

FAED: Financial Aid Eligibility Detector

Language

Final Report



ICOM 4036: Prof. Wilson Rivera

Isabel Andrea Muñiz

Ralph Ulysse

Erick Rodríguez

Daniel Crespo

Introduction

FAED stands for financial aid eligibility detector. This language at his core is created with hopes of making the process of verifying a student's eligibility for financial aid as simple as possible. We want all the calculations, all the decisions, all the implications to be handled by the simples of command, a macro. The system is flexible, meaning it can be modified and customized to handle different schools criteria; it can be expanded adding new requirements, and options. The design is maid to be used by anyone, no knowledge in computing nor mathematics is required; all you'll need is the student's name and his grades. From that you can learn his GPA, his status, meaning if he's eligible or not, and if he is how much he's getting and you also have the power if the student fails short to still consider his case if he satisfies other requirements. We understand this kind of work often take days, sometime even week, but this is where our language comes in. We will be using PLY to implement our language, the python parser and scanner tool that will allow us to make the language as easy straight forward as possible.

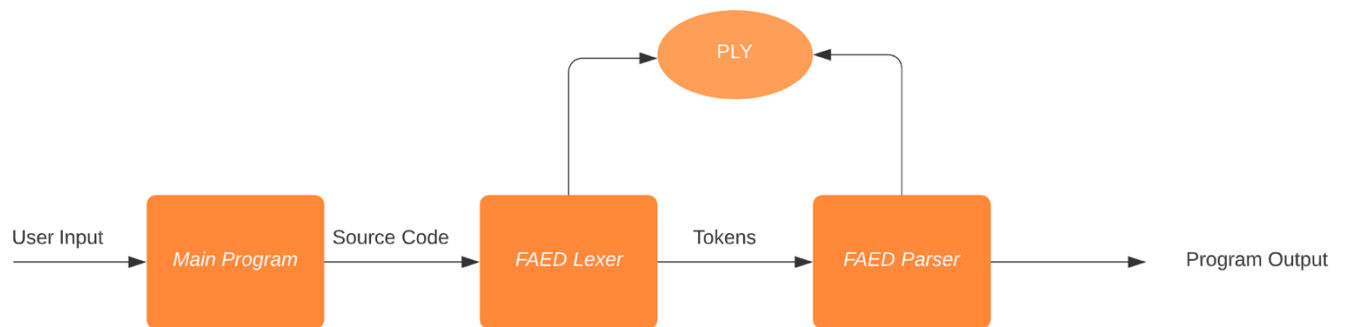
Language Tutorial

Steps of how to use our programming language:

1. Download FAED from our GitHub repository usinf the following link:
<https://github.com/4015a13/MacroProject>
2. Make sure to have Python 3 (3.6.5) installed in your computer.
3. Install the following python libraries: PLY.
4. Run the macro.py file on your favorite IDE or terminal/command prompt.
5. Start using FAED Language!

Language Development

Translator Architecture



Macro Grammar

<statement> → <method> | <assignment>

<assignment> → <stu_assignment> | <method_assignment>

<method> → <method_moveup> | <method_print>

<method_print> → ID DOT PRINT

<method_moveup> → ID DOT METHOD_MoveUp

<stu_assignment> → STUDENT EQUALS ID COMMA ID COMMA ID COMMA ID COMMA ID

<method_assignment> → ID DOT METHOD_INQ

Macro commands:

Register: when used, the statement “student = X,A,B,C,D” effectively registers a student into the system. The student X is added into a list of all the students and his grades A,B,C,D are saved, computed in order to determine his gpa.

GPA: the command X.gpa yield the GPA of the student X but does nothing else.

Status: the command X.status informs you of the student GPA and if he's eligible or not for aid and also informs you of the amount of the aid, if it's full, half, a quarter or nothing at all.

Evaluate: the command X.evaluate set in motion the accreditation of the student by adding him to the corresponding list. The list fullQualified represents the set of student with a GPA of 3.5 or higher, halfQualified the ones between 3 to 3.5, quarterQualified the ones between 2.75 to 3 and notQualified is the list where all the students that have a GPA below 2.75 are added.

Force: the command X.force is designed for special cases. A student not qualified can be bumped into the group of the students qualified for a quarter of the aid.

Print: the print command is designed for the list. If at any time one wish to know who are the list's occupants, all you have to do is write down the words full, half or quarter followed by .print for the students qualified and register.print for all the students registered and none.print for the students who are not qualified. In theory all the students from the lists full, half, quarter and none added together are the ones in the list register.

Expel: this command get rid of a student. If a student gets expelled he will no longer be in any of the list.

Macro FAED legend

Let's say we have a case of a student named Joseph:

name: Joseph

grade 1: A

grade 2: C

grade 3: B

grade 4: A

Using our language, to register the student in the system, you do the following:

```
C:\Users\User\PycharmProjects\lexel\venv\Scripts\python.exe C:/Users/User/PycharmProjects/lexel/Macro.py
Macro >>student = Joseph,A,B,C,A
Macro >>
```

Let's say you wish to know Joseph's GPA only:

```
Macro >>Joseph.gpa
Joseph 's GPA is: 3.425
GPA 001
Macro >>
```

If you know his GPA, perhaps you'd want to know where he stands, how much aid he's entitled to:

```
Macro >>Joseph.status
With a GPA of 3.425 the student Joseph is qualified for half of the financial aid
STATUS 002
Macro >>
```

If you then want Joseph to be added into the system and actually receive the aid:

```
Macro >>Joseph.evaluate
Student should receive the funds shortly
EVALUATE 002
Macro >>
```

It should be noted that at this point the student has been added to the list were all the students receiving the same amount as him are already and if one should want to see who is in that list:

```
Macro >>half.print
Joseph
Marco
Ralph
Lucia
Raymond
```

There's also a case where a student just fell short of getting in and the person in charge wish to reconsider the case. Providing the student's GPA is at a specified minimum, it might still be possible:

```
Macro >>student = Felix,C,C,B,C  
Macro >>Felix.evaluate  
Student will not receive the founds  
EVALUATE 004  
Macro >>Felix.force  
Operation Successful (FORCE)  
Student should receive the founds shortly (FORCED)  
FORCE 001
```

You can also decide to revoke a student's aid due to other reason:

```
Macro >>Felix.expel  
Student expelled  
EXPEL 001
```

These are some of the cases currently in our language, of course there are a few cases not presented here, like trying to “force” a student in when he’s not qualified, the system will not allow it, trying to expel someone who hasn’t been in the system etc.