

REPORT

보고서 작성 서약서

1. 나는 타학생의 보고서를 복사(Copy)하지 않았습니다.
2. 나는 타학생의 보고서를 인터넷에서 다운로드 하여 대체하지 않았습니다.
3. 나는 타인에게 보고서 제출 전에 보고서를 보여주지 않았습니다.
4. 보고서 제출 기한을 준수하였습니다.

나는 보고서 작성시 위법 행위를 하지 않고,
성.균.인으로서 나의 명예를 지킬 것을 약속합니다.

과 목 : 프로그래밍 기초와 실습
과 제 명 : 실습 3 : flow of control
담당교수 : 민 형 복 교수
학 과 : 자연과학계열
학 년 : 1학년
학 번 : 2014310407
이 름 : 이준혁
제 출 일 : 2014년 10월 10일

1, Introduction

이번 실습에서는 flow of control에 대해 공부한다. 저번 실습에서 배운 것을 더 확장해서 다룬다. 저번 실습에서는 조건문인 if문만 이용했지만, 이번 실습에서는 더 많은 반복문, 조건문을 다룬다. flow of control에는 if 문처럼 특정 조건에서만 실행되기도 하고, while문처럼 특정 조건을 만족할 때까지 실행하는 것도 있다. 그리고 이 두 가지 문장을 합쳐놓은 것과 유사한 for문도 있고, switch문, do-while 문 등도 있다. 또, break 와 continue라는 statement를 이용해 loop내에서 실행문이 어떻게 실행되는 지를 완전히 바꿀 수도 있다. 심지어 각 문장들 안에 다른 문장들을 넣어서 다양하게 실행해 볼 수도 있다.

이번 실습 문제는 이런 flow of control을 이용해 두 학점의 평점 평균을 계산하는 프로그램을 만들어 볼 것이다. 이 때 다양한 flow of control을 이용해 프로그램이 정상적으로 작동하도록 해 볼 것이다.

이런 flow of control을 잘 사용하면, 이전까지 배웠던 것들로는 구현할 수 없는 여러 프로그램을 만들 수도 있고, 만들기 힘들었던 것들 역시 간단하게 만들 수 있다. 그러므로 이번 실습은 굉장히 중요하다고 할 수 있겠다.

2. Problem Statement

(A) Describe what the problem is

영어와 수학의 학점을 A, B, C, D, F 중에 하나로 입력 받아서, 이 학점을 각각 4.0, 3.0, 2.0, 1.0, 0.0의 평점으로 환산 한 후에 두 평점의 평균을 구하는 프로그램을 짜는 문제이다. 단, 영어는 if문, 수학은 switch문을 이용해야 한다. 이 때, do-while문장을 이용해 이 과정을 두 학점 중 어느 하나라도 'E'가 입력되기 전까지는 계속해야 한다. 또, 사용자가 A, B, C, D, E, F중 하나가 아닌 다른 입력을 준다면 오류 메시지를 보내고 다시 입력하도록 요구해야 한다.

(B) Describe how you solve the problem.

학점을 입력 받아 평점의 평균을 구하는 것이므로, 학점 변수 2개(각각 영어, 수학)와 평점 변수 2개의 선언이 필요하다고 생각했다.

그 다음에 do-while 문장으로 시작할 것이라 생각했다. 적어도 한 번은 학점을 입력 받아야 하니 말이다. 학점을 입력을 받고 난 이후에는 입력 받은 학점 중에서 하나라도 E가 있는 지 판별해야 한다고 생각했다. 그래서 if문을 이용해 E가 하나라도 있으면 do-while문을 빠져나가기로 했다.

그 다음에는 원래 학점이 A, B, C, D, F가 아닌 경우에 오류 메시지를 주고 입력을 다시 요구하려 했다. 그러나 그렇게 표현하는 것 보다 학점이 A, B, C, D, F인 경우를 생각하고 그것이 아닐 때만 오류 메시지를 출력하는 것이 낫다고 생각 해

그렇게 하기로 했다. 학점이 A, B, C, D, F인 경우에 그에 맞는 평점을 평점 변수에 넣기로 했다. 만약 이 경우가 아니면 일단 오류 메시지를 출력한 다음에, 어떻게 할 지 고민했다. continue문을 쓸까 고민도 했었지만, 별로인 것 같았다. 그래서 평점 변수에 음수 값을 넣기로 생각했다. 평점 출력은 평점이 모두 양수일 때만 출력하면 된다고 생각했기 때문이다.

얘기한대로 if문을 이용해 평점에 음수가 없는 경우, 평점이 모두 양수여서 정상적으로 입력이 된 경우만 평균을 출력하면 됐다. 그래서 if문을 이용해 학점 입력이 제대로 된 경우만 평균을 출력할 것이라 생각했다. 아닌 경우는 그대로 놔 두기로 했다.

