 중위표기법으로 기술된 수식을 후위 표기법으로 변환하고, 그 변환된 후위표기법 수식을 스택을 이용하여 계산하여 결과를 산출하기까지 잘 따라왔다. 그런데 단 한가지 문제는 한 자리 수의 피연산자로만 계산이 가능하다는 점이였다. 이것이 5점 워밍업 과제에서 완벽한 코드를 제출하지 못했던 이유이다. 그 당시에 일단 점수는 받아야 하니 출력 되게만 해서 제출했는데, 이 부분은 지금까지 해결하지 못한 채로 남아있었다.

그러던 어느 날! 평소와 같이 월요일 기초 프로그래밍 수업이 있는 날 이였다. 수업 시간 도중에 교수님께서 atoi 함수에 대해 언급하셨다. 우리가 지금 하고 있는 과제에 이 함수의 사용이 나올 것이라는 그런 말씀이셨다. 나는 그 순간 2번 문제를 말씀하고 계시는건가 라고 생각했다. 나는 2번 다른 접근 방법으로 풀기 전에, 처음에는 문자열로 다루어서 atoi를 사용한 바 있다. 코드가 너무 지저분해 보이고 복잡해질까봐 결국 그 방법을 사용하지 않고 결국 다른 방법으로 바꿨었다. 그래서 '나는 atoi를 쓰지 않았는데.. 2번이 아닌건가? 다른 어느 문제에서 atoi함수를 써야하는 부분이 있었나?'하며 혼란에 빠지기 시작했다. 수업시간 도중 잠깐의 쉬는 시간이 생겼다. 그래서 쉬는 시간을 틈타 **내 바로 옆에 있는 '이은정'이라는 친구에게 과제 중에 atoi함수를 써야 될 문제가 있었는지 물어보았다. 그랬더니 친구는 3번 문제에서 사용할 것이라고 대답했다. 처음에 의구심이 들었고, 3번에 atoi를 사용해야하는 이유를 파악하지 못했다. 은정이는 atoi는 아주 유용하게 쓰이는 함수라며 아주 고마운 함수라고 웃으면서 대답했다. 그 순간 내 머릿 속에 엄청난 시나리오가 그려졌다.** 그 짧은 시나리오는 다음과 같다.

*(1) strtok()를 이용하여 문자열을 토큰으로 분리한다. (피연산자와 연산자를 분리하기 위함. 이 렇게 하면 피연산자를 두 자리 수 이상이든, 한 자리 수 이상이든 알맞게 분리가 가능하다.)*

*(2) 분리한 문자열 세그먼트를 모두 atoi()를 거쳐 숫자로 변환한다.*

*(3) 숫자로 변환된 것들을 차례차례 새롭게 선언한 int형 배열에 저장한다.*

*(4) 이제 그 배열에 있는 요소들을 가지고, 스택을 이용한 계산기를 구현하면 된다.*

(1)~(3)까지의 시나리오가 머릿 속에 주마등처럼 스쳐지나갔다. 그 때 엄청난 새로운 깨달음을 얻었다. 문자열 토큰 기법, atoi()를 사용하여 '두 자리 수 이상의 피연산자로 계산하기'를 가능하게 하다니. **사실은 쉽게 생각할 수 있던 방안이었는데, 본인은 이 방법에 대해서는 전혀 생각치 못했다. 이 단순한 것을 왜 혼자서는 생각하지 못했던 것일까? 은정이를 통해 엄청난 큰 영감을 얻은 것이다. 은정이는 별 생각없이 대답한 것이였겠지만, 내게는 한 문제의 완성이 걸린 문제였다. 은정이 덕분에 3번 문제를 이제서야 완벽하게 완성할 수 있겠구나 라고 생각되어 마음이 놓이기 시작했다. 이미 3번 문제는 내 마음 속에서 완성되었다. 너무 기뻤다.**

**이 외에도 나는 다른 문제들에 대해서도 '이은정'이라는 친구에게 몇 번 더 물어본 적이 있다. 직접적으로 코드를 공유하며 도움 받고 그런 것은 아니였지만, 문제의 방향성을 잡는 데에 큰 도움을 받을 수 있었던 것 같다. 그리고 문제를 풀다가도 뭔가 이상하다는 생각이 들면 과제에 대해 가끔 서로 이야기 해보기도 했다. 이를 통해 큰 위안을 받을 수 있었다.**

**무엇보다 3번 문제에서 '유레카'를 발견할 수 있었던 계기는 바로 '이은정'이라는 친구의 존재 덕분이였다.**

(3) 과제 수행을 위해 참고한 문헌이나 인터넷주소: 그 참고 내용과 범위

**<1번>**

- 오로지 내 머리를 바탕으로 코딩하였다.

**<2번>**

- 음식을 만드는 데 필요한 재료와 양 (네이버 블로그 검색을 바탕으로 음식에 관한 데이터를 구성하였음)

**<3번>**

- 스택 알고리즘, 후위표기법으로의 전환 알고리즘 및 후위 표기법 수식 계산 알고리즘

(윤성우 저자의 열혈 자료구조 책에서 배운 것을 이용)



P213~218 에서 스택 연결 리스트 기반 구현 코드 참고

P219~247 을 공부하며 우선 순위 계산기 프로그램 작성

**<4번>**

- 정렬 코드 알고리즘 ( http://itng.tistory.com/92 사이트에서 참고)

- 파일에서 데이터 읽어오고 파일에 출력하는 방법 (여러 번의 구글 searching을 통해 파악)

(4) 투자한 순수 집중 시간 및 과제 결과로 본, 본인이 원하는 과제 점수

25만점 중 23점(?!)

* **투자한 순수 집중 시간 self 평가**

솔직히 나는 정말 열심히 했다고 말할 수 있다. 충분한 기간을 두고 프로젝트를 진행하였고, 프로그램 하면서 부딪히는 내적 갈등 과정들도 정말 많이 겪었다. 또한, 그러면서 나의 실전 코딩 능력도 많이 향상 되었음을 직감하였다.



