_____unchanged_portion_omitted_
  89 /*
  90 * Some entries in this structure have both leading _ and non leading _
  91 * versions for compatibility with various versions of the specifications.
  92 * It is not expected that both forms will be required for new entries.
  93 */

  95   /* POSIX.2-1992 table 2-17 */
  96   _POSIX2_BC_BASE_MAX,   "POSIX2_BC_BASE_MAX",   SELFCONF, NOFLAGS,
  97   _POSIX2_BC_BASE_MAX,   "_POSIX2_BC_BASE_MAX", SELFCONF, NOFLAGS,
  98   _POSIX2_BC_DIM_MAX,   "POSIX2_BC_DIM_MAX",   SELFCONF, NOFLAGS,
  99   _POSIX2_BC_DIM_MAX,   "_POSIX2_BC_DIM_MAX",   SELFCONF, NOFLAGS,
 100   _POSIX2_BC_SCALE_MAX,   "POSIX2_BC_SCALE_MAX", SELFCONF, NOFLAGS,
 101   _POSIX2_BC_SCALE_MAX,   "_POSIX2_BC_SCALE_MAX", SELFCONF, NOFLAGS,
 102   _POSIX2_BC_STRING_MAX, "POSIX2_BC_STRING_MAX", SELFCONF, NOFLAGS,
 103   _POSIX2_BC_STRING_MAX, "_POSIX2_BC_STRING_MAX", SELFCONF, NOFLAGS,
 104   _POSIX2_COLL_WEIGHTS_MAX, "POSIX2_COLL_WEIGHTS_MAX", SELFCONF, NOFLAGS,
 105   _POSIX2_COLL_WEIGHTS_MAX, "_POSIX2_COLL_WEIGHTS_MAX", SELFCONF, NOFLAGS,
 106   _POSIX2_EXPR_NEST_MAX, "POSIX2_EXPR_NEST_MAX", SELFCONF, NOFLAGS,
 107   _POSIX2_EXPR_NEST_MAX, "_POSIX2_EXPR_NEST_MAX", SELFCONF, NOFLAGS,
 108   _POSIX2_LINE_MAX,   "POSIX2_LINE_MAX",   SELFCONF, NOFLAGS,
 109   _POSIX2_LINE_MAX,   "_POSIX2_LINE_MAX",   SELFCONF, NOFLAGS,
 110   _POSIX2_RE_DUP_MAX,   "POSIX2_RE_DUP_MAX",   SELFCONF, NOFLAGS,
 111   _POSIX2_RE_DUP_MAX,   "_POSIX2_RE_DUP_MAX",   SELFCONF, NOFLAGS,
 112   /* See limits.h section below for _POSIX_RE_DUP_MAX */

 114   /* UNIX 98 */
 115   _POSIX_AIO_LISTIO_MAX, "_POSIX_AIO_LISTIO_MAX", SELFCONF,   NOFLAGS,
 116   _POSIX_AIO_MAX,   "_POSIX_AIO_MAX",   SELFCONF,   NOFLAGS,
 117   _POSIX_DELAYTIMER_MAX, "_POSIX_DELAYTIMER_MAX", SELFCONF,   NOFLAGS,
 118   _POSIX_LOGIN_NAME_MAX, "_POSIX_LOGIN_NAME_MAX", SELFCONF,   NOFLAGS,
 119   _POSIX_MQ_OPEN_MAX,   "_POSIX_MQ_OPEN_MAX",   SELFCONF,   NOFLAGS,
 120   _POSIX_MQ_PRIO_MAX,   "_POSIX_MQ_PRIO_MAX",   SELFCONF,   NOFLAGS,
 121   _POSIX_RTSIG_MAX,   "_POSIX_RTSIG_MAX",   SELFCONF,   NOFLAGS,
 122   _POSIX_SEM_NSEMS_MAX,   "_POSIX_SEM_NSEMS_MAX", SELFCONF,   NOFLAGS,
 123   _POSIX_SEM_VALUE_MAX,   "_POSIX_SEM_VALUE_MAX", SELFCONF,   NOFLAGS,
 124   _POSIX_SIGQUEUE_MAX,   "_POSIX_SIGQUEUE_MAX", SELFCONF,   NOFLAGS,
 125   _POSIX_TTY_NAME_MAX,   "_POSIX_TTY_NAME_MAX", SELFCONF,   NOFLAGS,

 127   /* UNIX 03 */
 128   _POSIX_HOST_NAME_MAX,   "_POSIX_HOST_NAME_MAX", SELFCONF, NOFLAGS,
 129   _POSIX_RE_DUP_MAX,   "_POSIX_RE_DUP_MAX",   SELFCONF, NOFLAGS,
 130   _POSIX_CLOCKRES_MIN,   "_POSIX_CLOCKRES_MIN",   SELFCONF, NOFLAGS,
 131   _POSIX_SYMLINK_MAX,   "_POSIX_SYMLINK_MAX",   SELFCONF, NOFLAGS,
 132   _POSIX_SYMLOOP_MAX,   "_POSIX_SYMLOOP_MAX",   SELFCONF, NOFLAGS,
 133   CHARCLASS_NAME_MAX, "_POSIX2_CHARCLASS_NAME_MAX", SELFCONF, NOFLAGS,

 135   /* POSIX.2-1992 table 2-17 */
 136   _SC_2_VERSION,   "POSIX2_VERSION",   SYSCONF,   NOFLAGS,
