

Function Pointer and C Standard Library

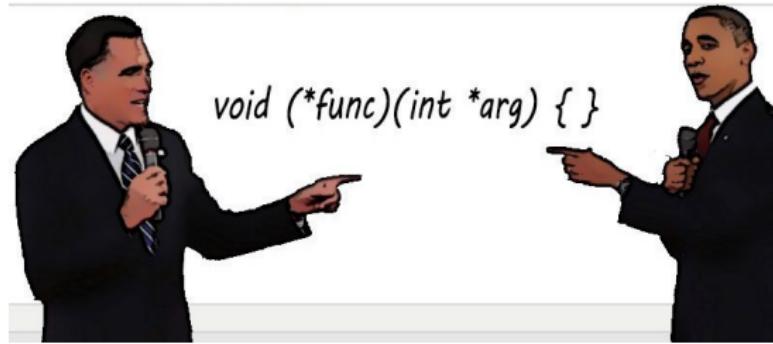
魏恒峰

hfwei@nju.edu.cn

2017 年 11 月 24 日



Function Pointer



Sort for any types

COMPARE & SWAP

sort_ints

sort_floats

sort_strings

sort_persons

...

```
(#include <stdlib.h>

void qsort (void *base, size_t num, size_t size,
            int (*compar)(const void*, const void*));
```

(compare.c)

```
int (*fptr)(int); // fptr is a function pointer

int square(int num) {
    return num * num;
}
```

```
int n = 5;
fptr = square; // fptr points to a function
fptr(n); // call 'square'
```

(fptr-square.c)

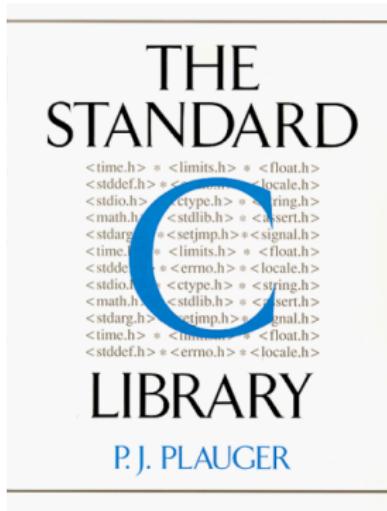
```
typedef int (*predicate)(int n); // type!
```



```
void filter(int *vals, int n, predicate cond);
```

(filter-fp.c)

C Standard Library



The screenshot shows a web browser displaying the cplusplus.com website. The main header features a blue 3D cube logo followed by the text "cplusplus.com". Below the header, a dark blue navigation bar contains the word "C++" in white. To the right of the navigation bar is a vertical sidebar with a light gray background. The sidebar lists several categories in purple text: "Information", "Tutorials", "Reference", "Articles", and "Forum". The "Reference" category is highlighted with a gray background. Below the sidebar, a larger content area has a dark blue header with the word "Reference" in white. To the right of this header is a small "az" icon. The main content area lists five sections, each preceded by a blue square icon with a plus sign: "C library:", "Containers:", "Input/Output:", "Multi-threading:", and "Other:". The entire screenshot is set against a red background.

C++

Information

Tutorials

Reference

Articles

Forum

Reference az

- + **C library:**
- + **Containers:**
- + **Input/Output:**
- + **Multi-threading:**
- + **Other:**

C library:

- ... <cassert> (assert.h)
- ... <cctype> (ctype.h)
- ... <cerrno> (errno.h)
- ... <cfenv> (fenv.h)
- ... <cfloat> (float.h)
- ... <cinttypes> (inttypes.h)
- ... <ciso646> (iso646.h)
- ... <climits> (limits.h)
- ... <clocale> (locale.h)
- ... <cmath> (math.h)
- ... <csetjmp> (setjmp.h)
- ... <csignal> (signal.h)
- ... <cstdarg> (stdarg.h)
- ... <cstdbool> (stdbool.h)
- ... <cstddef> (stddef.h)
- ... <cstdint> (stdint.h)
- ... <cstdio> (stdio.h)
- ... <cstdlib> (stdlib.h)
- ... <cstring> (string.h)
- ... <ctgmath> (tgmath.h)
- ... <ctime> (time.h)
- ... <cuchar> (uchar.h)
- ... <cwchar> (wchar.h)
- ... <cwctype> (wctype.h)

```
(#include <assert.h>)
```

```
scanf('' %d '' , &n);  
assert(n > 0)
```

```
int *vals = malloc(sizeof(int) * n);  
assert(vals != NULL)
```

```
(#include <ctype.h>)
```

```
isdigit
```

```
isalpha
```

```
isalnum
```

```
islower
```

```
isupper
```

```
isspace
```

```
tolower
```

```
toupper
```

```
(#include <limits.h>)
```

```
CHAR_MIN
```

```
CHAR_MAX
```

```
INT_MIN
```

```
INT_MAX
```

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

```
sin
```

```
cos
```

```
exp
```

```
log
```

```
pow
```

```
sqrt
```

```
ceil
```

```
floor
```

```
(#include <stdarg.h>)
```

```
int printf(const char *format, ...);
```

```
printf('' %d %c %s '' , num, ch, str);
```

- (1) unnamed va_list // type
- (2) # unknown va_start
- (3) “...” must be at the end va_arg
- (4) ≥ 1 named argument va_end

(miniprintf.c)

(Command-line arguments)

```
int main(int argc, char *argv[]) {  
}
```

argv[0]: program name

(add-cmdarg.c)

(man ls)

```
(#include <stddef.h>)

typedef /*implementation-defined*/ size_t;

sizeof(int)

void* malloc (size_t size);
```

```
(#include <stdio.h>)
```

```
scanf
```

```
printf
```

```
getchar
```

```
putchar
```

```
fopen
```

```
fclose
```

```
EOF
```

```
(#include <stdlib.h>)
```

```
atoi
```

```
atof
```

```
srand
```

```
rand
```

```
malloc
```

```
free
```

```
bsearch
```

```
qsort
```

```
(#include <string.h>)
```

```
char book[] = "The C Book";
char *pbook = "The C Book";
```



'\0'

strncpy
strncat
strcmp

strlen

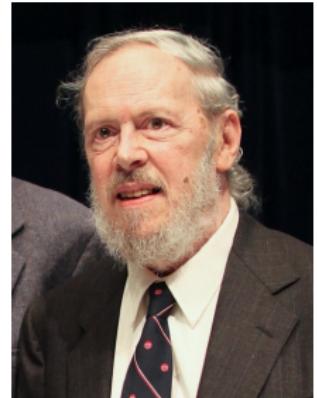
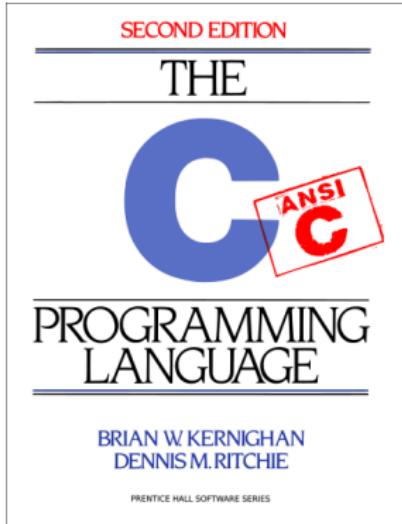
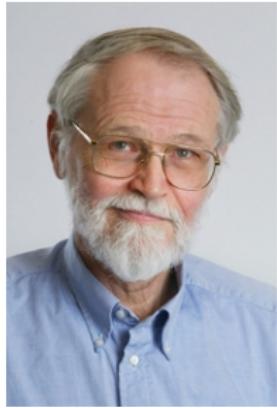
strchr
 strrchr
 strstr

strtok

```
(#include <time.h>)
```



Highly Recommended!





Thank You!