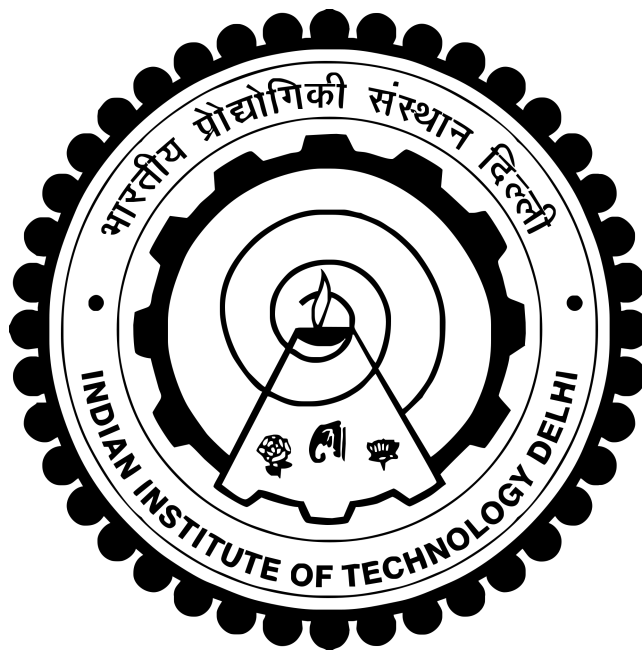


ELP718 Telecom Software Laboratory
1st Semester, 2016-18
Abhishek Mishra
08 Nov 2016, 5pm
Assignment-13



Contents

- 0.1 Introduction 3
- 0.2 Problem Statement 1 4
 - 0.2.1 Assumptions 4
- 0.3 Problem Statement 2 4
 - 0.3.1 Assumptions 4
- 0.4 Structure Chart 4
- 0.5 Screenshots 5
- 0.6 Epilogue 8

0.1 Introduction

This assignment aims to provide a better understanding of the following topics:

1. **Socket Programming**

Sockets allow communication between two different processes on the same or different machines. To be more precise, it's a way to talk to other computers using standard Unix file descriptors. In Unix, every I/O action is done by writing or reading a file descriptor. A file descriptor is just an integer associated with an open file and it can be a network connection, a text file, a terminal, or something else.

To a programmer, a socket looks and behaves much like a low-level file descriptor. This is because commands such as `read()` and `write()` work with sockets in the same way they do with files and pipes.

0.2 Problem Statement 1

Write a lex/lex-yacc code to evaluate the consistency of given .pgn file, which record chess moves in given game. You have to find any wrong notations and its location in metadata or using move number

0.2.1 Assumptions

The files that are read follow strict pgn rules. Design Database for given system using MySQL i.e. create one database and the relational tables described above. Also write a python code to populate the tables. The generated table looks like this:

0.3 Problem Statement 2

You are given with list of students in given format with their marks in respective subject. You have to calculate their SGPA and write that down after every student. Use Lex and Yacc to for the purpose, such that it will check consistency of input format and calculate CGPA.

0.3.1 Assumptions

The files that are read follow strict pgn rules. Design Database for given system using MySQL i.e. create one database and the relational tables described above. Also write a python code to populate the tables. The generated table looks like this:

0.4 Structure Chart

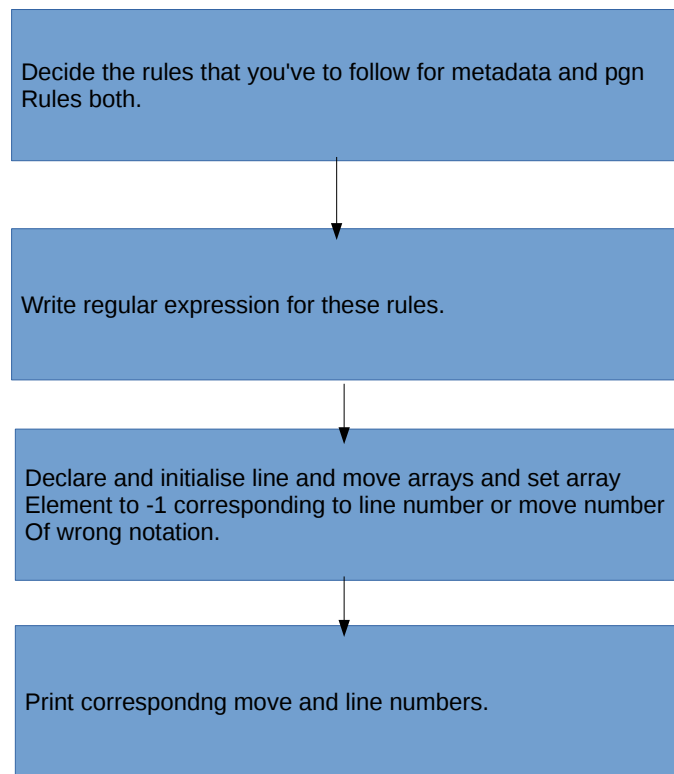


Figure 1: Structure chart for Problem 1

0.5 Screenshots

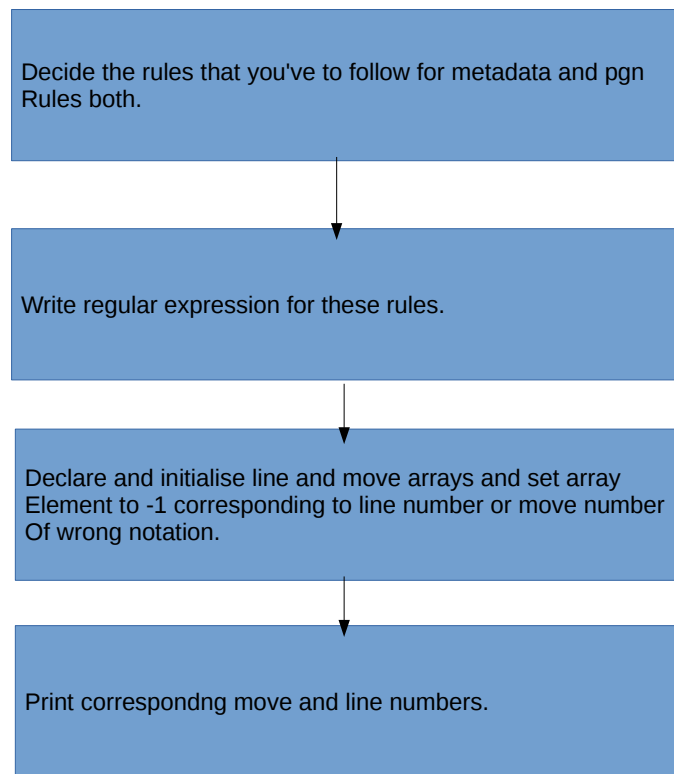
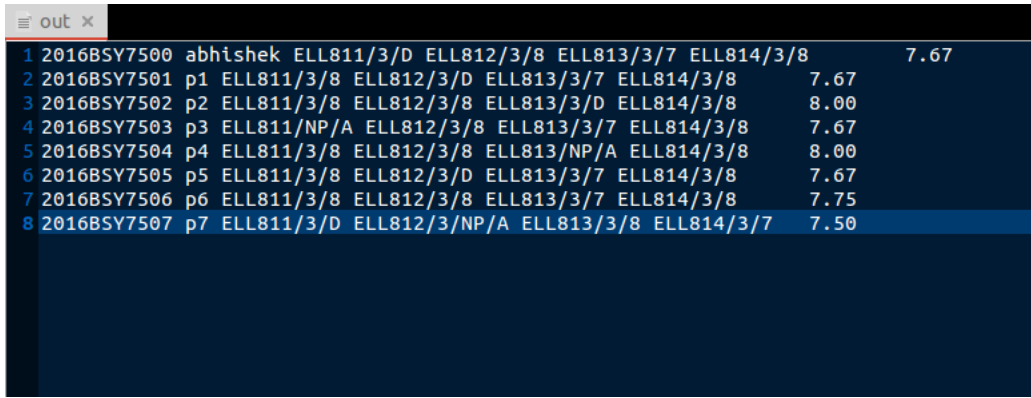


Figure 2: Structure chart for Problem 2

```
abhishek@admin:~/Desktop/Uploads/bsy17508_12$ ./run1.sh  
  
Error at line 1  
Error in move 3  
Error in move 18
```

Figure 3: Screenshot for problem 1



1	2016BSY7500	abhishek	ELL811/3/D	ELL812/3/8	ELL813/3/7	ELL814/3/8	7.67
2	2016BSY7501	p1	ELL811/3/8	ELL812/3/D	ELL813/3/7	ELL814/3/8	7.67
3	2016BSY7502	p2	ELL811/3/8	ELL812/3/8	ELL813/3/D	ELL814/3/8	8.00
4	2016BSY7503	p3	ELL811/NP/A	ELL812/3/8	ELL813/3/7	ELL814/3/8	7.67
5	2016BSY7504	p4	ELL811/3/8	ELL812/3/8	ELL813/NP/A	ELL814/3/8	8.00
6	2016BSY7505	p5	ELL811/3/8	ELL812/3/D	ELL813/3/7	ELL814/3/8	7.67
7	2016BSY7506	p6	ELL811/3/8	ELL812/3/8	ELL813/3/7	ELL814/3/8	7.75
8	2016BSY7507	p7	ELL811/3/D	ELL812/3/NP/A	ELL813/3/8	ELL814/3/7	7.50

Figure 4: Screenshot for problem 2

0.6 Epilogue

This week's assignment tested our scanning and parsing skills using lex and yacc implemented in linux using flex and bison. The tokens are created with the help of lex and grammar is analysed using yacc.

Bibliography

- [1] “Sockets tutorial.” http://www.linuxhowtos.org/C_C++/socket.htm.