 137   _SC_2_VERSION,   "_POSIX2_VERSION",   SYSCONF,   NOFLAGS,

 139   /* POSIX.2-1992 table 2-18 */
 140   _SC_BC_BASE_MAX,   "BC_BASE_MAX",   SYSCONF,   NOFLAGS,
 141   _SC_BC_DIM_MAX,   "BC_DIM_MAX",   SYSCONF,   NOFLAGS,
 142   _SC_BC_SCALE_MAX,   "BC_SCALE_MAX",   SYSCONF,   NOFLAGS,
 143   _SC_BC_STRING_MAX,   "BC_STRING_MAX",   SYSCONF,   NOFLAGS,
 144   _SC_COLL_WEIGHTS_MAX,   "COLL_WEIGHTS_MAX",   SYSCONF,   NOFLAGS,
 145   _SC_EXPR_NEST_MAX,   "EXPR_NEST_MAX",   SYSCONF,   NOFLAGS,
 146   _SC_LINE_MAX,   "LINE_MAX",   SYSCONF,   NOFLAGS,
 147   _SC_RE_DUP_MAX,   "RE_DUP_MAX",   SYSCONF,   NOFLAGS,

 149   /* POSIX.2-1992 table 2-19 */
 150   _SC_2_C_BIND,   "POSIX2_C_BIND",   SYSCONF,   NOFLAGS,
 151   _SC_2_C_BIND,   "_POSIX2_C_BIND",   SYSCONF,   NOFLAGS,
 152   _SC_2_C_DEV,   "POSIX2_C_DEV",   SYSCONF,   NOFLAGS,
 153   _SC_2_C_DEV,   "_POSIX2_C_DEV",   SYSCONF,   NOFLAGS,
 154   _SC_2_CHAR_TERM,   "POSIX2_CHAR_TERM",   SYSCONF,   NOFLAGS,
 155   _SC_2_CHAR_TERM,   "_POSIX2_CHAR_TERM",   SYSCONF,   NOFLAGS,
 156   _SC_2_FORT_DEV,   "POSIX2_FORT_DEV",   SYSCONF,   NOFLAGS,
 157   _SC_2_FORT_DEV,   "_POSIX2_FORT_DEV",   SYSCONF,   NOFLAGS,
 158   _SC_2_FORT_RUN,   "POSIX2_FORT_RUN",   SYSCONF,   NOFLAGS,
 159   _SC_2_FORT_RUN,   "_POSIX2_FORT_RUN",   SYSCONF,   NOFLAGS,
 160   _SC_2_LOCALEDEF,   "POSIX2_LOCALEDEF",   SYSCONF,   NOFLAGS,
 161   _SC_2_LOCALEDEF,   "_POSIX2_LOCALEDEF",   SYSCONF,   NOFLAGS,
 162   _SC_2_SW_DEV,   "POSIX2_SW_DEV",   SYSCONF,   NOFLAGS,
 163   _SC_2_SW_DEV,   "_POSIX2_SW_DEV",   SYSCONF,   NOFLAGS,
 164   _SC_2_UPE,   "POSIX2_UPE",   SYSCONF,   NOFLAGS,
 165   _SC_2_UPE,   "_POSIX2_UPE",   SYSCONF,   NOFLAGS,

 167   /* POSIX.1-1990 table 2-3 */
 168   _POSIX_ARG_MAX,   "_POSIX_ARG_MAX",   SELFCONF, NOFLAGS,
 169   _POSIX_CHILD_MAX,   "_POSIX_CHILD_MAX",   SELFCONF, NOFLAGS,
 170   _POSIX_LINK_MAX,   "_POSIX_LINK_MAX",   SELFCONF, NOFLAGS,
 171   _POSIX_MAX_CANON,   "_POSIX_MAX_CANON",   SELFCONF, NOFLAGS,
 172   _POSIX_MAX_INPUT,   "_POSIX_MAX_INPUT",   SELFCONF, NOFLAGS,
 173   _POSIX_NAME_MAX,   "_POSIX_NAME_MAX",   SELFCONF, NOFLAGS,
 174   _POSIX_NGROUPS_MAX,   "_POSIX_NGROUPS_MAX",   SELFCONF, NOFLAGS,
 175   _POSIX_OPEN_MAX,   "_POSIX_OPEN_MAX",   SELFCONF, NOFLAGS,
 176   _POSIX_PATH_MAX,   "_POSIX_PATH_MAX",   SELFCONF, NOFLAGS,
 177   _POSIX_PIPE_BUF,   "_POSIX_PIPE_BUF",   SELFCONF, NOFLAGS,
 178   _POSIX_SSIZE_MAX,   "_POSIX_SSIZE_MAX",   SELFCONF, NOFLAGS,
 179   _POSIX_STREAM_MAX,   "_POSIX_STREAM_MAX",   SELFCONF, NOFLAGS,
 180   _POSIX_TIMER_MAX,   "_POSIX_TIMER_MAX",   SELFCONF, NOFLAGS,
 181   _POSIX_THREAD_DESTRUCTOR_ITERATIONS,
 182   "_POSIX_THREAD_DESTRUCTOR_ITERATIONS", SELFCONF, NOFLAGS,
 183   _POSIX_THREAD_KEYS_MAX,
 184   "_POSIX_THREAD_KEYS_MAX",   SELFCONF, NOFLAGS,
 185   _POSIX_THREAD_THREADS_MAX,
 186   "_POSIX_THREAD_THREADS_MAX",   SELFCONF, NOFLAGS,
 187   _POSIX_TZNAME_MAX,   "_POSIX_TZNAME_MAX",   SELFCONF, NOFLAGS,

