

Programming Languages and Compilers

Lectured by Prof. Chung Yung

# **Programming Assignment 2**

410821312 張宸瑋 資工三

410921217 鄭宇婕 資工二

## 1. Problem description

Practice on programming the same exercises in 5 different programming languages with Java, Python, R, ML, and Prolog.

The grade of a student is translated according to the following table:

Score	Grade
0-49	E
50-59	D
60-62	C-
63-66	C
67-69	C+
70-72	B-
73-76	B
77-79	B+
80-84	A-
85-89	A
90-100	A+

The overall score of a student is calculated by the following formula:

$$\text{Score} = \text{HW1} * 0.1 + \text{HW2} * 0.1 + \text{HW3} * 0.1 + \text{Midterm} * 0.3 + \text{Final} * 0.4$$

Write a program in the above 5 different programming languages to translate the overall score of each student into a grade.

## 2. Highlight the way you write the program

- Python
  - We first read the CSV file to our program and then store it in dataframe.
  - Operate the data frame to get each score (HW, midterm and final) of all students
  - Calculate the average score based on the given formula and add it to a list
  - Translated the calculated score to the corresponding grade and add it to another list
  - Add the score and grade list to the end of the original dataframe
  - Print out the dataframe

- R
  - We first read the CSV file to our program and store it in a 2D-array
  - Calculate the average score based on the given formula and add it to an array named score
  - Go through the score array and find out the corresponding grade, then add it to another array named grade
  - Add the original 2D-array, score array and grade array to dataframe
  - Print out the data frame
  
- Java
  - Use File to read the csv file
  - Create a 2D-list to store the csv file
  - Declare an arraylist named score to store the calculated score based on the given formula
  - Declare another arraylist named grade to store the corresponding grade according to the score
  - Print out the 2D-list (original data include name, id, HW, mid and final), score arraylist and grade arraylist
  
- ML
  - Declare the name, id, hw1-3, midterm and final list in the program
  - For list hw1 ~ hw3, \* 0.1, for the midterm list, \*0.3 and final list \*0.4
  - Add the above list together so we can get the score for each student
  - Using a function to go through the score list and store the corresponding grade to each independent variable
  - Store each student's name, id, score in different variable
  - Print out those variable(adjust it to print out like a dataframe)
  
- Prolog
  - Declare arrays for each students, which contains NO, id, name and score for HW1-3, midterm and final.
  - For each array, calculate the score based on the given formula and store it in an independent variable
  - Write a function for printing out.
  - While printing out the original data, convert each student's score to the corresponding grade and then also print it out.

### 3. Program Listing

Link to the code file on HackMD

- [Program listing for Assignment 2](#)

### 4. Test run results

Python

```
(base) yujie@zhengyujiedeMacBook-Pro programing assignment_2 % python -u "/Users/yujie/Desktop/programing assignment_2/p.py"
NO      ID      Name  HW1  HW2  HW3  Midterm  Final  Score  Grade
0  1  410021001  Alan   90  84.5  117     60    66   73.5    B
1  2  410021002   Bob   85  49.0   80     57    64   64.1    C
2  3  410021003 Carrie   90  110.5  117     68    62   76.9   B+
3  4  410021004 David  117  85.0    0     44    55   55.4    D
4  5  410021005 Ethan   85  56.0   50     57    67   63.0    C
5  6  410021006 Frank   90  65.0   65     72    66   70.0   B-
6  7  410021007 Gary   117  110.5  65     69    43   67.1   C+
7  8  410021008 Helen  117  65.0   50     43    54   57.7    D
8  9  410021009 Igor    63  59.5   50     51    75   62.5    C
9 10  410021010 Jeff   117  110.5  117     53    75   80.3   A-
```

R

```
> source("/Users/yujie/Desktop/programing assignment_2/HW2.R")
NO      ID      Name HW1  HW2 HW3 Midterm Final score grade
1  1  410021001  Alan   90  84.5  117     60    66 73.55    B
2  2  410021002   Bob   85  49.0   80     57    64 64.10    C
3  3  410021003 Carrie   90  110.5  117     68    62 76.95   B+
4  4  410021004 David  117  85.0    0     44    55 55.40    D
5  5  410021005 Ethan   85  56.0   50     57    67 63.00    C
6  6  410021006 Frank   90  65.0   65     72    66 70.00   B-
7  7  410021007 Gary   117  110.5  65     69    43 67.15   C+
8  8  410021008 Helen  117  65.0   50     43    54 57.70    D
9  9  410021009 Igor    63  59.5   50     51    75 62.55    C
10 10  410021010 Jeff   117  110.5  117     53    75 80.35   A-
```

