**NAME**
     XtAppAddInput, XtRemoveInput - register and remove an input source

**SYNTAX**
     #include <X11/Intrinsic.h>

     XtInputId XtAppAddInput(XtAppContext *app_context*, int *source*, XtPointer *condition*,
          XtInputCallbackProc *proc*, XtPointer *client_data*);

     void XtRemoveInput(XtInputId *id*);

**ARGUMENTS**

| | |
|---|---|
| *app_context* | Specifies the application context that identifies the application. |
| *client_data* | Specifies the argument that is to be passed to the specified procedure when input is available. |
| *condition* | Specifies the mask that indicates a read, write, or exception condition or some operating system dependent condition. |
| *id* | Specifies the ID returned from the corresponding **XtAppAddInput** call. |
| *proc* | Specifies the procedure that is to be called when input is available. |
| *source* | Specifies the source file descriptor on a UNIX-based system or other operating system dependent device specification. |

**DESCRIPTION**
     The **XtAppAddInput** function registers with the Intrinsics read routine a new source of events, which is
     usually file input but can also be file output. Note that file should be loosely interpreted to mean any
     sink or source of data. **XtAppAddInput** also specifies the conditions under which the source can
     generate events. When input is pending on this source, the callback procedure is called.

     The legal values for the condition argument are operating-system dependent. On a UNIX-based
     system, the condition is some union of **XtInputReadMask**, **XtInputWriteMask**, and
     **XtInputExceptMask**. The **XtRemoveInput** function causes the Intrinsics read routine to stop watching
     for input from the input source.

**SEE ALSO**
     XtAppAddTimeOut(3),XtAppAddSignal(3)

*X Toolkit Intrinsics - C Language Interface*
*Xlib - C Language X Interface*

**BUGS**

In ANSI C it is necessary to cast the condition to an XtPointer, e.g.:

    XtAppAddInput(app_context,
        source,
        **(XtPointer)** (XtInputReadMask | XtInputWriteMask),
        proc,
        client_data);