위 사진은 #3 진도 보고서를 작성하는 데에 나의 노력이 얼마나 투자되었는지를 증명하는 사진이다. 11/23부터 데일리별 리포트를 작성하기 시작했고, 일기를 쓰면서 나의 문제점이 무엇이고 다음에는 무엇을 해결해야 할지 파악을 했다.

뿐만 아니라, #2 블록 레포트 또한 정말 심혈을 기울여 작성하였다. 내가 구성한 프로그램의 의미들이 잘 전달될 수 있도록 하는 마음에서 나름 내 기준으로 성실히 작성하였다. 특히 3번 같은 경우, 알고리즘은 모두 내가 기술한 것은 아니지만 완벽히 이해하고, 그 이해한 내용을 바탕으로 열심히 블록 보고서를 작성하였다.

그리고 프로젝트를 시작한 날 이후부터 끊임 없이 계속 텀 프로젝트에 대한 생각을 놓지 않았다. 코딩을 하다가 어느 문제에 직면하면 바로바로 해결해버리고 싶은 충동과 욕심에 거의 다음 날, 매일매일을 시도하고 고민하며 살았던 것 같다.

* **과제 결과 self 평가**

일단, 1번과 2번은 완벽한 것 같다.

3번에서 해결하지 못한 것은 ‘두 자리 이상의 피연산자로도 계산할 수 있게 하기’ 이다. 좋은 방법도 생각해내고, 구상도 하고, 여러 번 시도도 해보았지만 결국은 해결하지 못했다. 하지만 그것을 제외하면 모두 완벽하다. 알고리즘은 비록 카피하였지만, 그 내용(스택 이용, 알고리즘, 프로그램 흐름, 구조체 포인터 개념 이용 등등)을 완벽하게 이해하고 프로그램을 구성하였다. 따라서 3번에서는 그 부분 때문에, 총 6점에서 1~2점 정도를 감점하면 되지 않을까 싶다.

4번도 1,2번과 같이 문제에서 제시한 모든 것들을 완벽하게 해냈다. 파일에서의 입출력도 모두 가능하게 만들었고, 항목 번호를 줄 때마다 정렬도 완벽하게 출력해낸다. 한 가지 아쉬운 점은 메인 함수가 너무나 길어진 점이다. 사실, 정렬 코드를 함수 처리 하고, 그 함수에 대한 인자로 student구조체 배열을 넘기려고 했는데 생각보다 마음처럼 되지 않았다. 그 부분을 함수 처리하지 못하여 메인 함수가 너무 많이 길어졌다. 일단 코드를 보기좋고 깔끔하게 해놓았긴 했는데, 메인코드는 여전히 너무 길다. 실전 개발자로서의 자질은 부족한 것 같다. 하지만 학생 수준에서는 그렇게 크게 문제되는 것 같지는 않다. 사실 문제에서 제시한 모든 요건은 다 충족시켜 놓았기 때문이다. 이는 아쉬운 점일 뿐이고, 실제로 구현을 하지 못한 것은 아니라서 점수를 깎을 필요는 없을 것 같다.

나머지 보고서 등등의 요인으로 추가적으로 깎일 수도 있다고 생각한다. 양심없지만 3번에서 해결하지 못한 한 부분에서 -2를 하여 23점을 외쳐본다.

(5) 과제 수행 과정을 상세히 기록한 시작~종료 날자 별 데일리 리포트

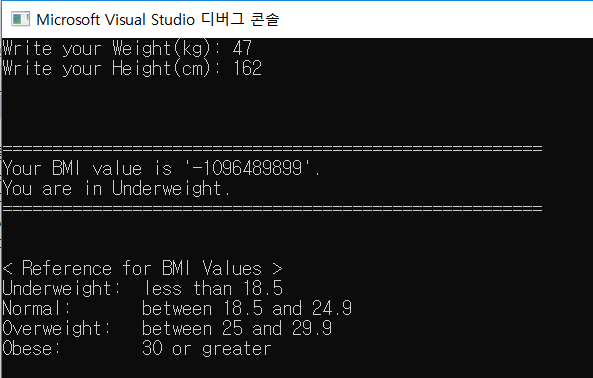
**11/20(화)**

***프로젝트를 시작하다. 1번 문제부터 시작!***

프로젝트의 3, 4번 문제들이 hell이라는 친구의 말을 듣고, 우선 얼른 1번과 2번을 후딱 끝내버려야겠다는 생각이 들었다. 그래서 바로 실행에 옮겼다.

1번은 BMI값을 계산하여, 이 수치가 어느 범위에 속하는지 출력시키면 되는 문제이다. 이미 BMI 구하는 식이 문제에 명시 되어있어 어떻게 프로그램을 구성해 나갈지에 대해서만 잘 고민하면 되는 문제였다. 바로 함수를 사용해야겠다라는 생각이 들었고, 크게 어렵게 느껴지지 않은 문제였다.

다만, BMI수치 결과 값이 정확히 나오지 않는 문제에 계속 부딪혔다. 나는 실행결과 화면에 저체중이냐, 비만이냐를 출력시키기 전에, BMI 결과값도 볼 수 있게끔 출력시키도록 하였다. 나의 키와 몸무게를 입력하면 BMI가 18.~이 나와야 하는데 자꾸 틀린 수치(마이너스 값)가 떴다. 초반에 계속 컴파일하면서는 결과 값이 정확히 잘 나와줬었는데, 함수 2개를 더 추가시키고 나니 내가 원하는 결과값이 나오지 않았다. syntax적인 오류는 없었다. 어디가 잘못된 것일까. 졸리니까 일단은 자야겠다. 1번 프로그램은 전체모습을 구성하기까지 대략 한 시간 걸린 것 같다. 오류는 다음날 해결해볼 것이다.



(원하는 BMI결과 수치가 나오지 않는 문제의 화면 1)

**11/21(수)**

***1번 문제에서 생긴 문제를 해결하다.***

오늘 수업시간에 교수님께서 저번시간에 배웠던 '값에 의한 호출', '참조에 의한 호출'을 다시 한 번 언급해 주셨다. 수업을 들으면서 내가 코딩한 1번 문제의 오류가 여기서 발생한 것 같다고 생각했다. 따라서 수업 끝나고 바로 나의 1번 프로그램의 함수 부분을 수정하며 컴파일을 반복해 보았다.