Java

```
(base) yujie@zhengyujiedeMacBook-Pro programing assignment_2 % cd "/Users/yujie/Desktop/programing assignment_2/" && javac HW2.java
&& java HW2
NO      ID      Name  HW1  HW2  HW3  Midterm  Final  Score  Grade
1  410021001  Alan   90  84.5  117     60    66   73.55    B
2  410021002   Bob   85    49    80     57    64   64.10    C
3  410021003 Carrie   90  110.5  117     68    62   76.95   B+
4  410021004 David  117    85    0     44    55   55.40    D
5  410021005 Ethan   85    56    50     57    67   63.00    C
6  410021006 Frank   90    65    65     72    66   70.00   B-
7  410021007 Gary   117  110.5  65     69    43   67.15   C+
8  410021008 Helen  117    65    50     43    54   57.70    D
9  410021009 Igor    63  59.5   50     51    75   62.55    C
10 410021010 Jeff   117  110.5  117     53    75   80.35   A-
```

## ML

```
Standard ML of New Jersey
54  ^ Real.toString std8 ^ " " ^ grade8 ^ "\n"
= Int.toString no9 ^ " " ^ id9 ^ " " ^ stdname9 ^ " " ^ " 63 59.5 50 51
75  ^ Real.toString std9 ^ " " ^ grade9 ^ "\n"
= Int.toString no10 ^ " " ^ id10 ^ " " ^ stdname10 ^ " " ^ " 117 110.5 117 5
3 75  ^ Real.toString std10 ^ " " ^ grade10 ^ "\n");

NO      ID      NAME      HW1      HW2      HW3 Midterm  Final  Score  Grade
1 410021001 Alan      90      84.5    117      60      66      73.55  B
2 410021002 Bob       85       49      80       57      64      64.1   C
3 410021003 Carrie    90     110.5   117      68      62      76.95  B+
4 410021004 David    117      85       0      44      55      55.4   D
5 410021005 Ethan    85       56      50       57      67      63.0   C
6 410021006 Frank    90       65      65       72      66      70.0   B-
7 410021007 Gary     117    110.5   65       69      43      67.15  C+
8 410021008 Helen    117      65      50       43      54      57.7   D
9 410021009 Igor     63     59.5    50       51      75      62.55  C
10 410021010 Jeff    117    110.5   117      53      75      80.35  A-
```

## Prolog

```
Last login: Tue Jun  7 20:39:34 on ttys002
[(base) yujie@zhengyujiedeMacBook-Pro ~ % gprolog
GNU Prolog 1.5.0 (64 bits)
Compiled Jul  9 2021, 00:33:49 with clang
Copyright (C) 1999-2021 Daniel Diaz

| ?- change_directory('/Users/yujie/Desktop/programing assignment_2').

(1 ms) yes
| ?- ['HW2'].

compiling /Users/yujie/Desktop/programing assignment_2/HW2.pl for byte code...
/Users/yujie/Desktop/programing assignment_2/HW2.pl compiled, 76 lines read -
42553 bytes written, 12 ms

NO      ID      NAME      HW1      HW2      HW3 Midterm  Final  Score  Grade
1 410021001 Alan      90      84.5    117      60      66      73.55  B
2 410021002 Bob       85       49      80       57      64      64.10  C
3 410021003 Carrie    90     110.5   117      68      62      76.95  B+
4 410021004 David    117      85       0      44      55      55.40  D
5 410021005 Ethan    85       56      50       57      67      63.00  C
6 410021006 Frank    90       65      65       72      66      70.00  B-
7 410021007 Gary     117    110.5   65       69      43      67.15  C+
8 410021008 Helen    117      65      50       43      54      57.70  D
9 410021009 Igor     63     59.5    50       51      75      62.55  C
10 410021010 Jeff    117    110.5   117      53      75      80.35  A-

(5 ms) yes
```

## **5. Discussion**

For Python, R and Java, we didn't encounter many problems. It was easier to write compared to ML and prolog. For ML, it took us a lot of time to figure out how to do "for" loops like other languages. ML is a language for functional programming language, it is more popular among compiler writers and programming language researchers. So using ML to operate data in CSV file isn't that easy. As for prolog, it's more powerful when associated with AI and computational linguistics. So using it to operate data in CSV is also inappropriate. But after lots of studies and discussion, we finally figured out how to write these two languages. But unfortunately, we are not able to read the CSV file and operate it. But we believed with more given time, we may find out some way to operate the CSV file.