Project Title

An interpreter for a simple machine language SML in an .sml file. The general form of a machine language instruction is label instruction register-list
where label is the label for the line. Other instructions might “jump" to that label.
Instruction is the actual instruction.
In SML, there are instructions for adding, multiplying and so on, for storing and retrieving integers, and for conditionally branching to other labels (like an if statement).
Register-list is the list of registers that the instruction manipulates.
Registers are simple, integer, storage areas in computer memory, much like variables. In SML, there are 32 registers, numbered 0, 1, . . . , 31.

Instructions of a program are executed in order (starting with the first one), unless the order is changed by execution of a bnz instruction. Execution terminates when its last instruction has been executed (and doesn't change the order of execution).

Installing

Copy the classes and instantiate through the standard SBT file structure.

Running the tests

Tests can be run in IntelliJ or through SBT

Built With

* [IntelliJ]

Authors

* **Nathan Hanak**
** Initial source code provided by SDP class at Birkbeck