 189   /* POSIX.1-1990 table 4-2 */
 190   _SC_ARG_MAX,   "ARG_MAX",   SYSCONF,   NOFLAGS,
 191   _SC_CHILD_MAX,   "CHILD_MAX",   SYSCONF,   NOFLAGS,
 192   _SC_CLK_TCK,   "CLK_TCK",   SYSCONF,   NOFLAGS,
 193   _SC_JOB_CONTROL,   "_POSIX_JOB_CONTROL",   SYSCONF,   NOFLAGS,
 194   _SC_NGROUPS_MAX,   "NGROUPS_MAX",   SYSCONF,   NOFLAGS,
 195   _SC_OPEN_MAX,   "OPEN_MAX",   SYSCONF,   NOFLAGS,
 196   _SC_SAVED_IDS,   "_POSIX_SAVED_IDS",   SYSCONF,   NOFLAGS,
 197   _SC_STREAM_MAX,   "STREAM_MAX",   SYSCONF,   NOFLAGS,
 198   _SC_TZNAME_MAX,   "TZNAME_MAX",   SYSCONF,   NOFLAGS,
 199   _SC_VERSION,   "_POSIX_VERSION",   SYSCONF,   NOFLAGS,

 201   /* pathconf() variables */

 203   /* POSIX.1-1990 table 5-2 */

 205   _PC_LINK_MAX,   "LINK_MAX",   PATHCONF, NOFLAGS,
 206   _PC_MAX_CANON,   "MAX_CANON",   PATHCONF, NOFLAGS,
 207   _PC_MAX_INPUT,   "MAX_INPUT",   PATHCONF, NOFLAGS,
 208   _PC_NAME_MAX,   "NAME_MAX",   PATHCONF, NOFLAGS,
 209   _PC_PATH_MAX,   "PATH_MAX",   PATHCONF, NOFLAGS,
 210   _PC_PIPE_BUF,   "PIPE_BUF",   PATHCONF, NOFLAGS,
 211   _PC_CHOWN_RESTRICTED,   "_POSIX_CHOWN_RESTRICTED", PATHCONF, NOFLAGS,
 212   _PC_NO_TRUNC,   "_POSIX_NO_TRUNC",   PATHCONF, NOFLAGS,
 213   _PC_VDISABLE,   "_POSIX_VDISABLE",   PATHCONF, NOFLAGS,

```
215         /* UNIX 98 additions */

217         _PC_ASYNC_IO,           "_POSIX_ASYNC_IO",      PATHCONF,  NOFLAGS,
218         _PC_PRIO_IO,            "_POSIX_PRIO_IO",       PATHCONF,  NOFLAGS,
219         _PC_SYNC_IO,            "_POSIX_SYNC_IO",       PATHCONF,  NOFLAGS,

221         /* Large File Summit name (UNIX 98) */

223         _PC_FILESIZEBITS,       "FILESIZEBITS",         PATHCONF,  NOFLAGS,

225         /* UNIX 03 Additions */

227         /* Advisory Information (ADV) */
228         _PC_ALLOC_SIZE_MIN,     "POSIX_ALLOC_SIZE_MIN", PATHCONF,  NOFLAGS,
229         _PC_REC_INCR_XFER_SIZE, "POSIX_REC_INCR_XFER_SIZE", PATHCONF,  NOFLAGS,
230         _PC_REC_MAX_XFER_SIZE,  "POSIX_REC_MAX_XFER_SIZE",  PATHCONF,  NOFLAGS,
231         _PC_REC_MIN_XFER_SIZE,  "POSIX_REC_MIN_XFER_SIZE",  PATHCONF,  NOFLAGS,
232         _PC_REC_XFER_ALIGN,     "POSIX_REC_XFER_ALIGN", PATHCONF,  NOFLAGS,

234         _PC_2_SYMLINKS,         "POSIX2_SYMLINKS",      PATHCONF,  NOFLAGS,
235         _PC_SYMLINK_MAX,        "SYMLINK_MAX",          PATHCONF,  NOFLAGS,

237         /* UNIX 08 additions */

239         _PC_TIMESTAMP_RESOLUTION, "_POSIX_TIMESTAMP_RESOLUTION",
240             PATHCONF,   NOFLAGS,

242         /* confstr() variables */

244         /* POSIX.2-1992 description of getconf utility */
245         _CS_PATH,               "PATH",                 CONFSTR,   NOFLAGS,

247         /* POSIX.2-1992 table B-18 */
248         _CS_PATH,               "CS_PATH",              CONFSTR,   NOFLAGS,
249         _CS_PATH,               "_CS_PATH",             CONFSTR,   NOFLAGS,

251         /* command names for large file configuration information */