그러던 순간, 눈에 밟히는 이상한 곳이 있었다. 바로 printf문 안의 형식지정자를 %d로 쓴 것이었다. 나는 사용자로부터 입력 받는 키, 몸무게, BMI수치를 모두 실수형으로 통일 시키는 것이 좋다고 판단하여 모든 변수들을 실수형으로 통일시켰었다. 그런데 BMI결과값을 출력시키는 프린트 문에서 형식지정자를 %lf가 아닌 %d로 써버린 것이다. 모든 변수들을 double형으로 선언 해놓고 결과 값 역시 소수로 계산까지 해놓고는 정수형으로 출력시키려 하다니 당연히 값을 잃을 수 밖에...

이보다 허무할 수가 있나! 이렇게 1번문제의 오류를 10분 만에 바로 해결하였다. 마음이 편해졌다. 어제는 보이지도 않던 오류가 오늘은 한 눈에 들어왔다. 내내 붙잡고 있는 것보다 휴식을 취하고 다시 보는 것도 좋은 방법인 것 같다.

**11/22(목)**

***2번 문제를 시작하다. 어떻게 프로그램을 짤지 구상해보자***

* ***(1) 구조체 개념으로 음식 데이터 관리를 해보자.***

2번 문제를 시작했다. 대충 얼핏 보고 당연히 쉬운 문제일 것이라고 생각했는데, 막상 시작하려니 프로그램을 어떻게 짜야 할지에 대한 감이 단번에 오지 않았다.

전체적인 구성은 잘 잡히지 않는다. 일단은 한 개의 음식을 입력하면 음식을 만드는 데에 필요한 재료들과, 그에 대응되는 양을 출력시키도록 해야 한다. 그래서 ‘구조체’개념을 떠올렸다. 왜냐하면, 한 개의 음식 정보에는 <(1)음식의 번호(숫자int형), (2)음식 이름 정보(문자), (3)음식 재료들(문자열 배열), (4)재료 각각에 대응되는 양(숫자형 배열) > 이렇게 들어가야 할 것 같기 때문이었다. 그런데 구조체를 사용하려니 부딪히는 생각들이 많아졌다. 막상 코딩해보니 더 난잡해졌고, 외관상으로도 깔끔한 코드가 되지 못했기 때문이다. 우선 굳이 음식의 번호, 음식 이름을 포함시킬 필요가 있을까..? 또, 재료와 양을 쌍으로 이루고 싶은데, c에서는 파이썬에서처럼 key와 value값의 개념도 존재하지 않는다. 물론 함수를 내가 따로 정의하여 쓰는 방법도 있겠지만, 이 길은 아닌 것 같다는 생각이 들었다. 게다가 재료 각각에 대한 양은 ‘몇 개’, ‘몇 그램’ 처럼 단위 역시 다 다르다. 그래서 구조체로 구현하려 했던 코딩을 다 지워 버렸다.

프로그램을 어떻게 짤지 구상만 하다가 시간이 다 간 것 같다. 빠른 시간 안에 끝낼 수 있을 줄 알았는데 그 기대가 좌절되니 자괴감이 들기 시작했다.

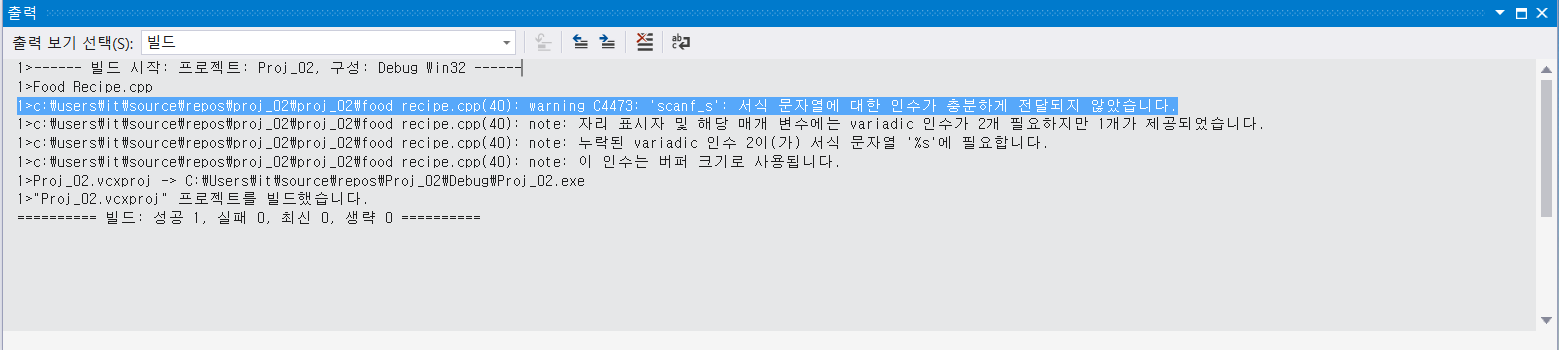
**11/23(금)**

***2번 문제, 어떻게 프로그램을 짤지 다시 한 번 구상해보자.***

* ***(2) 문자열 이중배열 변수에 음식 정보들을 저장해놓자. 그 이후 그 문자열 각각 에 대해, 사람 수만큼 곱하기 위한 처리를 진행해 보도록 한다.***

어제의 다음 날인 오늘, 또 다른 생각이 들었다. 바로 ‘문자열 이중배열’의 개념을 이용하자! 문자열 이중배열 변수를 char ingredients\_of\_foods[num\_of\_food][100]; 이렇게 선언한다. 행은 음식의 개수이다. 즉, 행 하나 당, 하나의 음식 정보를 담고 있는 문자열 하나(“~~”)를 갖고 있는 것이다. 예를 들어, ingredients\_of\_foods[0]은 **"라면:1 개, \n 물:100 grams, \n 파: 20 grams, \n 계란: 1개"** 라는 문자열을 가리키고 있기 때문에, ‘라면’음식 하나에 대한 정보를 갖고 있다고 생각하면 된다. 이렇게 문자열 이중 배열을 이용하여서 문자로 처리하는 방법을 택했다. 이렇게 하니 코드도 더 깔끔해지고, 다음엔 어떻게 해야할지 감이 잡혔다. 이 방식을 이용하여서, 일단 ‘라면’을 입력하면 문제에서 제시된 보기대로 출력되도록 만들었다.

