
C: more macros, separate files

CSE 413, Autumn 2007
10-29-2007

1

Topics

- more macros
- sf printf & scanf
- multiple files

2

Macros can take parameters

```
#define TWICE_AWFUL(x) x*2
#define TWICE_BAD(x) ((x)+(x))
#define TWICE_OK(x) ((x)*2)
```

Each "call" to a macro is replaced with the body (with all instances of the formal arguments (x in this example) replaced with whatever is "passed in" as arguments to the macro.

What happens here?

```
y=3;
z=4;
w=TWICE_AWFUL(y+z);

y=7;
z=TWICE_BAD(++y);
z=TWICE_BAD(y++);
```

3

A useful macro

```
#define NEW(type,count) (type*)malloc(sizeof(type)*(count))
```

Example Uses:

```
char* c = NEW(char,100);
struct MYPOINT* myPoint = NEW(struct MYPOINT,1);
```

4

[sf]printf

```
int printf(const char *format, ... );
int sprintf(char *s, const char *format, ... );
int fprintf(FILE *stream, char *s, const char *format, ... );
```

```
char result[100]; int answer;
sprintf(result, "Answer is %d\n", answer);
```

sprintf is the same as **printf**, except it prints output to the string ***s**.

- » ***s** must be big enough to hold the string!
- » **sprintf** will null terminate the string.

5

[sf]scanf

```
int scanf(const char *format, ... );
int sscanf(char *s, const char *format, ... );
int fscanf(FILE *stream, const char *format, ... );
```

```
char name[100]; int x;
scanf("%s %d\n", name, &x);
```

Reads from **stdin**, trying to match the format string.

- » what is read must match the format string exactly (or runtime error)
- » returns the # of %s that matched successfully
- » %s reads up to the next whitespace
- » **name** better have room for whatever %s is...

6

Multiple files

foo.h

- » contains prototypes for functions in foo.c

foo.c

- » contains code bodies for function prototypes in foo.h
- » may contain static functions that are for internal use only

test.c

- » Client code that uses functions from foo.h, should #include foo.h at the top of the file.

7