252         /* large file compilation environment configuration */

254         _CS_LFS_CFLAGS,         "LFS_CFLAGS",           CONFSTR,   NOFLAGS,
255         _CS_LFS_LDFLAGS,        "LFS_LDFLAGS",          CONFSTR,   NOFLAGS,
256         _CS_LFS_LIBS,           "LFS_LIBS",             CONFSTR,   NOFLAGS,
257         _CS_LFS_LINTFLAGS,      "LFS_LINTFLAGS",        CONFSTR,   NOFLAGS,

259         /* transitional large file interface configuration */

261         _CS_LFS64_CFLAGS,       "LFS64_CFLAGS",         CONFSTR,   NOFLAGS,
262         _CS_LFS64_LDFLAGS,      "LFS64_LDFLAGS",        CONFSTR,   NOFLAGS,
263         _CS_LFS64_LIBS,         "LFS64_LIBS",           CONFSTR,   NOFLAGS,
264         _CS_LFS64_LINTFLAGS,    "LFS64_LINTFLAGS",      CONFSTR,   NOFLAGS,

266         /*
267          *  UNIX 98:
268          *  confstr additions, these support
269          *  programming specifications/environments.
270          *
271          * Unfortunately there is some uncertainty with
272          * the standard at this point about the constant
273          * name, thus we need to recognize BOTH with and
274          * without the initial underscore.
275          */
276         _CS_XBS5_ILP32_OFF32_CFLAGS,    "XBS5_ILP32_OFF32_CFLAGS",
277                 CONFSTR,        NOFLAGS,
278         _CS_XBS5_ILP32_OFF32_CFLAGS,    "_XBS5_ILP32_OFF32_CFLAGS",
279                 CONFSTR,        NOFLAGS,
```

```
281         _CS_XBS5_ILP32_OFF32_LDFLAGS,   "XBS5_ILP32_OFF32_LDFLAGS",
282                 CONFSTR,        NOFLAGS,
283         _CS_XBS5_ILP32_OFF32_LDFLAGS,   "_XBS5_ILP32_OFF32_LDFLAGS",
284                 CONFSTR,        NOFLAGS,

286         _CS_XBS5_ILP32_OFF32_LIBS,      "XBS5_ILP32_OFF32_LIBS",
287                 CONFSTR,        NOFLAGS,
288         _CS_XBS5_ILP32_OFF32_LIBS,      "_XBS5_ILP32_OFF32_LIBS",
289                 CONFSTR,        NOFLAGS,

291         _CS_XBS5_ILP32_OFF32_LINTFLAGS, "XBS5_ILP32_OFF32_LINTFLAGS",
292                 CONFSTR,        NOFLAGS,
293         _CS_XBS5_ILP32_OFF32_LINTFLAGS, "_XBS5_ILP32_OFF32_LINTFLAGS",
294                 CONFSTR,        NOFLAGS,

296         _CS_XBS5_ILP32_OFFBIG_CFLAGS,   "XBS5_ILP32_OFFBIG_CFLAGS",
297                 CONFSTR,        NOFLAGS,
298         _CS_XBS5_ILP32_OFFBIG_CFLAGS,   "_XBS5_ILP32_OFFBIG_CFLAGS",
299                 CONFSTR,        NOFLAGS,

301         _CS_XBS5_ILP32_OFFBIG_LDFLAGS,  "XBS5_ILP32_OFFBIG_LDFLAGS",
302                 CONFSTR,        NOFLAGS,
303         _CS_XBS5_ILP32_OFFBIG_LDFLAGS,  "_XBS5_ILP32_OFFBIG_LDFLAGS",
304                 CONFSTR,        NOFLAGS,

306         _CS_XBS5_ILP32_OFFBIG_LIBS,     "XBS5_ILP32_OFFBIG_LIBS",
307                 CONFSTR,        NOFLAGS,
308         _CS_XBS5_ILP32_OFFBIG_LIBS,     "_XBS5_ILP32_OFFBIG_LIBS",
309                 CONFSTR,        NOFLAGS,

311         _CS_XBS5_ILP32_OFFBIG_LINTFLAGS, "XBS5_ILP32_OFFBIG_LINTFLAGS",
312                 CONFSTR,        NOFLAGS,