이제는 ‘1인분 재료의 양’에 ‘먹는 사람의 수’만큼의 수만큼 곱하여 출력시키도록 해야 한다. 다 문자열로 처리해버렸는데.. 어떻게 숫자만 바꿔서 출력시키지? 또 머리가 새하얘졌다. 시간이 많이 지났으니 내일 다시 생각해보도록 해야겠다.



(사용자로부터 문자열을 입력 받았을 때 생긴 오류)

참고로, 사용자로부터 음식 이름을 입력 받으면 그에 해당하는 처리를 할 수 있도록 다중if-else문을 사용했는데 여기서도 문제가 생겼었다. 사용자에게 음식 이름들이 적힌 메뉴판을 보여주고, 만약 사용자가 ‘라면’을 입력하면, 해당 if문으로 가서 그와 관련된 처리를 진행하도록 해야 한다. 하지만 ‘라면’을 입력하고는 내가 원하는 처리 화면이 전혀 뜨지 않았다.

그 문제의 원인은 바로 if문 안을 ‘if(str=="라면")’ 이런 식으로만 단순하게 코딩했기 때문이다. 이는 옳지 않다. 대신, if ( strcmp(input\_food, "라면") == 0 ) 이렇게 쓰도록 해야 한다. 문자열 복사 개념을 이용해야 하는데, 이는 인터넷을 써칭하며 알게 된 사실이다. 수업 시간에도 언급된 바 있다. 사소한 부분일 수도 있겠지만, 문자열 복사의 중요한 개념을 제대로 알고 갈 수 있게 된 소중한 시간이 되었다.

오늘 5시간 가량 정도를 2번 문제에 투자하였다. 2번이 생각보다 너무 많은 시간이 걸려 마음이 급해졌다.

**11/24(토)**

***2번 문제의 어느 부분이 계속 걸린다. 따라서 2번은 나중에 건들기로 하고,***

***4번 문제를 시작하게 되다. 4번 중 파일의 모든 데이터를 읽어 오기까지 완료하다.***

**<2번 문제>**

여태까지는 2번 문제에서 제시된 보기대로 출력되는 모습을 잘 보여주고 있는 상태이다. 이제부터는 ‘먹는 사람의 수’ 만큼 곱해서 잘 출력하면 된다.

이를 어떻게 하면 될지에 대해 학교 오고 가면서 구글 검색을 좀 했다. 그래서 ‘strtok’ 함수를 써야겠다는 생각을 했는데, 나의 원래 의도와는 성질이 다른 함수인 것 같아 과감히 다른 방식을 알아보기 시작했다.

나는 문자열에 있는 숫자들을 각각 추출해서, 숫자로 바꿔서, 거기에 ‘먹을 사람의 수’만큼 곱하여, 다시 각 숫자들이 포함된 전체 문자열을 출력하도록 해야 한다. 따라서, ‘strchr’함수를 이용해보기 시작했다. ‘strchr’ 함수는 인자로 받은 문자열에서, 인자 문자 하나가 나타나는 위치를 반환하는 함수이다. 즉, 문자열 속에서 내가 원하는 ‘문자 하나’의 위치를 반환해준다. 그 위치를 따라가서 요소 값을 숫자로 바꾸고(문자니까), 그 숫자에 ‘먹는 사람 수’ 를 곱하고, 그 수를 문자열에 다시 출력하도록 하는 것이 나의 목적이다. 이를 위해 각 과정마다 프린트문을 삽입하여 잘 진행되는지 확인하였다. 문자열 하나 속에서 내가 다루고 싶은 문자의 위치를 반환 받고, 그 위치에 있는 문자열 요소 하나 값을 atoi를 이용하여 숫자로 변환한 뒤, 이 숫자에 또 ‘먹을 사람 수’를 곱하여 \*inum에 대입하였다. \*inum에 대입함으로써 inum주소가 가리키는 위치에 있는 요소 값을 변화시킨 것이다. 그리고 원래의 문자열 하나를 출력하여서, 내가 원하는 위치에 있는 문자가 잘 바뀌어졌는지 확인해보았다.

그런데!! 내가 변경하고 싶은 문자위치 부분이 자꾸 이상한 문자로 바뀌어 출력되었다. 중간 중간 프린트문을 넣어서 확인해본 결과, 바꾸고 싶은 문자의 위치 값도 잘 반환하였고, 포인터 변수가 가리키고 있는 그 위치의 문자 요소를 숫자로 변환까지 잘 됐다. 거기에 ‘먹을 사람 수’를 곱한 결과의 숫자 값도 잘 출력되었다. 여기까지는 잘 되고 있음을 확인했지만, 원본 문자열이 잘 바뀌었는지 확인 하기위해 문자열을 출력해보면 그 부분이 이상한 문자로만 출력이 되었다. 어디가 문제일까?

포인터가 가리키는 주소를 따라가서 그 요소를 변경하면 되는 줄 알았는데( \*inum = 원하는 값 ), 처음에는 이게 일시적으로만 변화시키는건가 하고 생각하여 온갖 검색을 다 해보았다. 무엇이 문제인지 계속해서 과정을 곱씹어봤다. 문제의 원인은 바로 내가 숫자로 다 처리 해놓고, 숫자인 것을 아무 처리 없이 바로 문자로 출력되게 했으니 이런 문제가 생긴 것이다. 문자열 안에 있는 한 부분을 추출하여, 숫자로 바꾸고, 그 수에 먹을 사람 수만큼 곱해주고, 이는 숫자이니 다시 문자열로 출력하기 위해서는 문자로 바꿔주어야 한다.