그리고 while문을 검사하는 식에 1을 두기로 했다. 왜냐하면 여러 번 검사를 해야 했기 때문이다. 그리고 E일 때 탈출해야 하므로, 위에서 E일 때 break문을 이용해 탈출하기로 했다. 만약 이 break가 없으면 do-while문의 조건식이 항상 참이 되어 무한루프에 빠질 것이라 생각했기 때문이다.

3. Implementation

가장 먼저 영어와 수학의 학점 변수를 입력 받을 문자형 변수 두 개를 선언했다. 각각의 이름은 영어와 수학의 학점이라는 뜻에서 Eng_credit 과 Math_credit 으로 명명했다. 또, 입력 받은 학점 변수를 실수로 바꿔서 계산할 영어와 수학의 평점 변수 두 개 역시 선언했다. 이 때 소수점까지 계산해야 해서 실수형 변수를

이용했는데, 2자리까지만 출력하면 되므로 굳이 double을 쓰지 않고 float형 변수를 이용했다. 이들의 이름 역시 일관성을 위해 Eng_grade 와 Math_grade 로 두었다.

그리고 문제 해결을 할 때 생각한 것처럼 적어도 한번의 학점을 입력 받기 위해 do-while문으로 시작했다.

do문의 중괄호 안에 먼저 사용자에게 학점 입력을 요구할 문구를 띄우고 학점을 입력 받기로 했다. 그래서 다음과 같이 프로그램 코드를 짰다.

```
printf("영어와 수학의 학점을 순서대로 대문자로 입력하십시오\n");
```

```
scanf(" %c%c", &Eng_credit, &Math_credit);
```

영어와 수학의 학점의 순서가 바뀌면 안되므로 '순서대로'라는 문구를 출력했고, c언어는 대문자와 소문자를 구별하므로 '대문자로'라는 문구 역시 출력했다. 또, scanf문장을 입력할 때, 민형복 교수님이 작성하신 practice 3의 내용 중 " %c"에 대한 이해를 제대로 하지 못해 처음에는 "%c%c"와 같이 작성했다. 그러나 이렇게 작성하면 do-while루프를 돌면서, 입력할 때 마다 쳤던 <Enter>키, 즉 \n가 "%c"에 대응해 변수에 저장된다는 사실을 깨닫게 되었다. 따라서, 민형복 교수님이 작성하신 practice 3의 내용 중에서 " %c"는 그 이전에 white space character등을 무시하고 입력을 받아들인다는 점을 참고해 " %c%c"로 변경했다. 이렇게 되면 두 문자를 입력 받고 그 후에 입력 받은 \n은 " %c"에 의해 무시될 것이고, 프로그램이 정상적으로 작동할 것이라 보았다.

문제에서 학점 변수에 입력된 값이 영어와 수학 중 하나라도 E이면 프로그램을 끝내라고 했으므로, 영어가 E 혹은 수학이 E일 때 break문을 이용해 탈출하기 위한 코드를 다음과 같이 짰다.

```
if ((Eng_credit == 'E') || (Math_credit == 'E'))  
  
    break;
```

그 다음에는 학점이 A, B, C, D, F인 경우를 생각하고 이에 따라서 다른 값을 평점 변수에 넣기로 했다. 어떤 값을 넣을 지는 주어져 있으므로, 그대로 따르면 됐다.

영어의 경우에는 실습 2에서 했던 것처럼 if-else문을 이용해 마치 다중 선택처럼 이용 한 후, A, B, C, D, F인 경우 학점에 따라 평점 변수에 값을 대입했다. 그러나 이 경우가 아닌 경우에는 출력하면 안 됐다. 따라서, 오류 메시지를 주고 다시 점수 입력을 요구해야 했다. 이 방법은 문제 해결법에서 구상한 대로 평점 변수에 음수를 대입했다. 그리고 나중에 이 평점 변수가 둘 다 양수인 경우만 출력하기로 했다. 즉, A, B, C, D, F가 아닌 경우 다음과 같이 프로그래밍했다.

.....(상략)

```
else                                     (즉, A, B, C, D, F가 아닌 경우)  
  
    printf("영어 학점의 오류입니다"), Eng_grade = -1.0;
```

수학의 경우에는 원리 자체는 영어에서 했던 것과 같이 했으나, 다만 switch문을 사용했다. case'A', case'B', case'C', case'D', case'F'인 경우로 나눠 이에 따라 평점

변수에 값을 대입했다. 역시 이 경우가 아닌 경우, 즉 default에서는 출력하면 안되므로 오류 메시지를 주고 다시 점수 입력을 요구해야 했다. 즉, A, B, C, D, F가 아닌 경우 영어에서 했던 것과 같이 다음과 같이 프로그래밍했다.