313         _CS_XBS5_ILP32_OFFBIG_LINTFLAGS, "_XBS5_ILP32_OFFBIG_LINTFLAGS",
314                 CONFSTR,        NOFLAGS,

316         _CS_XBS5_LP64_OFF64_CFLAGS,     "XBS5_LP64_OFF64_CFLAGS",
317                 CONFSTR,        NOFLAGS,
318         _CS_XBS5_LP64_OFF64_CFLAGS,     "_XBS5_LP64_OFF64_CFLAGS",
319                 CONFSTR,        NOFLAGS,

321         _CS_XBS5_LP64_OFF64_LDFLAGS,    "XBS5_LP64_OFF64_LDFLAGS",
322                 CONFSTR,        NOFLAGS,
323         _CS_XBS5_LP64_OFF64_LDFLAGS,    "_XBS5_LP64_OFF64_LDFLAGS",
324                 CONFSTR,        NOFLAGS,

326         _CS_XBS5_LP64_OFF64_LIBS,       "XBS5_LP64_OFF64_LIBS",
327                 CONFSTR,        NOFLAGS,
328         _CS_XBS5_LP64_OFF64_LIBS,       "_XBS5_LP64_OFF64_LIBS",
329                 CONFSTR,        NOFLAGS,

331         _CS_XBS5_LP64_OFF64_LINTFLAGS,  "XBS5_LP64_OFF64_LINTFLAGS",
332                 CONFSTR,        NOFLAGS,
333         _CS_XBS5_LP64_OFF64_LINTFLAGS,  "_XBS5_LP64_OFF64_LINTFLAGS",
334                 CONFSTR,        NOFLAGS,

336         _CS_XBS5_LPBIG_OFFBIG_CFLAGS,   "XBS5_LPBIG_OFFBIG_CFLAGS",
337                 CONFSTR,        NOFLAGS,
338         _CS_XBS5_LPBIG_OFFBIG_CFLAGS,   "_XBS5_LPBIG_OFFBIG_CFLAGS",
339                 CONFSTR,        NOFLAGS,

341         _CS_XBS5_LPBIG_OFFBIG_LDFLAGS,  "XBS5_LPBIG_OFFBIG_LDFLAGS",
342                 CONFSTR,        NOFLAGS,
343         _CS_XBS5_LPBIG_OFFBIG_LDFLAGS,  "_XBS5_LPBIG_OFFBIG_LDFLAGS",
344                 CONFSTR,        NOFLAGS,
```

```
346          _CS_XBS5_LPBIG_OFFBIG_LIBS,      "XBS5_LPBIG_OFFBIG_LIBS",
347                  CONFSTR,          NOFLAGS,
348          _CS_XBS5_LPBIG_OFFBIG_LIBS,      "_XBS5_LPBIG_OFFBIG_LIBS",
349                  CONFSTR,          NOFLAGS,

351          _CS_XBS5_LPBIG_OFFBIG_LINTFLAGS, "XBS5_LPBIG_OFFBIG_LINTFLAGS",
352                  CONFSTR,          NOFLAGS,
353          _CS_XBS5_LPBIG_OFFBIG_LINTFLAGS, "_XBS5_LPBIG_OFFBIG_LINTFLAGS",
354                  CONFSTR,          NOFLAGS,

356          /*
357           * UNIX 03 confstr() additions for compilation environments
358           */
359          _CS_POSIX_V6_ILP32_OFF32_CFLAGS, "POSIX_V6_ILP32_OFF32_CFLAGS",
360                  CONFSTR,          NOFLAGS,
361          _CS_POSIX_V6_ILP32_OFF32_LDFLAGS, "POSIX_V6_ILP32_OFF32_LDFLAGS",
362                  CONFSTR,          NOFLAGS,
363          _CS_POSIX_V6_ILP32_OFF32_LIBS, "POSIX_V6_ILP32_OFF32_LIBS",
364                  CONFSTR,          NOFLAGS,
365          _CS_POSIX_V6_ILP32_OFFBIG_CFLAGS, "POSIX_V6_ILP32_OFFBIG_CFLAGS",
366                  CONFSTR,          NOFLAGS,
367          _CS_POSIX_V6_ILP32_OFFBIG_LDFLAGS, "POSIX_V6_ILP32_OFFBIG_LDFLAGS",
368                  CONFSTR,          NOFLAGS,
369          _CS_POSIX_V6_ILP32_OFFBIG_LIBS, "POSIX_V6_ILP32_OFFBIG_LIBS",
370                  CONFSTR,          NOFLAGS,
371          _CS_POSIX_V6_LP64_OFF64_CFLAGS, "POSIX_V6_LP64_OFF64_CFLAGS",