다시 문자로 바꾸는 처리를 해야하는데.. 또 머리가 새하얘졌다. 이 과정을 숫자 나올 때마다 해줘야 한다고..? 너무나도 비효율적이다. 문자로 바꾸기를 시도하다가 결국 뇌가 거부하여 2번을 잠시 쉬기로 결정했다.

**<4번문제>**

2 번 문제를 오늘 끝내 버리려고 했는데 계속 어느 부분이 막혀서 결국 다른 문제를 다루고 다시 돌아가기로 하였다. 3번 문제는 5점짜리 워밍업 과제를 하면서 꽤 오랜 시간 건드려본 적이 있다. 3번은 한 부분 빼고 일단은 어느정도 구현할 수 있으니, 먼저 4번부터 풀기로 결정하였다.

4번 문제는 파일에서 데이터들을 읽어와 그 데이터들을 정렬 처리한 후, 원하는 결과를 파일 또는 실행결과에 출력하는 문제였다. 일단은 학생 한 명당 정보가 9개 들어있어야 하니 ‘구조체’ 개념을 이용해야겠다는 생각이 제일 먼저 들었다. 학생 한 명 당 성명, 나이, 전화번호, 점수 등등의 서로 다른 자료형이 필요하기 때문이다.

구조체를 사용하기 앞서, 일단 텍스트파일에 있는 데이터들을 모두 잘 읽어와야 한다. 그래서 구조체 배열부터 먼저 선언하지 않고, 일단 구조체 변수 하나만 먼저 선언하여 텍스트 파일 한 줄만 읽어오는 것부터 시작하였다(처음에는 한 명의 이름, 성별, 나이 이렇게 3개의 데이터만 가져옴). 계속 컴파일을 진행해보면서 데이터가 잘 가져와지는지 확인을 하면서, 데이터 양을 점점 늘려갈 예정이다.

그런데!! 파일 읽어오는 것인 초반부터 계속 헤맸다. 구글에 검색하여 내가 직면한 오류가 무엇인지 계속 해결하려고 노력했다. 프로젝트 속성도 건드려보고 계속 고민을 하였다. 꽤 오랜 시간이 지나더니 문득 예전 파이썬 수업시간에 배웠던 것 하나가 떠올랐다. 바로 내가 텍스트 파일을 해당 소스가 담긴 폴더 속에 넣지 않은 것이었다. 텍스트파일 위치 경로를 배경화면에 놓고서 데이터들을 읽어오려 하다니… 그렇게 그 문제는 스스로 잘 해결해냈다.

파일 읽기를 성공시키고나서, 텍스트 파일에 있는 데이터들을 잘 읽어오는지 확인하기 위하여 프린트문을 이용하여 계속 출력시켜보았다. 계속 컴파일을 반복 진행하며 서서히 가져오는 데이터들을 확장시켜 나갔다. 그렇게 총 (10명) x (한 명당 9개의 정보) 해서, 총 90개의 데이터들을 모두 텍스트 파일에 입력하고 잘 읽어오도록 만들었다.

이제는 주어진 기준대로 각각에 대한 정렬을 시작하면 된다. 버블정렬, 선택정렬 등등 많은 정렬 방법이 존재한다는 사실을 잘 알고있지만, 현재는 예전에 배웠던 기억들이 다 사라진 상태이다. 이 파트는 정렬 공부를 더 하고 이따가 다시 건드리도록 한다.

오늘 보고서까지 7시간 넘게 노트북을 붙잡고 있으면서, 진도가 꽤나 나갔다. 비록 2번의 앞 길은 까마득하긴 하지만, 4번 문제의 입력을 파일로부터 받게 할 수 있게 되어 약간 뿌듯하다.

**11/25(일)새벽**

***4번 문제, 파일에서 읽어온 데이터들을 가지고 정렬을 시작해보자.***

옛날에 c언어 몇 회독한 적이 있다. 그때 ‘정렬’과 ‘탐색’에 대해서도 유심히 공부했던 기억이 난다. 그래서 옛날 c언어 책을 펴고 정렬 파트를 찾았다. 나에게는 ‘선택 정렬’이 가장 익숙하다. 가장 기본적인 정렬 방법이기도 하고 그래서 나는 내림차순 또는 오름차순을 하기 위해 ‘선택 정렬’을 택하기로 하였다.

참고로 나는 1. 숫자를 정렬하는 함수(점수, 나이 같은 것을 다루기 위해), 2.문자를 정렬하는 함수(이름, 대학 같은 것을 다루기 위해). 이렇게 두 함수를 구성해볼 예정이다.

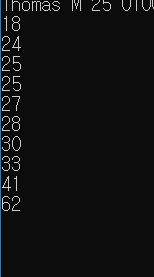
그래서 오늘 새벽에는 숫자를 정렬하는 코드, 문자를 정렬하는 코드 이렇게 두 파트를 코딩해보았다. 아직까지 함수를 구성하지는 않았고, 일단은 메인 함수에 모두 코딩을 해 놓은 상태이다. 이제 프로그램을 깔끔하게 하기 위하여 이 둘을 함수 2개로 따로 정의하여서, 메인 함수에서 각각 기준번호에 따라 필요한 함수를 호출하여 프로그램을 짜보도록 할 예정이다.

**11/25(일) 아침**

***4번 문제, 그런데, 내가 원하는 정렬의 모습이 나오지 않는다.***

빨리 4번을 끝내 버리고 싶었다. 그래서 숫자를 정렬하는 코드, 문자를 정렬하는 코드 이렇게 두 개에 대한 함수 각각 만들기를 시도하였다.

그런데 문제가 생겼다. 나는 바로 전에 정렬을 하기위한 코딩을 마무리 했었다. 메인코드에 써놓은 코드들을 함수로 정의하여 정리하려고 하던 찰나, 내가 했던 정렬 방식에 문제가 있다는 사실을 알아차리게 되었다.



