

CS113: Lecture 6

Topics:

- The C Preprocessor
- Examples

Remember the name: Pre-processor

- Most commonly used features: `#include`, `#define`.
- Think of the preprocessor as processing the file so as to “remove” all `#whatevers`.
- `#include` inserts the entire contents of a file, verbatim, exactly where the `#include` appeared.

Two ways to use:

- `#include "filename"` - typically searches for `filename` in the directory where the source program is
- `#include <filename>` - searches for `filename` in an implementation-defined way.

Typically, use angle brackets for standard library header files.

#define: a macro facility

```
#define SIZE 10
#define CENT_TO_INCHES 2.54

void main()
{
    int i;
    int inches[SIZE], centimeters[SIZE];

    /* read in inches[] ... */

    for( i = 0; i < SIZE; i++ )
        centimeters[i] = inches[i] * CENT_TO_INCHES;
}
```

Why use?

- Making global changes is easier.
- Makes programs easier to read (assuming descriptive names)

Some will go so far as to say that no constants should appear in your code (other than perhaps some zeroes and ones) - they should all be `#defined`.

#define: more details

- Substitutions only made for “tokens” – do not take place within quoted strings

- Possible to define macros with arguments:

```
#define sum(A,B) ((A)+(B))
```

```
x = sum( 3 * x, 1 ); becomes
```

```
x = (( 3 * x ) + ( 1 ));
```

- Be very, very careful.

```
#define square(x) x * x    /* Doh! */
```

```
y = 2 * square(z + 1);
```

- One use of #define was to speed up programs; I say go for readability.

```
#define max(x,y) ((x) > (y) ? (x) : (y))
```

More on the preprocessor

- Conditional compilation:

```
#define DEBUG

void main()
{
    int x;

    /* blah blah blah */

    #ifdef DEBUG
        printf( "current value of x is %d\n", x );
    #endif
}
```

- There are other features: `#if`, `#else`, etc.

Example: temperature table

```
#include <stdio.h>

void main()
{
    int fahr;
    for( fahr = 0; fahr <= 300; fahr += 20 )
    {
        printf( "%3d %6.1f\n", fahr,
                (5.0 / 9.0) * (fahr - 32));
    }
}
```

Example: string to int conversion

```
int string_to_int( char *s )
{
    /* assumes positive integer */
    /* returns -1 if string not an int */
    int i, mult = 1, value = 0;

    for( i = 0; s[i] != 0; i++ )
    {

    for( i--; i >= 0; i-- )
    {
        if( s[i] < '0' || s[i] > '9' )
            return( -1 );
        value += (s[i] - '0') * mult;
        mult *= 10;
    }
    return( value );
}

void main()
{
    printf( "value: %d\n", string_to_int("4213"));
}
```

Example: int to string conversion

```
void int_to_string( int a, char *s )
/* assumes s has enough space to hold the new string */
/* assumes a is positive */
{
    int i = 0;
    while( a > 0 )
    {
        s[i] = (a % 10) + '0';
        a /= 10;
        i++;
    }
    s[i] = 0;
    reverse( s );
}

void main()
{
    char s[10];
    int_to_string( 5497, s );
    printf( "string: %s\n", s );
}
```

Example: Encrypting a string

```
#include <stdio.h>

void encrypt( char *s )
{
    int i;
    for( i = 0; s[i] != 0; i++ )
    {
        if( s[i] >= 'a' && s[i] <= 'y' )
        {
            s[i]++;
        }
        else if( s[i] == 'z' )
            s[i] = 'a';
    }
}

void decrypt( char *s )
{
    int i;
    for( i = 0; s[i] != 0; i++ )
    {
        if( s[i] >= 'b' && s[i] <= 'z' )
        {
            s[i]--;
        }
        else if( s[i] == 'a' )
            s[i] = 'z';
    }
}
```

Example cont'd: Calling the functions

```
void main()
{
    char message[81];

    printf( "Enter a word in lower-case"
           "(80 chars. or less): " );
    scanf( "%s", message );

    encrypt( message );
    printf( "Here's the encrypted version: %s\n", message );
    decrypt( message );
    printf( "And the decrypted version: %s\n", message );
}
```

Word Counting

(Adapted from K & R.)

```
#include <stdio.h>

void main()
{
    int c, nl, nw, nc, state;
    int IN = 1, OUT = 0;

    state = OUT;
    nl = nw = nc = 0;

    while(( c = getchar()) != EOF )
    {
        nc++;
        if( c == '\n' )
            nl++;
        if( c == ' ' || c == '\n' || c == '\t' )
            state = OUT;
        else if( state == OUT )
        {
            state = IN;
            nw++;
        }
    }
}
```