

# REPORT

---

P



## [Pascal Program]

과목	프로그래밍언어론
담당교수	서영훈 교수님
학과	소프트웨어학과
학번	2021041057
이름	이은서
제출날짜	2024. 03. 30.

## Contents

1. 과제 개요	3
2. 알고리즘	3
3. 소스코드	4~6
4. 시행결과	6~7
5. Discussion	7~8

## 1. 과제개요

- ❖ 여러 학생의 과목별 점수를 입력하고, 이를 합산하여 정렬(sorting)하는 Pascal program을 작성하라.
  - 한 학생의 data는 학번, 이름, 과목별 성적(3개 이상)으로 하라.
  - Quick sorting algorithm을 이용하라.
  - 서브 프로그램을 두개 이상 정의하여 작성하라.
  - EOF, EOLN을 이용하여, 임의의 학생수 및 과목수에 대해 처리할 수 있도록 하라. (확장성이 좋도록 작성하라)

여기서 구체적으로 정해야 하는 것은 크게 두 가지를 생각해볼 수 있다.

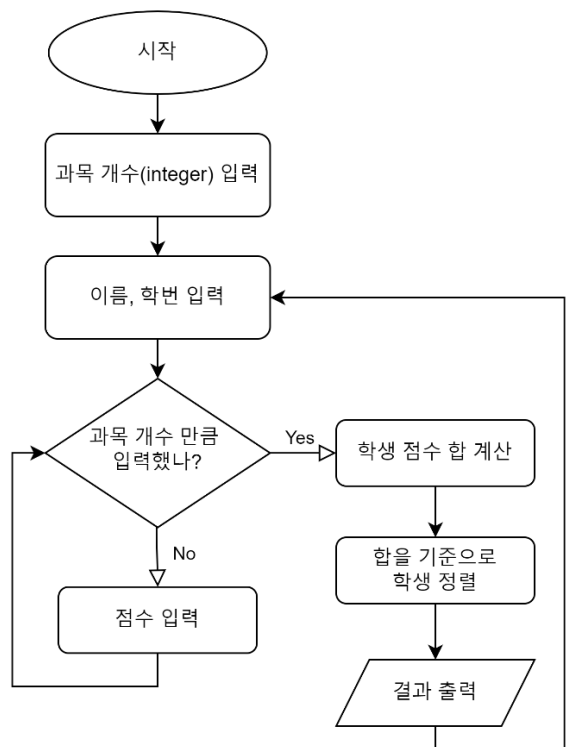
1. 여러 학생의 정보 입력 방법은 어떻게 할 것인가?
2. 확장성을 좋게 하기 위해서는 어떻게 할 것인가?

본 보고서에서는 입력할 과목의 개수를 입력받고 반복문을 이용해 학생의 정보와 성적을 입력받은 후, Quick Sort한 결과를 보여주는 것을 반복한다.

## 2. 알고리즘

1. 과목 개수 입력
2. 학번, 이름 입력
3. 반복문을 이용해 과목 개수만큼의 점수 입력
4. 점수의 합 계산
5. 총합을 기준으로 학생 점수 정렬
6. 결과를 출력
7. 프로그램을 종료할 때까지 2~6을 반복

학생은 배열로 학생의 정보는 구조체 배열을 이용했다.



### 3. 소스코드

```
program firstPascal(input, output);
type
    s_Record = record
        ID: Integer;
        Name: String;
        Grade: array [1..20] of Integer;
        GradeSum: Integer;
    end;

var
    subjects: Integer;
    n: Integer;
    start: Integer;
    students: array [1..100] of s_Record;
    tmp: s_Record;

procedure ASK(var subjects:Integer);
begin
    Write('Enter number of Subjects (Max 20) : ');
    ReadLn(subjects);
end;

procedure addStudent(var students:array of s_Record; var n:Integer; var
subjects:Integer);
var
    i: Integer;
begin
    Write('Enter Name : ');
    ReadLn(students[n+1].Name);

    Write('Enter ID : ');
    ReadLn(students[n+1].ID);

    for i:=1 to subjects do
    begin
        Write('Enter Grade ', i, '/', subjects, ' : ');
        ReadLn(students[n+1].Grade[i]);
        students[n+1].GradeSum += students[n+1].Grade[i];
    end;
    n := n + 1;
end;

procedure showResult(var n:Integer; var subjects:Integer; var students:array of
s_Record);
var
    stu, sub : Integer;
```

```

begin
    WriteLn();
    write('num':4,'Name':10, 'ID':10);
    for sub := 1 to subjects do
    begin
        write(sub:5);
    end;
    write('sum':7);

    WriteLn();
    write('----':4,'-----':10, '-----':10);
    for sub := 1 to subjects do
    begin
        write('-----':5);
    end;
    write('-----':8);
    WriteLn();

    for stu := 1 to n do
    begin
        Write(stu:4, students[stu].Name:10, students[stu].ID:10);
        for sub := 1 to subjects do
        begin
            write(students[stu].Grade[sub]:5);
        end;
        write(students[stu].GradeSum:7);
        WriteLn();
    end;
    WriteLn();
end;

procedure quickSort(var students:array of s_Record; var n:Integer; var
subjects:Integer; var left:Integer; var right:Integer; var tmp:s_Record);
var
    pivot, l, r: Integer;
begin
    l := left;
    r := right;
    pivot := students[(left + right) div 2].GradeSum;

    while l<=r do begin
        while students[l].GradeSum < pivot do l+=1;
        while students[r].GradeSum > pivot do r-=1;
        if l <= r then begin
            if l <> r then begin
                tmp := students[l];
                students[l] := students[r];
                students[r] := tmp;
            end;
        end;
    end;
end;

