

```

1:  /* Copyright (c) 1993 UNIX System Laboratories, Inc.  */
2:  /* (a wholly-owned subsidiary of Novell, Inc.).  */
3:  /* All Rights Reserved.  */
4:
5:  /* THIS IS UNPUBLISHED PROPRIETARY SOURCE CODE OF UNIX SYSTEM  */
6:  /* LABORATORIES, INC. (A WHOLLY-OWNED SUBSIDIARY OF NOVELL, INC.).  */
7:  /* The copyright notice above does not evidence any actual or  */
8:  /* intended publication of such source code.  */
9:
10: #ifndef _STROPTS_H
11: #define _STROPTS_H
12:
13: #ident      "@(#)sgs-head:common/head/stropts.h  1.10"
14:
15: #if defined(__cplusplus)
16: extern "C" {
17: #endif
18:
19: /*
20:  * Streams user options definitions.
21:  */
22:
23: #ifndef _SYS_STROPTS_H
24: #include <sys/stropts.h>
25: #endif
26:
27: #if defined(__STDC__)
28:
29: extern int getmsg(int, struct strbuf *, struct strbuf *, int *);
30: extern int putmsg(int, const struct strbuf *, const struct strbuf *, int);
31:
32: extern int getpmsg(int, struct strbuf *, struct strbuf *, int *, int *);
33: extern int putpmsg(int, const struct strbuf *, const struct strbuf *, int, int);
34:
35: extern int isastream(int);
36:
37: #else
38:
39: extern int getmsg();
40: extern int putmsg();
41:
42: extern int getpmsg();
43: extern int putpmsg();
44:
45: extern int isastream();
46:
47: #endif
48:
49: #if defined(__cplusplus)
50: }
51: #endif
52: #endif /* _STROPTS_H */

```

```

1:  /* Copyright (C) 1998, 1999, 2000 Free Software Foundation, Inc.
2:  This file is part of the GNU C Library.
3:
4:  ...
5:
6: 36:
37: /* Get system specific contants.  */
38: #include <bits/stropts.h>
39:
40:
41: __BEGIN_DECLS
42:
43: /* Test whether FILDES is associated with a STREAM-based file.  */
44: extern int isastream (int __fildes) __THROW;
45:
46: /* Receive next message from a STREAMS file.  */
47: extern int getmsg (int __fildes, struct strbuf *__restrict __ctlptr,
48:                  struct strbuf *__restrict __dataptr,
49:                  int *__restrict __flagsp) __THROW;
50:
51: /* Receive next message from a STREAMS file, with *FLAGSP allowing to
52: control which message.  */
53: extern int getpmsg (int __fildes, struct strbuf *__restrict __ctlptr,
54:                   struct strbuf *__restrict __dataptr,
55:                   int *__restrict __bandp, int *__restrict __flagsp)
56:                   __THROW;
57:
58: /* Perform the I/O control operation specified by REQUEST on FD.
59: One argument may follow; its presence and type depend on REQUEST.
60: Return value depends on REQUEST. Usually -1 indicates error.  */
61: extern int ioctl (int __fd, unsigned long int __request, ...) __THROW;
62:
63: /* Send a message on a STREAM.  */
64: extern int putmsg (int __fildes, __const struct strbuf *__ctlptr,
65:                  __const struct strbuf *__dataptr, int __flags) __THROW;
66:
67: /* Send a message on a STREAM to the BAND.  */
68: extern int putpmsg (int __fildes, __const struct strbuf *__ctlptr,
69:                   __const struct strbuf *__dataptr, int __band, int __flags)
70:                   __THROW;
71:
72: /* Attach a STREAMS-based file descriptor FILDES to a file PATH in the
73: file system name space.  */
74: extern int fattach (int __fildes, __const char *__path) __THROW;
75:
76: /* Detach a name PATH from a STREAMS-based file descriptor.  */
77: extern int fdetach (__const char *__path) __THROW;
78:
79: __END_DECLS
80:
81: #endif /* stropts.h */

```