372                  CONFSTR,          NOFLAGS,
373          _CS_POSIX_V6_LP64_OFF64_LDFLAGS, "POSIX_V6_LP64_OFF64_LDFLAGS",
374                  CONFSTR,          NOFLAGS,
375          _CS_POSIX_V6_LP64_OFF64_LIBS, "POSIX_V6_LP64_OFF64_LIBS",
376                  CONFSTR,          NOFLAGS,
377          _CS_POSIX_V6_LPBIG_OFFBIG_CFLAGS, "POSIX_V6_LPBIG_OFFBIG_CFLAGS",
378                  CONFSTR,          NOFLAGS,
379          _CS_POSIX_V6_LPBIG_OFFBIG_LDFLAGS, "POSIX_V6_LPBIG_OFFBIG_LDFLAGS",
380                  CONFSTR,          NOFLAGS,
381          _CS_POSIX_V6_LPBIG_OFFBIG_LIBS, "POSIX_V6_LPBIG_OFFBIG_LIBS",
382                  CONFSTR,          NOFLAGS,
383          _CS_POSIX_V6_WIDTH_RESTRICTED_ENVS, "POSIX_V6_WIDTH_RESTRICTED_ENVS",
384                  CONFSTR,          NOFLAGS,
385          /* End of UNIX 03 compilation environments */

387 #ifdef _SC_2_C_VERSION          /* POSIX.2 table B-4 */
388          _SC_2_C_VERSION,          "_POSIX2_C_VERSION",     SYSCONF,   NOFLAGS,
389 #else
390          _POSIX2_C_VERSION,        "_POSIX2_C_VERSION",     SELFCONF,  NOFLAGS,
391 #endif

393 /*
394  * XPG4 support BEGINS
395  */

397 #ifndef CHARCLASS_NAME_MAX
398 #define CHARCLASS_NAME_MAX      -1
399 #endif
400 #ifndef CHAR_BIT
401 #define CHAR_BIT         -1
402 #endif
403 #ifndef CHAR_MAX
404 #define CHAR_MAX         -1
405 #endif
406 #ifndef CHAR_MIN
407 #define CHAR_MIN         -1
408 #endif
409 #ifndef INT_MAX
410 #define INT_MAX -1
411 #endif
```

```
412 #ifndef INT_MIN
413 #define INT_MIN -1
414 #endif
415 #ifndef LONG_BIT
416 #define LONG_BIT         -1
417 #endif
418 #ifndef LONG_MAX
419 #define LONG_MAX         -1
420 #endif
421 #ifndef LONG_MIN
422 #define LONG_MIN         -1
423 #endif
424 #ifndef MB_LEN_MAX
425 #define MB_LEN_MAX       -1
426 #endif
427 #ifndef NL_NMAX
428 #define NL_NMAX -1
429 #endif
430 #ifndef NL_ARGMAX
431 #define NL_ARGMAX        -1
432 #endif
433 #ifndef NL_LANGMAX
434 #define NL_LANGMAX       -1
435 #endif
436 #ifndef NL_MSGMAX
437 #define NL_MSGMAX        -1
438 #endif
439 #ifndef NL_SETMAX
440 #define NL_SETMAX        -1
441 #endif
442 #ifndef NL_TEXTMAX
443 #define NL_TEXTMAX       -1
444 #endif
445 #ifndef NZERO
446 #define NZERO    -1
447 #endif
448 #ifndef SCHAR_MAX
449 #define SCHAR_MAX        -1
450 #endif
451 #ifndef SCHAR_MIN
452 #define SCHAR_MIN        -1
453 #endif
454 #ifndef SHRT_MAX
455 #define SHRT_MAX         -1
456 #endif
457 #ifndef SHRT_MIN
458 #define SHRT_MIN         -1
459 #endif
460 #ifndef TMP_MAX
461 #define TMP_MAX -1
462 #endif
463 #ifndef WORD_BIT
464 #define WORD_BIT         -1
465 #endif


468 #ifndef _XOPEN_XPG2
469 #define _XOPEN_XPG2      -1
470 #endif
471 #ifndef _XOPEN_XPG3
472 #define _XOPEN_XPG3      -1
473 #endif
474 #ifndef _XOPEN_XPG4
475 #define _XOPEN_XPG4      -1
476 #endif
```

```
478          /*
479           * the following are values that we should find in <limits.h>
480           * so we use SELFCONF here.