.....(상략)

```
default : printf("수학 학점의 오류입니다"), Math_grade = -1.0; break;
```

이제 평점 변수에 값들이 할당되었으므로, 평균을 출력하면 된다. 그러나 아까 언급한 듯이 둘 중에 하나라도 음수 값이 있으면 입력이 잘못된 경우, 즉 A, B, C, D, F가 아닌 경우이다. 이 때는 출력하지 말아야 한다. 그러려면 평점 변수가 둘 다 0이상일 때만 출력하면 되므로, if문을 이용하기로 했다. 그리고 단지 $(Eng_grade + Math_grade)/2$ 를 출력하면 됐다. 바로 이렇게 쓴 이유는 이들이 실수형 변수라 소수점 아래를 버리지 않는다는 사실을 알았기 때문이다. 그리고 소수점 2자리 까지 출력한다는 사실에 입각해 "%.2f"라고 작성했다. 이 내용을 종합해 다음과 같이 프로그래밍 했다.

```
if ((Eng_grade >= 0.0) && (Math_grade >= 0.0))  
    printf("%.2f\n", (Eng_grade + Math_grade)/2);
```

그리고 문제 해결을 구상할 때 했던 것처럼 while문을 검사하는 식에 1을 두기로 했다. E를 입력할 때까지 반복해서 실행해야 하기 때문이다. 그래서 E에 break문을 사용한 이유도 이 때문이다. 0의 값을 가지지 않는 새로운 변수를 사용해 이 변수가 E일 때만 0이 되게 해 두고, while식을 검사하는 곳에 그 변수를 두어 E를 입력할 때만 빠져 나갈 수도 있었겠지만, 이 방법보다는 내가

사용한 방법이 직관적으로 보기 쉽다고 생각했다.

4. Result

```
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
AC
3.00
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
DF
0.50
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
AB
3.50
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
ab
영어 학점의 오류입니다.
수학 학점의 오류입니다.
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
Ra
영어 학점의 오류입니다.
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
AG
수학 학점의 오류입니다.
영어와 수학의 학점을 순서대로 대문자로 입력하십시오
ED
계속하려면 아무 키나 누르십시오 . . .
```

여러 가지로 나누어서 입력해 보았다.

1) 두 개의 학점을 정상적으로 준 경우

문제에서 제시된 평점을 바탕으로 계산한 값과 같은 정상적인 평균값이 출력되었다.

2) 두 개의 학점을 소문자로 준 경우

예상과 같이 c언어는 소문자와 대문자를 구별하므로, 인식되지 않아 오류 메시지가 나오면서 재입력을 요구했다.

3) 영어 학점만 잘못 준 경우

‘영어 학점의 오류입니다’라는 문구를 출력했다.

4) 수학 학점만 잘못 준 경우

‘수학 학점의 오류입니다’라는 문구를 출력했다.

5) 둘 중 하나라도 E를 준 경우

문제에서 요구한 대로 do - while문장을 빠져 나와 프로그램을 종료시켰다.

이번에도 결과는 예상한 것과 같이 나왔다.

5. Conclusion & Evaluation

이번 실습은 저번에 배웠던 if문의 확장이라고 생각할 수 있겠다. 여러 문장(flow of control)들을 사용하는데, 처음에는 꽤 어려웠다. 하지만 이런 문장들을 몇 번 사용하다 보니 익숙해지고, 어느 정도 능숙하게 다룰 수 있었다. 이렇게 해서 주어진 문제를 어렵지 않게 풀 수 있었다.

딱 한번 헤맨 것이 " %c"에 대한 것이다. 처음에는 저것이 왜 필요한지 몰랐다. 어차피 두 번 입력하고 그 다음에 입력된 new line character는 더 이상 쓸모 없는 것 아닌가? 라고 생각했었다. 그러나 do-while문장을 통해 계속 반복하게 되므로, 입력된 new line character들이 변수 값에 들어갈 수도 있다는 사실을 깨닫게 되었고, 그 결과"%c%c"를 " %c%c"로 바꿀 수 있었다. 스스로 시행착오를 겪으며 오류를 수정해 나가면서, 논리적인 분석력과 프로그래밍 실력이 많이 상승한 것 같다.

또 switch문은 정수형 변수만 이용 가능한 줄 알았는데, 조교님께 여쭙어 본 결과 문자형 변수 역시 이용 가능하다는 것에서 놀랐다. 이런 과정을 통해 문자형 상수를 이용할 때에는 항상 작은 따옴표를 붙이는 것을 잊지 말아야 한다는 점도 각인 할 수 있었다.

다음 실습은 아마 함수에 대한 것으로 기억한다. 함수는 printf, scanf 밖에 모르는데, 어떤 것에 대해 배울 지 기대 되면서도 잘 하지 못할까 걱정스럽다. 그러나 이번 실습을 통해 익숙하지 않았던 여러 flow of control에 대해 익숙해진 것처럼, 다음 실습도 함수에 대해 익숙해 질 수 있기를 바란다.

6. 참고 문헌

- [1] 민형복, program template, p3.c.
- [2] 프로그래밍 기초와 실습 사이트, <http://class.icc.skku.ac.kr/~min/C/>, 보고서 작성
요령, 2014 년 10 월.
- [3] Al Kelly, Ira Pohl, *C by Dissection: The Essentials of C Programming, Fourth Edition*,
Pearson, p. 99. 102 ~ 104. 112.
- [4] 민형복, practice3.pdf.