바로 위와 같은 정렬의 모습이 출력되었다. 내가 원한 것은 한 기준에 의해 각 사람의 모든 정보(이름, 성별, 나이, 성적 등)가 담긴 문자열도 다같이 일률적으로 정렬시켜버리는 것이였다. 하지만 나는 점수면 점수들만, 이름이면 이름들만 정렬이 되도록 만들어버렸다. 너무 생각없이 정렬을 시도한 나의 문제였다.

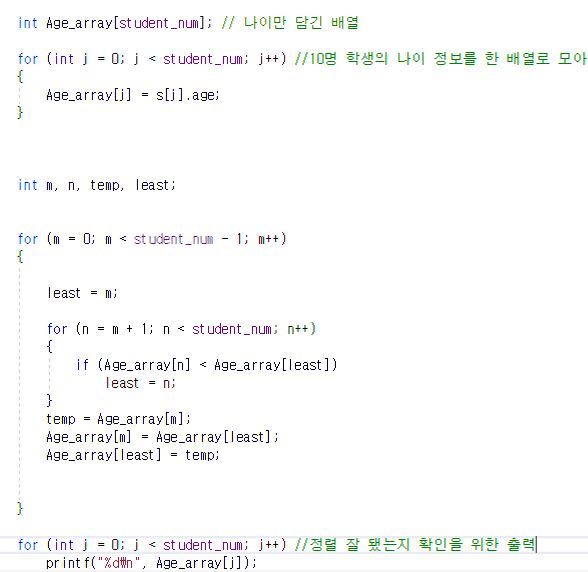
예를 들어, 위의 그림처럼 나이 순으로 정렬을 시킬 때, 학생 10명의 나이 변수들만 모아서 정렬 하여 나이만 출력하게 하면 안 된다. 나이 순으로 정렬시키더라도, 나머지 이름, 성별, 학교명, 성적의 정보도 같이 일률적으로 정렬되어 출력되어야 한다. 이를 위해 정렬을 다시 구상해보아야 한다. 아침에 빨리 끝내버리려다가 또 난관에 부딪혔다. 다시 머나먼 길을 걸어야 할 것 같다.

**11/25(일) 오후**

***4번 문제 도 막막해 졌다. 또다시 2번으로 돌아가서 곰곰히 고민해보자.***

***그렇게 했더니 어떻게 2번을 완료하게 되다.***

정렬을 어떻게 다시 손봐야 할까. 코드를 계속 바라봤다. 정렬의 처음부터 잘못됐다.



나는 오름차순을 하기 위해 ‘선택정렬’ 알고리즘을 이용했고, 위와 같이 코딩을 진행했었다. 나이 순으로 정렬하기 위해서 애초에 10명 학생 각각의 나이 정보만 모아 정렬을 진행했다. 애초에 나이만 모아 배열에 넣고 이를 대상으로만 정렬하게 한 것이 잘못이었다. 곰곰히 생각해보아도 명확한 해답이 떠오르지 않았다.

그래서 일단 다시 또 2번 문제로 돌아갔다. 나는 과거에 2번 문제 어느 부분에서 장시간 막혀있었다. 다시 구상했던 방식 역시 좋은 방법은 아닌 것 같다는 생각이 들었다. 결국, 노가다를 시전하기로 결정했다. 도저히 생각해도 2번 문제를 기발하게 코딩하는 법은 잘 모르겠다. 원래 노가다하는 문제였던가? 차라리 노가다 하는 것이 코드가 더 깔끔할 것 같았다. 그래서 노가다를 했다. 2번 문제를 어떻게 풀었는지에 대한 설명은 ‘진도 보고서’가 아닌 ‘함수 보고서’에서 기술해놓도록 한다. 간단히 말하자면 그냥 프린트 문으로 하지는 않았고, 구조체를 이용하여 숫자 문자 숫자 문자 숫자…. 이렇게 음식 데이터를 저장하는 방식으로 노가다를 진행하였다.

그렇게해서 2번을 끝냈다. 10개의 음식 데이터를 입력 시키고, 노가다를 하니 시간이 꽤나 걸렸다. 오늘은 8시간 가량 노트북을 붙잡고 있었다. 아무튼 2번의 막이 내렸다. 얼른 다시 4번을 해결해보도록 해야겠다.

**11/26(월) 새벽**

***4번 문제에서, 드디어 정렬을 내가 원하는 모습대로 출력하게 하다.***

오늘 새벽, 묵혀 두었던 4번의 문제점을 이제서야 해결해냈다. 이제 내가 원하는 모습대로 정렬시킬 수 있게 되었다.

그리고 사실 숫자, 문자형을 따로 분리하여 정렬을 진행할 필요도 없었다. 나는 이전에 숫자는 숫자대로, 문자형은 문자대로 따로 구분해서 각각의 정렬을 처리하는 함수 2개를 만들려고 했었다. 하지만 숫자형이든 문자형이든 모두 같은 정렬 알고리즘을 적용해도 된다는 사실을 깨달았다. 따라서 정렬을 처리하기 위한 함수 1개만 정의해주어도 된다. 이렇게 하면 코드도 훨씬 깔끔해지고 필요한 코드만 쓰도록 하여 효율적일 것이다.

오늘은 이름, 나이, 대학, 전화번호, 국어, 영어, 수학, 과학점수 이렇게 총 8개 기준을 가지고 8번의 정렬을 모두 진행해 보았다. 일단은 모두 메인코드에 코딩해놨고, 8개 기준에 따라 각각 잘 정렬되는 모습을 눈으로 확인하였다. 8개의 정렬 모두 같은 정렬 알고리즘을 사용하니, 이제는 하나로 일반화 시키는 작업을 하면 된다. 다시 말해, 정렬을 처리하는 함수를 따로 만들고, 메인함수에서 각각 기준 번호가 주어질 때마다 함수를 호출하여 원하는 정렬을 진행할 예정이다.

