

**NAME**

**getsubopt** - get sub options from an argument

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

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

```
extern char *suboptarg;
```

```
int
```

```
getsubopt(char **optionp, char * const *tokens, char **valuep);
```

**DESCRIPTION**

The **getsubopt()** function parses a string containing tokens delimited by one or more tab, space or comma (',') characters. It is intended for use in parsing groups of option arguments provided as part of a utility command line.

The argument *optionp* is a pointer to a pointer to the string. The argument *tokens* is a pointer to a NULL-terminated array of pointers to strings.

The **getsubopt()** function returns the zero-based offset of the pointer in the *tokens* array referencing a string which matches the first token in the string, or, -1 if the string contains no tokens or *tokens* does not contain a matching string.

If the token is of the form "name=value", the location referenced by *valuep* will be set to point to the start of the "value" portion of the token.

On return from **getsubopt()**, *optionp* will be set to point to the start of the next token in the string, or the null at the end of the string if no more tokens are present. The external variable *suboptarg* will be set to point to the start of the current token, or NULL if no tokens were present. The argument *valuep* will be set to point to the "value" portion of the token, or NULL if no "value" portion was present.

**EXAMPLES**

```
char *tokens[] = {
    #define ONE    0
        "one",
    #define TWO    1
        "two",
```

```
        NULL
    };

    ...

extern char *optarg, *suboptarg;
char *options, *value;

while ((ch = getopt(argc, argv, "ab:")) != -1) {
    switch(ch) {
        case 'a':
            /* process ‘a’ option */
            break;
        case 'b':
            options = optarg;
            while (*options) {
                switch(getsubopt(&options, tokens, &value)) {
                    case ONE:
                        /* process ‘one’ sub option */
                        break;
                    case TWO:
                        /* process ‘two’ sub option */
                        if (!value)
                            error("no value for two");
                        i = atoi(value);
                        break;
                    case -1:
                        if (suboptarg)
                            error("illegal sub option %s",
                                suboptarg);
                        else
                            error("missing sub option");
                        break;
                }
            }
            break;
    }
}
```

**SEE ALSO**

getopt(3), strsep(3)

## **HISTORY**

The **getsubopt()** function first appeared in 4.4BSD.