

## Op:string.h

string.h - string operations

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

**Contents** [1 DESCRIPTION](#) [2 APPLICATION USAGE](#) [3 RATIONALE](#) [4 FUTURE DIRECTIONS](#) [5 RELATED](#)  
[6 COPYRIGHT](#) [7 CATEGORY](#)

### DESCRIPTION

Some of the functionality described on this reference page extends the ISO C standard. Applications shall define the appropriate feature test macro (see the System Interfaces volume of IEEE Std 1003.1-2001, Section 2.2, The Compilation Environment) to enable the visibility of these symbols in this header.

The `<string.h>` header shall define the following:

`NULL` Null pointer constant.

`size_t` As described in `<stddef.h>` .

The following shall be declared as functions and may also be defined as macros. Function prototypes shall be provided.

```
void      *memccpy(void *restrict, const void *restrict, int, size_
t);

void      *memchr(const void *, int, size_t);
int       memcmp(const void *, const void *, size_t);
void      *memcpy(void *restrict, const void *restrict, size_t);
void      *memmove(void *, const void *, size_t);
```

```

void      *memset(void *, int, size_t);
char      *strcat(char *restrict, const char *restrict);
char      *strchr(const char *, int);
int       strcmp(const char *, const char *);
int       strcoll(const char *, const char *);
char      *strcpy(char *restrict, const char *restrict);
size_t    strcspn(const char *, const char *);

char      *strdup(const char *);

char      *strerror(int);

int       *strerror_r(int, char *, size_t);

size_t    strlen(const char *);
char      *strncat(char *restrict, const char *restrict, size_t);
int       strncmp(const char *, const char *, size_t);
char      *strncpy(char *restrict, const char *restrict, size_t);
char      *strpbrk(const char *, const char *);
char      *strrchr(const char *, int);
size_t    strspn(const char *, const char *);
char      *strstr(const char *, const char *);
char      *strtok(char *restrict, const char *restrict);

char      *strtok_r(char *, const char *, char **);

size_t    strxfrm(char *restrict, const char *restrict, size_t);

```

Inclusion of the `<string.h>` header may also make visible all symbols from `<stddef.h>`.

*The following sections are informative.*

#### **APPLICATION USAGE**

None.

#### **RATIONALE**

None.

#### **FUTURE DIRECTIONS**

None.

#### **RELATED**

`<stddef.h>` , `<sys/types.h>` , the System Interfaces volume of IEEE Std 1003.1-2001, `memcpy()`, `memchr()`, `memcmp()`, `memcpy()`, `memmove()`, `memset()`, `strcat()`, `strchr()`, `strcmp()`, `strcoll()`, `strcpy()`, `strcspn()`, `strdup()`, `strerror()`, `strlen()`, `strncat()`, `strncmp()`, `strncpy()`, `strpbrk()`, `strrchr()`, `strspn()`, `strstr()`, `strtok()`, `strxfrm()`

#### **COPYRIGHT**

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1, 2003 Edition, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 6, Copyright (C) 2001-2003 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the

original IEEE and The Open Group Standard,  
the original IEEE and The Open Group Standard is the referee  
document. The original Standard can be obtained  
online at <http://www.opengroup.org/unix/online.html> .

IEEE/The Open Group 2003 <string.h>(P)