정렬에서 막혀서 4번 문제를 어떻게 해야될지 정말 막막했는데, 직면하는 문제를 뚫고 유레카를 외칠 수 있게되어 너무 기쁘다. 그냥 자려니 찝찝해서 못자겠고..그래서 다시 일어나서 했는데 풀리다니, 너무 뿌듯하다

**11/28(수)새벽**

***반복되는 코드들을 함수처리하여 프로그램을 깔끔하게 만들려는 전략 구상 / 실행결과 화면이 아닌, 파일로의 출력을 시도해보다.***

나는 각 기준대로 메인함수에서 총 8번의 정렬을 진행하였다. 그랬더니 같은 코드가 8번이나 반복돼서 메인함수가 엄청 길어졌다. 이 반복되는 똑같은 코드 8부분을 함수 하나로 정의하여 코드를 깔끔하게 만들고싶었다. 정렬을 담당하는 함수를 만들기 위해서는 매개변수의 인자로 구조체 배열을 넘겨주어야 한다. 그런데 사실 구조체 배열을 바로 인자로 보내는 법을 잘 모르겠다. 책 보니까 다음 수업시간에 배울 파트가 함수 인자로 구조체 변수를 넘기는 것이던데, 구조체 배열을 인자로 넘기는 방법에 대해서도 수업시간에 한 번 질문해보아야야겠다. 일단은 함수처리는 냅두고 파일에 출력하는 부분부터 먼저 다뤄보도록 한다.

여태까지는 각 기준에 따라 정렬 되는 모습이 실행화면에 잘 출력된 모습의 상태이다. 이제 파일로도 출력을 하게끔 만들어야 한다. 그래서 fscanf함수를 사용하여 파일에 출력하도록 했다. 형식도, 코드도 모두 맞게 타이핑 하였다. 그런데 계속 텍스트 파일에는 모습 그대로 남아있는 상태이다. 왜 안되는지 잘 모르겠다.

책을 읽다가 fopen 부분에서 모드를 지정해줄 때 “r”뿐만 아니라, “a”, “w” 등의 모드도 존재한다는 사실을 깨닫게 되었다. 나는 읽기 모드로만 파일을 열었으니까 당연히 파일에 써지지 않는거겠지? 라고 생각하였다. 그래서 기존 내용에 8번 정렬되는 모습들을 더 추가시키기 위해 “a”모드로 수정하였다. 그랬더니 실행결과화면에 알 수 없는 이상한 문자들이 마구 등장했다. 그래서 이번에는 “w”쓰기모드로도 해봤다. 역시나 같은 이상한 결과화면이 뜬다. 그.런.데. 텍스트파일의 상태를 계속 확인해보고 있었는데, 다시 열어보니 모든 내용이 지워져 있었다.. 힘들게 10명 정보 입력해놨는데.. 이렇게 입력해놓은 파일 내용을 한번에 다 지워버리다니. 다시 입력하면 되는 일이긴 한데.. 상당히 귀찮은 일이 발생했다. 특히 “w”모드는 쓰기 모드인데 파일이 이미 존재하면 내용을 덮어쓴다고 한다. 따라서 모드에 대한 정확한 지식을 갖추어야 앞으로 이러한 실수는 다시 겪지 않을 것이다.

그런데 왜 파일에 출력이 안되는 것일까? 해결하지 못한 채 잠에 든다. 파일의 지워진 데이터들은 다시 입력하고 잤다.

**11/29(목)새벽**

***4번 문제, 이제 파일에서도 출력을 가능하게 만들다.***

이제 4번 문제의 실행결과 화면 뿐만 아니라 파일로의 출력 역시 가능하도록 만들었다. 학교 가면서 인터넷 검색을 조금 해봤는데 아무리 생각해도 fopen에서 모드를 지정해줄 때 문제가 생긴 것 같다(어제 생각했던대로). 그래서 오늘 다시 노트북을 켜고 모드를 바꿔보았다. 내가 사용해야 할 모드는 아마도 “a”이거나 “a+” 인 것 같았다. 왜냐하면 나는 파일을 열어 그 파일의 끝에 값을 이어쓰는 것을 목적으로 하기 때문이다. “a”는 어제 한 번 입력했다가 결과 화면에 이상한 문자들이 등장했어서, “a+”부터 먼저 써보았다. 그런데!! 내가 원하던대로 파일에 출력이 되는 것 아니겠는가! 내가 원하던 대로, 텍스트 파일의 내용 끝에 정렬된 코드들이 이어서 써졌다. 혹시나 해서 모드를 “a”로도 바꿔보았다. 그랬더니 어제와 같이 또 결과 화면에 이상한 문자들이 등장했다. “a+”과 “a”는 둘 다 공통적으로 ‘추가’의 개념이 들어있어서 아무거나 써도 상관 없을 줄 알았는데 아니었다. 파일의 모드에 대해서 잘 알 수 있게 되었던 기회가 되었다. 파일로의 출력까지, 4번의 뼈대가 전반적으로 다 갖추어진 것 같다. 오늘은 별 거 안했는데 그냥 기쁘다.

참고로 4번 문제의 반복되는 긴 정렬 코드들을 함수처리 하려고 시도했는데 끝내 성공하지 못했다. 구조체 배열을 함수 인자로 보내서 처리하면 되는건 알겠는데, 막상 정렬 코드를 수정하려니 난해하다. 구조체 배열이므로 학생 한 명 한 명 넘기기 위해서는 구조체 배열을 가리키는 포인터 ptr++; 을 해주면 된다. 그런데 정렬 코드의 포문에서 다루는 것은 I j인데 포인터로 건너뛰려니 엄청 난해했다. 계속 깊게 생각해보았는데 결국 성공하지는 못했다.

**11/30(금)**

***4번 문제를 마무리하다. 그리고, 3번 문제를 시작하다.***

오늘은 4번 문제를 마무리 시킨 날이다. 이전에 3개 가량의 기준까지만 정렬해놓은 상태였는데 오늘은 마저 남은 기준들에 대한 정렬들까지 마쳤다. 그렇게해서 (2)번을 제외하고, 모든 항목에 대한 정렬에 대한 코딩을 마쳤다. 그리고 모든 정렬된 모습을 실행 결과 화면 뿐만 아니라, 파일로도 출력이 가능하게끔 만들었다. 보기좋게 다시 한 번 정리하고, 각 과목에 대한 평균도 출력하게끔 하여 4번 문제를 완료하였다. 조금 수정해야 할 부분이 더 있는데, 3번이 너무 급해서 시간이 남으면 더 정갈하게 하도록 하겠다.

