

BASIC Contest

Submission deadline: Friday, March 13, 5:00 P.M.

[Paul] Allen rushed to the dorm to find Bill Gates. They had to do a BASIC for this machine. They had to. If they didn't, the revolution would start without them.

—Stephen Manes and Paul Andrews, *Gates*, 1994

In Assignment #6, your goal is to build a minimal BASIC interpreter—the simplest language system you could write that is nonetheless able to do some real computing. For the BASIC contest, your challenge is to extend your Assignment #6 submission to be the best BASIC interpreter you can build—which is exactly the goal that Paul Allen and Bill Gates had in 1975 when they built a BASIC interpreter that would become the first product for a company that would later be called Microsoft. Your job is to complete the requirements of the assignment and then add as many extensions as you can, which can either be the ones listed here or any you can think of on your own. The most impressive interpreter wins.

Possible extensions

The following extensions would make your BASIC interpreter much more exciting as a programming language.

- In most implementations of BASIC, the `LET` keyword is optional. Incorporate this feature into your interpreter so that a statement like

```
total = n1 + n2
```

is interpreted as an assignment statement.

- Make expressions work with floating-point numbers instead of integers, which is what real BASIC does. This extension will require you to change the `exp.h` interface so that the value stored in a `ConstantExp` is a `double` rather than an `int`.
- Change the parser to allow the minus sign to be used as a unary operator as well as a binary operator.
- Add simple library functions like `SIN`, `COS`, and `SQR` (square root).
- Extend the `PRINT` statement so that it takes a list of expressions separated by commas. If you include this extension, you might also want to allow the list to contain quoted strings so that output messages can be displayed as in

```
PRINT "The answer is ", total
```

Fortunately for you, the scanner module already reads quoted strings, which simplifies this extension a bit.

- Implement string variables so that you can work with alphabetic as well as numeric data. In BASIC, the names of string variables end with a `$` so you can tell them apart from numeric ones. Strings are particularly useful if you also implement library functions to manipulate them.
- Try to implement other statement forms from BASIC, such as the `FOR/NEXT` statement used for looping, the `GOSUB/RETURN` statements used to simulate procedure calls, or the `DIM` statement used to introduce an array.
- Make it possible to list only a part of the program by allowing `LIST` to take a numeric range, as in `LIST 10-50`.
- Add the command `OLD`, which loads an existing BASIC program from a file, and `SAVE`, which writes out the current program.
- Add graphical capabilities that allow BASIC programs to draw lines in a graphics window. You might want to look at books on BASIC for suggestions as to how you might go about designing the statements to provide this capability.

Prizes

As you are all now aware, winning the grand prize in a contest entitles you to substitute a 100% for whatever individual score most negatively affects your grade at the end of the term. Thus, if you win this contest and end up bombing an assignment, the midterm, or even the final, we will overlook that misstep and count it as a 100%. We will probably also award various runner-up prizes and designate some entries as worthy of honorable mention, all of which increases your chances for winning the random contest drawing at the review session for the final.

Contest rules

1. Only students registered in CS 106B are eligible to submit entries.
2. Only one entry per person will be accepted.
3. All entries must be submitted electronically by 5:00P.M. on Friday, March 13, using the procedure for submitting any other assignment. Late entries will not be accepted.
4. Each submission must consist of a BASIC interpreter written in C++, along with any BASIC program files that you want us to consider. The `HELP` command in your interpreter must list any extensions that the judges should take into account. We have less than two days to evaluate these entries; if your `HELP` text doesn't tell us what extensions you've added, we'll assume that you haven't made any.
5. The contest entries will be judged by Eric Roberts and Kevin Miller, who will award the final prize to the most impressive BASIC interpreter. Decisions of the judges are final. Winners will be announced at the review session on Sunday, March 15.