
BASIC YEAH

Hours

Kevin Miller
(kmiller4@stanford.edu)

How to BASIC

I'm going to walk through the steps of BASIC
and some tips

How to BASIC

1. Modify Program.h
 2. Implement Program.cpp
 3. TEST steps one and 2
 4. statement.h and statement.cpp
 5. Parsing “easy” Statements
 6. Parsing “hard” Statements
-

Program Interface

You need to choose a data structure (or two... hint hint) to do the following:

- `getFirstLineNumber`,
`getNextLineNumber`, and
`getParsedStatement` must run in $O(1)$
 - All other methods can be $O(n)$
-

Program.cpp

Once you chose your data structure(s), implementing Program.cpp becomes an exercise in using them

Testing Program

The next step is testing what you implemented. Doing this requires modifying Basic.cpp to do the following:

- typing in a line starting with a number should store the entire line (including the number) by calling `addSourceLine`
-

Testing Program

The next step is testing what you implemented. Doing this requires modifying Basic.cpp to do the following:

- typing in a line that contains **only** a line number should delete that line by calling `removeSourceLine`.
-

Testing Program

The next step is testing what you implemented. Doing this requires modifying Basic.cpp to do the following:

- Typing LIST, the interpreter should list the text for every line in the program in order
 - Typing CLEAR should discard all the lines in the program, freeing any associated heap memory
-

How to BASIC

1. ~~Modify Program.h~~
 2. ~~Implement Program.cpp~~
 3. ~~TEST steps one and 2~~
 4. statement.h and statement.cpp
 5. Parsing “easy” Statements
 6. Parsing “hard” Statements
-

Making a Statement

Your next step is making statement.h and statement.cpp. Read through exp.h and exp.cpp since it is **very similar**

Each statement should have a

- Constructor
 - Execute Method
 - Destructor
-

Parsing Statements

- We give you the code for parsing a Print Statement (see page 14 of the handout)
 - 3 of the others are almost identical (LET, REM, INPUT)
-

Parsing Statements

- To implement GOTO, IF, and END, you need to modify the current line number
- Easy way: Modify evaluationcontext to have a `getCurrentLineNumber` and `setCurrentLineNumber`

General Tips

- Start Early, there is a lot of code to write
 - Don't be discouraged by the apparent size of this assignment, it is not nearly as hard as cutting stock or even Boggle
 - The book is your friend
-