481           *
482           */

484          /* POSIX.1-1990 */
485          CHARCLASS_NAME_MAX,     "CHARCLASS_NAME_MAX",     SELFCONF,   NOFLAGS,
486          SSIZE_MAX,              "SSIZE_MAX",     SELFCONF,   NOFLAGS,

488          /* POSIX.1-1988 */
489          CHAR_BIT,               "CHAR_BIT",      SELFCONF,   NOFLAGS,
490          CHAR_MAX,               "CHAR_MAX",      SELFCONF,   NOFLAGS,
491          CHAR_MIN,               "CHAR_MIN",      SELFCONF,   NOFLAGS,
492          INT_MAX,                "INT_MAX",       SELFCONF,   NOFLAGS,
493          INT_MIN,                "INT_MIN",       SELFCONF,   NOFLAGS,
494          LONG_BIT,               "LONG_BIT",      SELFCONF,   NOFLAGS,
495          LONG_MAX,               "LONG_MAX",      SELFCONF,   NOFLAGS,
496          LONG_MIN,               "LONG_MIN",      SELFCONF,   NOFLAGS,
497          MB_LEN_MAX,             "MB_LEN_MAX",    SELFCONF,   NOFLAGS,
498          NL_NMAX,                "NL_NMAX",       SELFCONF,   NOFLAGS,
499          NL_ARGMAX,              "NL_ARGMAX",     SELFCONF,   NOFLAGS,
500          NL_LANGMAX,             "NL_LANGMAX",    SELFCONF,   NOFLAGS,
501          NL_MSGMAX,              "NL_MSGMAX",     SELFCONF,   NOFLAGS,
502          NL_SETMAX,              "NL_SETMAX",     SELFCONF,   NOFLAGS,
503          NL_TEXTMAX,             "NL_TEXTMAX",    SELFCONF,   NOFLAGS,
504          NZERO,                  "NZERO",         SELFCONF,   NOFLAGS,
505          SCHAR_MAX,              "SCHAR_MAX",     SELFCONF,   NOFLAGS,
506          SCHAR_MIN,              "SCHAR_MIN",     SELFCONF,   NOFLAGS,
507          SHRT_MAX,               "SHRT_MAX",      SELFCONF,   NOFLAGS,
508          SHRT_MIN,               "SHRT_MIN",      SELFCONF,   NOFLAGS,
509          TMP_MAX,                "TMP_MAX",       SELFCONF,   NOFLAGS,

511          /*
512           * for the unsigned maximums, we cannot rely on the value -1
513           * to indicate "undefined".
514           */
515 #ifndef UCHAR_MAX
516          0,                      "UCHAR_MAX",     SELFCONF,   UNDEFINED,
517 #else
518          (long)UCHAR_MAX,        "UCHAR_MAX",     SELFCONF,   UNSIGNED_VALUE,
519 #endif /* UCHAR_MAX */
520 #ifndef UINT_MAX
521          0,                      "UINT_MAX",      SELFCONF,   UNDEFINED,
522 #else
523          (long)UINT_MAX,         "UINT_MAX",      SELFCONF,   UNSIGNED_VALUE,
524 #endif /* UINT_MAX */
525 #ifndef ULONG_MAX
526          0,                      "ULONG_MAX",     SELFCONF,   UNDEFINED,
527 #else
528          (long)ULONG_MAX,        "ULONG_MAX",     SELFCONF,   UNSIGNED_VALUE,
529 #endif /* ULONG_MAX */
530 #ifndef USHRT_MAX
531          0,                      "USHRT_MAX",     SELFCONF,   UNDEFINED,
532 #else
533          (long)USHRT_MAX,        "USHRT_MAX",     SELFCONF,   UNSIGNED_VALUE,
534 #endif /* USHRT_MAX */
535 #ifndef SIZE_MAX
536          0,                      "SIZE_MAX",      SELFCONF,   UNDEFINED,
537 #else
538          (long)SIZE_MAX,         "SIZE_MAX",      SELFCONF,   UNSIGNED_VALUE,
539 #endif /* SIZE_MAX */
540          WORD_BIT,               "WORD_BIT",      SELFCONF,   NOFLAGS,

543          /* XPG4 Version 2 */
```

```
544          _XOPEN_IOV_MAX,         "_XOPEN_IOV_MAX",  SELFCONF, NOFLAGS,

546          /* UNIX 03 */
547          _XOPEN_NAME_MAX,        "_XOPEN_NAME_MAX", SELFCONF, NOFLAGS,
548          _XOPEN_PATH_MAX,        "_XOPEN_PATH_MAX", SELFCONF, NOFLAGS,

550          /*
551           * The following are defined via sysconf().  These are considered
552           * an extension to sysconf().