그리고 오늘, 묵혀두었던 3번 문제를 다루기 시작했다. 앞서도 말했듯이, 5점짜리 워밍업 과제를 하면서 ‘우선순위 계산기 구현 프로그램’에 대해서는 이미 많은 시간을 투자한 바 있다. 3번의 전체 알고리즘과 어떻게 파일과 함수를 구성 해야할지에 대해서 전반적으로 잘 알고 있는 상태이다. 내가 한 가지 구현하지 못한 부분은 바로 ‘한 자리 수가 아닌 두 자리 이상의 피연산자를 받도록 하기’이다. 그래서 계산기 프로그램을 전체적으로 모두 구성하고, 그 문제의 부분도 다루기 시작했다.

3번에서 사용자로부터 수식으로 된 문자열을 입력받도록 한다. 이는 문자열이다. 따라서 두 자리의 피연산자의 숫자들이 따로 분리되지 않도록 문자열을 토큰으로 분리할 계획이다. 그러기 위해 strtok함수를 이용하였다. 하지만 strtok함수를 사용하기에 익숙해질 때까지 조금은 오랜 시간이 걸렸다. 토큰 분리법에 대해 완벽하게 숙지를 했을 때쯤, 나는 또 난관에 부딪혔다. 토큰으로 분리한 문자열을 어느 또 다른 배열에 저장해 놓는다. 그런데 배열 자체는 주소를 상징한다. 즉 포인터 변수인데, 나는 3번 프로젝트에서 함수의 인자를 넘길 때 모두 char형 문자열 수식으로 넘기는 것으로 통일시켰다. 전자는 포인터 형태이고, 후자는 포인터 형태가 아니다. 둘의 형식이 다르기 때문에, 토큰을 다루고 나면 그 이후 함수들 간의 호출 시에 인자가 제대로 넘어가지 않게 되어 이상한 결과값이 뜨게 된다. 여기 부분에서 계속 씨름하다가 결국 오늘 내가 원하는 부분을 처리하지 못하게 됐다.

거의 3번과 4번을 하루 가까이 넘게 붙잡고 있었는데, 3 번의 길이 보이지 않는다. 스트레스 받기 시작했다.

**12/2(일)**

***진도 보고서 마무리하다. / 3번 처리하지 못한 부분 !!!***

이제 슬슬 보고서들도 완성시켜야 한다. 오늘은 진도 보고서를 작성하였다. 진도 보고서의 (5)항목 데일리별 리포트는 프로젝트 할 때마다 작성해놓았다. 나머지 항목에 대한 보고서를 작성하였다.

그리고 어제 3번에서 해결하지 못한 부분과 다시 씨름하기 시작했다. Strtok를 이용하여 토큰을 분리하여 새로운 배열에 저장하면(그렇게 하면 두 자리 이상의 피연산자를 떨어뜨리지 않고 붙여놓을 수 있다.), 그 배열은 주소를 상징하기 때문에 포인터이다. 하지만 나는 파일에서 함수의 인자를 거의 모두 ‘char형 문자열 ’로 넘기는 것으로 통일시켰었다. 서로 다른 형이 일치하지 않기 때문에, 여기서 막혀서 여기 코드 이후로 더 이상 함수의 인자가 제대로 넘어가지 않는다. 그래서 모든 파일의 함수 매개변수를 배열 즉, 포인터로 통일시키기로 결정하였다. 이전 소스코드들을 백업 시켜놓고, 다른 이름으로 또 저장해 거기서 새로운 마음으로 코드들을 다시 다뤘다. 하지만, 결국은 산으로 갔다. 점점 더 오류 투성이가 된 것 같다. 결국 체념하고 3번의 두 자리 이상의 피연산자 다루기를 포기했다. 조금 더 시간이 지나서 다시 다뤄보면 다른 새로운 깨달음을 얻을 가능성도 있다. 따라서 시간이 지나고 다시 다뤄보도록 하고 오늘은 이만 다시 접는다. 6시간 가량 투자한 것 같은데, 성과가 없어서 우울하다.

**12/3(월)**

***1, 2 번에 대한 블록 레포트 작성***

1, 2번 프로그램의 #02블록 보고서를 작성하였다. 소스 캡쳐, 소스캡쳐마다의 설명, 실행결과까지 설명하니 시간이 꽤나 오래 걸렸다. 최종 코드 정리와 주석까지 정리하니 대략 7시간 가량 앉아 있었다.

**12/4(화)**

***4 번에 대한 블록 레포트 작성***

4번 프로그램의 #02블록 보고서를 작성하였다. 수업이 저녁까지 있어 공강시간마다, 그리고 새벽 시간을 활용하여 4번에 대한 블록 레포트 작성을 마쳤다. 이전에 그냥 바로 모든 번호에 대한 정렬 모습을 출력시키는 것까지만 해놨었다. 오늘은 사용자가 항목 번호를 주면 정렬하도록 코드를 수정 시켜 프로그램을 완벽하게 완성하였다. 이렇게 오늘은 4번 주석+ 코드 수정 + 코드 다듬기+블록 레포트 작성을 진행하였다.

**12/5(수)**

***3 번에 대한 블록 레포트 작성***

오늘은 텀 프로젝트 제출 날이다. 마지막으로 3 번 프로그램에 대한 블록 레포트 작성을 마쳤다. 모든 파일들에 대한 설명을 마쳤고, 결과 화면까지 마무리 했다. 3프로젝트에는 9개의 파일이 존재한다. 파일들이 꽤나 많아 설명하는 데 많은 시간이 걸렸다. 4시~9시까지 이 작업들을 진행하였다. 오늘 시간이 남으면, 3번의 두 자리 이상의 피연산자 받기까지 마무리해보려 했는데, 결국에는 시간이 남지 않았다. 아쉽지만 텀 프로젝트는 여기에서 종료해야 할 것 같다.