

1. The details will be filled in due course. The interface of this module is included first. It is also used by the main programs.

2. Ritchie writes in "The Development of the C Language", see `c_development.html`: "... but the most important was the introduction of the preprocessor ... The preprocessor performs macro substitution, using conventions distinct from the rest of the language. ..."

Here's what Wittgenstein says in the **TLP**, see `t1p.pdf`: "6.24 The method by which mathematics arrives at its equations is the method of substitution. For equations express the substitutability of two expressions, and we proceed from a number of equations to new equations, replacing expressions by others in accordance with the equations."

Stallman defines *header file* in "The C Preprocessor", see `cpp.pdf`: A *header file* is a file containing C declarations and macro definitions (see Chapter 3 [Macros], page 13) to be shared between several source files. You request the use of a header file in your program by including it, with the C preprocessing directive **#include**.

3. Include the *printf* declaration.

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

4. Declaration of the global variables or function simply declares that the variable or function exists, but the memory is not allocated for them.

argc: copy of *ac* parameter to *main*.

argv: copy of *av* parameter to *main*

```
extern int argc;
```

```
extern char **argv;
```

5. Coming to the definition of the global variables, when we define a variable or function, in addition to everything that a declaration does, it also allocates memory for that variable or function.

```
int argc;
```

```
char **argv;
```

6. Index.

ac: [4](#).

argc: [4](#), [5](#).

argv: [4](#), [5](#).

av: [4](#).

main: [4](#).

LEXTERN

	Section	Page
Index	6	2