

**NAME**

getfiles() - gets next non flag-type argument

**SYNOPSIS**

```
#include <schily/getargs.h>
```

```
int getfiles(pac, pav, fmt)
```

```
    int *pac;    /* pointer to arg count */
    char *(*pav)[]; /* pointer to address of arg vector */
    char *fmt;   /* format string */
```

```
int getlfiles(pac, pav, props, fmt)
```

```
    int *pac;    /* pointer to arg count */
    char *(*pav)[]; /* pointer to address of arg vector */
    struct ga_props *props; /* control properties */
    char *fmt;   /* format string */
```

```
int getvfiles(pac, pav, props, vfmt)
```

```
    int *pac;    /* pointer to arg count */
    char *(*pav)[]; /* pointer to address of arg vector */
    struct ga_props *props; /* control properties */
    struct ga_flags *vfmt; /* array of formats and args */
```

**DESCRIPTION**

**getfiles()** is part of the advanced option parsing interface together with the **getargs()** and **getallrgs()** family. **getfiles()** skips options until a file type argument is encountered.

**getfiles()** returns, with a value of +1 (**NOTAFLAG**) or +2 (**FLAGDELIM**), whenever a non-flag type argument is encountered. Flag-type (option) arguments, along with their values, are ignored. The filename is at **\*pav[0]**. The caller must increment **\*pav** and decrement **\*pac** before calling **getfiles()** again.

**getlfiles()** is similar to **getfiles()** but it implements an additional **ga\_props** parameter that must be initialized with **getarginit()** before it is passed.

**getvfiles()** is similar to **getlfiles()** but uses a structure **ga\_flags** instead of a format string and a variable arg list with pointers. The array of structures **ga\_flags**:

```
struct ga_flags {
    const char *ga_format; /* Comma separated list for one flag */
```

```

void    *ga_arg; /* Ptr. to variable to fill for flag */
getpargfun ga_funcp; /* Ptr. for function to call (&/~) */
};

```

is terminated by an element with **ga\_format** == **NULL**. For a **ga\_format** that does not expect a function pointer, **ga\_funcp** is **NULL**.

## RETURNS

**FLAGDELIM 2** The command line argument "--" stopped flag processing.

**NOTAFLAG 1** The argument **\*pav** does not appear to be a flag.

**NOARGS 0** All arguments have been successfully examined.

**BADFLAG -1** A bad flag (option) argument was supplied to the program. The argument **\*pav** contains the offending command line argument.

**BADFMT -2** A bad format descriptor string has been detected. This means an error in the calling program, not a user input data error.

General rules for the return code:

> 0            A file type argument was found.

0             All arguments have been parsed.

< 0            An error occurred or not a file type argument.

Flag and file arg processing should be terminated after getting a return code <= 0.

## SEE ALSO

**getallargs(3)**, **getargerror(3)**, **getargs(3)**.

## NOTES

Initially, **\*pav** must point to the first argument (not the program name) and **\*pac** must not count the program name. Before calling **getfiles()** again, decrement **\*pac** and increment **\*pav**, or the same name will be pointed to by **\*pav**.

**BUGS**

None currently known.

Mail bugs and suggestions to [schilytools@mlists.in-berlin.de](mailto:schilytools@mlists.in-berlin.de) or open a ticket at <https://codeberg.org/schilytools/schilytools/issues>.

The mailing list archive may be found at:

<https://mlists.in-berlin.de/mailman/listinfo/schilytools-mlists.in-berlin.de>.

**AUTHOR**

Joerg Schilling and the schilytools project authors.