```

```

        end;
        l+=1; r-=1;
    end;
end;

    if left < r then quickSort(students, n, subjects, left, r, tmp);
    if l < right then quickSort(students, n, subjects, l, right, tmp);
end;

begin
    ASK(subjects);
    n := 0;
    start := 1;
    while true do begin
        addStudent(students, n, subjects);
        quickSort(students, n, subjects, start, n, tmp);
        showResult(n, subjects, students);
    end;
end.

```

## 4. 실행결과

```

Free Pascal Compiler version 3.2.2+dfsg-9ubuntu1 [2022/04/11]
Copyright (c) 1993-2021 by Florian Klaempfl and others
Target OS: Linux for x86-64
Compiling main.pas
Linking a.out
108 lines compiled, 0.1 sec
Enter number of Subjects (Max 20) : 5
Enter Name : eunseo
Enter ID : 57
Enter Grade 1/5 : 85
Enter Grade 2/5 : 99
Enter Grade 3/5 : 90
Enter Grade 4/5 : 80
Enter Grade 5/5 : 95

num      Name      ID      1      2      3      4      5      sum
-----
1      eunseo      57      85      99      90      80      95      449

Enter Name : min
Enter ID : 85
Enter Grade 1/5 : 70
Enter Grade 2/5 : 99
Enter Grade 3/5 : 98
Enter Grade 4/5 : 80
Enter Grade 5/5 : 92

num      Name      ID      1      2      3      4      5      sum
-----
1          min      85      70      99      98      80      92      439
2      eunseo      57      85      99      90      80      95      449

```

```

Enter Name : doby
Enter ID : 50
Enter Grade 1/5 : 20
Enter Grade 2/5 : 90
Enter Grade 3/5 : 66
Enter Grade 4/5 : 88
Enter Grade 5/5 : 40

```

num	Name	ID	1	2	3	4	5	sum
1	doby	50	20	90	66	88	40	304
2	min	85	70	99	98	80	92	439
3	eunseo	57	85	99	90	80	95	449

```

Enter Name : sanrio
Enter ID : 30
Enter Grade 1/5 : 60
Enter Grade 2/5 : 88
Enter Grade 3/5 : 40
Enter Grade 4/5 : 55
Enter Grade 5/5 : 90

```

num	Name	ID	1	2	3	4	5	sum
1	doby	50	20	90	66	88	40	304
2	sanrio	30	60	88	40	55	90	333
3	min	85	70	99	98	80	92	439
4	eunseo	57	85	99	90	80	95	449

```

Enter Name : ponyo
Enter ID : 5
Enter Grade 1/5 : 40
Enter Grade 2/5 : 69
Enter Grade 3/5 : 55
Enter Grade 4/5 : 60
Enter Grade 5/5 : 87

```

num	Name	ID	1	2	3	4	5	sum
1	doby	50	20	90	66	88	40	304
2	ponyo	5	40	69	55	60	87	311
3	sanrio	30	60	88	40	55	90	333
4	min	85	70	99	98	80	92	439
5	eunseo	57	85	99	90	80	95	449

```

Enter Name : ^Z
Enter ID : Killed

```

```

...Program finished with exit code 9
Press ENTER to exit console.

```

## 5. Discussion

우선 프로그램에 대한 input을 받는 방식과 순서를 정하는 것부터 고민을 했었다. 하지만 이를 정하고 난 이후에 프로그래밍을 하면서, Pascal이 C와 닮은 듯 다른 면들이 체감되었다. 그래서 알고리즘을 짜는 것 자체가 어렵진 않았지만, begin과 end를 사용하는 것, ";"를 쓸 때와 "."를 쓸 때를 구문하

는 등 사용하는 키워드나 문법에 대한 약간의 어려움이 있었다.