Control systems and Computer Networks

The C pre-processor & C Macros

Dr Alun Moon

Lecture 1.c

The C Pre-Processor

The C pre-processor has been a part of the C standard since the beginning of C.

It acts to transform the source code according to it's rules, prior to the modified code being fed to the compiler.

Note Well

This is a modification of the source code. The Compiler will report errors in the code after the pre-processor has modified the code

Errors introduced via the pre-processor can be ***very*** *hard* to debug!

Syntax

The syntax is very simple,

1. lines beginning with a hash character # are pre-processor directives
2. tokens are split using white-space

Includes

```
#include
```

The *#include* directive *copies in* the file given, at that point in the source code.

By convention these files are given the `.h` suffix, and are called *header files*

Files are searched for using two rules given by the kind of quotes used

- <> looks for standard header files in a system defined place (the include-path)
- "" looks for files in the same directory as the original source code

Define

```
#define BUFFSIZE 1024
#define pi 3.141592654
```

The simple version of *#define* creates a token and a *textual* substitution

Where the pre-processor finds a matching token in the source code, it is replaced with the text.

```
char inputbuffer[BUFFSIZE];→char inputbuffer[1024];
A = pi*r*r                 →A = 3.141592654*r*r
```