

## C Standard Library

# CS 241

## Data Organization using C

Instructor: **Joel Castellanos**

e-mail: [joel@unm.edu](mailto:joel@unm.edu)

Web: <http://cs.unm.edu/~joel/>

Office: Farris Engineering  
Center (FEC) room 321

Lab Instructor: **Dongye Meng**

e-mail: [dymeng@cs.unm.edu](mailto:dymeng@cs.unm.edu)



3/6/2009

## The Standard C Library by Plauger

### THE STANDARD

<time.h> \* <limits.h> \* <float.h>  
<stddef.h> \* <locale.h>  
<stdio.h> \* <ctype.h> \* <string.h>  
<math.h> \* <stdlib.h> \* <assert.h>  
<stdarg.h> \* <setjmp.h> \* <signal.h>  
<time.h> \* <limits.h> \* <float.h>  
<stddef.h> \* <errno.h> \* <locale.h>  
<stdio.h> \* <ctype.h> \* <string.h>  
<math.h> \* <stdlib.h> \* <assert.h>  
<stdarg.h> \* <setjmp.h> \* <signal.h>  
<time.h> \* <limits.h> \* <float.h>  
<stddef.h> \* <errno.h> \* <locale.h>

### LIBRARY

P. J. PLAUGER

- Comprehensive treatment of ANSI and ISO standards for the C Library.
- Contains the complete code of the Standard C Library and includes practical advice on using all 15 headers.
- Focus on the concepts, design issues, and trade-offs associated with library building.
- Using this book, programmers will make the best use of the C Library and will learn to build programs with maximum portability and reusability.

2

## #include <stdlib.h>

- `exit(int status)` function terminates the current process.
- Before termination, `exit` performs the following functions in the order listed:
  - Call the functions registered with the `atexit` function, in the reverse order of their registration (`atexit` registers functions to be called at program exit).
  - Flush all open output streams.
  - Close all open streams.
  - Unlink all files created with the `tmpfile` function.
- Standard C defines the values 0, `EXIT_SUCCESS`, and `EXIT_FAILURE` as possible values of status.

3

## C String Library

See page 249 of your textbook (Appendix B)

`#include <string.h>`

`char *strcpy(s, ct)`: copy string `ct` to string `s`, including `'\0'`; returns `s`.

`char *strncpy(s, n, ct)`: copy at most `n` characters of string `ct` to string `s`, including `'\0'` iff `ct` has fewer than `n` characters; returns `s`.

`char *strcmp(cs, ct)`: compare `cs` to string `ct`; returns `<0` if `cs < ct`, `0` if `cs == ct`, or `>0` if `cs > ct`.

Names like these are left over from the odd days

4

## C String Library: `strlen`

`strlen(s)` returns the number of bytes in the string to which `s` points, not including the terminating NULL byte.

```
#include <stdio.h>
int strlen(char* str)
{ int i=0;
  while (str[i]){ i++;}
  return i; }

int main()
{ char *myName = "Joel Castellanos";
  printf("%d\n", strlen(myName));
  return 0;
}
```

Ok to return a local basic type because it is return by value.

5

## `char* strchr(char* s, char c)`

- locates the first occurrence of `c` in the string pointed to by `s`. The terminating null byte is considered to be part of the string.
- Upon successful completion, `strchr()` returns a pointer to the byte or a null pointer if `c` does not occur in the string.
- Implement `strchr`

6

## C Math Library

```
#include <math.h>
```

Including `math.h` will tell the compiler that the math functions like `sqrt()` exist

```
gcc foo.c -lm
```

- gcc option telling the linker to link with the math library. The linker needs this to package with your executable.
- The library is called `libm.a`.
- On Unix-like systems, the rule for naming libraries is `lib[xxx].a`. Then if you want to link them to your executable you use `-l[xxx]`.

7

## <math.h>

1. `double exp(double x);` exponential of `x`.
2. `double log(double x);` natural logarithm of `x`.
3. `double log10(double x)` base-10 logarithm of `x`.
4. `double pow(double x, double y);` `x` raised to power `y`.
5. `double sqrt(double x);` square root of `x`.
6. `double ceil(double x);` smallest integer not less than `x`.
7. `double floor(double x);` largest integer not greater than `x`.
8. `double fabs(double x);` absolute value of `x`.

8