Build Your Own Lisp

Eric Bailey

May 10, 2018 [1]

Write an abstract

Contents

Prompt

1a ⟨*Print version and exit information.* 1a⟩≡

```
puts("Lispy v0.0.1");
puts("Press ctrl-c to exit\n");
```

This code is used in chunk 2c.

Declare a buffer for user input of size 2048.

1b ⟨*Declare a buffer for user input.* 1b⟩≡

```
#define INPUT_SIZE 2048

static char input[INPUT_SIZE];
```

Defines:
  input, used in chunk 2.
This code is used in chunk 1c.

1c ⟨*prompt.c* 1c⟩≡

⟨*Include the standard I/O functions.* 2d⟩

```
#include <editline/readline.h>
```

⟨*Declare a buffer for user input.* 1b⟩

This definition is continued in chunk 2.
Root chunk (not used in this document).

Given a `prompt` string, `read` user input.

2a   ⟨*prompt.c* 1c⟩+≡

```c
char *read(const char *prompt)
{
  fputs(prompt, stdout);
  return fgets(input, INPUT_SIZE, stdin);
}
```

Defines:
    read, used in chunk 2.
Uses input 1b.

2b   ⟨*prompt.c* 1c⟩+≡

```c
void eval()
{
  printf("< %s", input);
}
```

Uses input 1b.

2c   ⟨*prompt.c* 1c⟩+≡

```c
int main(int argc, char *argv[])
{
  ⟨Print version and exit information. 1a⟩

  while (read("> ") ≠ NULL) {
    eval();
  }


  return 0;
}
```

Uses read 2a.

Common Headers

2d   ⟨*Include the standard I/O functions.* 2d⟩≡

```c
#include <stdio.h>
```

This code is used in chunk 1c.

Chunks

⟨*Declare a buffer for user input.* 1b⟩   1b, 1c
⟨*Include the standard I/O functions.* 2d⟩   1c, 2d
⟨*Print version and exit information.* 1a⟩   1a, 2c
⟨*prompt.c* 1c⟩   1c, 2a, 2b, 2c

Index

input: 1b, 2a, 2b
read: 2a, 2a, 2c

Todo list