Students:
Section 5.15 is a part of 2 assignments: CSC108 CH05.8-5.16 C5B

Please browse to this assignment through BlackboardLearn so zyBooks knows where to send your activity. Learn more

Includes: CA
Due: 04/15/2025, 11:59 PM EDT

# 5.15 Preprocessor and include

The ***preprocessor*** is a tool that scans the file from top to bottom looking for any lines that begin with #, known as a ***hash symbol***. Each such line is not a program statement, but rather directs the preprocessor to modify the file in some way before compilation continues, each such line being known as a ***preprocessor directive***. The directive ends at the end of the line, no semicolon is used at the end of the line.

Perhaps the most commonly-used preprocessor directive is ***#include***, known as an ***include directive***. #include directs the compiler to replace that line by the contents of the given filename.

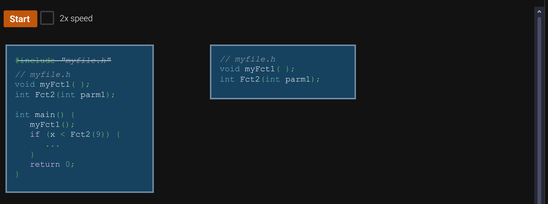
Construct 5.15.1: Include directives.

```
#include "filename"
#include <filename>
```

Feedback?

The following animation illustrates.

**PARTICIPATION ACTIVITY** 5.15.1: Preprocessor's handling of an include directive.




Captions

Feedback?

*Good practice is to use a .h suffix for any file that will be included in another file. The h is short for header, to indicate that the file is intended to be included at the top (or header) of other files.* Although any file can be included in any other file, convention is to only include .h files.

The characters surrounding the filename determine where the preprocessor looks for the file.

- `#include "myfile.h"` -- A filename in quotes causes the preprocessor to look for the file in the same folder/directory as the including file.
- `#include <stdfile>` -- A filename in angle brackets causes the preprocessor to look in the system's standard library folder/directory. Programmers typically use angle brackets only for standard library files, using quotes for all other include files. Note that nearly every previous example has included at least one standard library file, using angle brackets.

- Header files that are part of the standard C++ library do not have a .h extension.
- Items that were originally part of the C standard library have a "c" prepended, as in cmath.

**PARTICIPATION ACTIVITY** 5.15.2: Include directives.

1) The preprocessor processes any line beginning with what symbol?
   - # 
   - <filename>
   - "filename"

2) After a source file is processed by the preprocessor, is it correct to say that all hash symbols will be removed from the code remaining to be compiled?
   - yes
   - no

3) Do header files have to end in .h?
   - yes
   - no

4) Where does the preprocessor look for myfile.h in the line:
   `#include "myfile.h"`
   - Current folder
   - System folder
   - Unknown

5) What one symbol is incorrect in the following:
   `#include <stdlib.h>;`
   - #
   - <>
   - ;

Feedback?

Exploring further:

- Preprocessor tutorial on cplusplus.com
- Preprocessor directives on MSDN

**CHALLENGE ACTIVITY** 5.15.1: Preprocessor and include.

620890.5010016.qx3zqy7

Start

Include the following of the system's standard libraries:

- `iostream` for `cin` and `cout` to read from input and output boxWidth.
- `iomanip` for the `fixed` and `setprecision()` manipulators to format floating point numbers in output.

Ex: If the input is `37.5`, then the output is:

```
Box's width: 37.50
```

```
/* Your code goes here */

using namespace std;

int main() {
   double boxWidth;

   cin >> boxWidth;

   cout << "Box's width: " << fixed << setprecision(2) << boxWidth << endl;

   return 0;
}
```

1 2

Check Next level

Feedback?

How was this section? Provide section feedback

Activity summary for assignment: CSC108 CH05.8-5.16 C5B

Due: 04/15/2025, 11:59 PM EDT

0 / 19 points

Please browse to this assignment through BlackboardLearn so zyBooks knows where to send your activity. Learn more

Completion details