553           * The -0 is needed if _XOPEN_XPG? is defined to be an empty string.
554           */
555          _XOPEN_XPG2-0,          "_XOPEN_XPG2",  SELFCONF,   NOFLAGS,
556          _XOPEN_XPG3-0,          "_XOPEN_XPG3",  SELFCONF,   NOFLAGS,
557          _XOPEN_XPG4-0,          "_XOPEN_XPG4",  SELFCONF,   NOFLAGS,

559          /*
560           * The following should be provided by sysconf() (e.g use SYSCONF),
561           * so we  look for the appropriate _SC_* symbol in <unistd.h>.
562           * If it is not defined, then we use SELFCONF with the value of -1.
563           */
564          _SC_XOPEN_VERSION,      "_XOPEN_VERSION",      SYSCONF,  NOFLAGS,
565          _SC_XOPEN_XCU_VERSION,  "_XOPEN_XCU_VERSION",  SYSCONF,  NOFLAGS,
566          _SC_XOPEN_CRYPT,        "_XOPEN_CRYPT",        SYSCONF,  NOFLAGS,
567          _SC_XOPEN_ENH_I18N,     "_XOPEN_ENH_I18N",     SYSCONF,  NOFLAGS,
568          _SC_XOPEN_SHM,          "_XOPEN_SHM",          SYSCONF,  NOFLAGS,

570 /*
571  * XPG4 support ends
572  */

574 /* XPG4, Version 2 support */

576          _SC_ATEXIT_MAX,         "ATEXIT_MAX",          SYSCONF,      NOFLAGS,
577          _SC_PAGESIZE,           "PAGESIZE",            SYSCONF,      NOFLAGS,
578          _SC_PAGE_SIZE,          "PAGE_SIZE",           SYSCONF,      NOFLAGS,
579          _SC_XOPEN_UNIX,         "_XOPEN_UNIX",         SYSCONF,      NOFLAGS,
580          _SC_IOV_MAX,            "IOV_MAX",             SYSCONF,      NOFLAGS,

582 /*
583  * UNIX 98 - XPG5 additions
584  */
585          _SC_XOPEN_LEGACY,       "_XOPEN_LEGACY",       SYSCONF,      NOFLAGS,
586          _SC_PASS_MAX,           "PASS_MAX",            SYSCONF,      NOFLAGS,
587          _SC_AIO_LISTIO_MAX,     "AIO_LISTIO_MAX",      SYSCONF,      NOFLAGS,
588          _SC_AIO_MAX,            "AIO_MAX",             SYSCONF,      NOFLAGS,
589          _SC_AIO_PRIO_DELTA_MAX, "AIO_PRIO_DELTA_MAX",  SYSCONF,      NOFLAGS,
590          _SC_DELAYTIMER_MAX,     "DELAYTIMER_MAX",      SYSCONF,      NOFLAGS,
591          _SC_MQ_OPEN_MAX,        "MQ_OPEN_MAX",         SYSCONF,      NOFLAGS,
592          _SC_MQ_PRIO_MAX,        "MQ_PRIO_MAX",         SYSCONF,      NOFLAGS,
593          _SC_RTSIG_MAX,          "RTSIG_MAX",           SYSCONF,      NOFLAGS,
594          _SC_SEM_NSEMS_MAX,      "SEM_NSEMS_MAX",       SYSCONF,      NOFLAGS,
595          _SC_SEM_VALUE_MAX,      "SEM_VALUE_MAX",       SYSCONF,      NOFLAGS,
596          _SC_SIGQUEUE_MAX,       "SIGQUEUE_MAX",        SYSCONF,      NOFLAGS,
597          _SC_TIMER_MAX,          "TIMER_MAX",           SYSCONF,      NOFLAGS,

599          _SC_ASYNCHRONOUS_IO,    "_POSIX_ASYNCHRONOUS_IO", SYSCONF,      NOFLAGS,
600          _SC_FSYNC,              "_POSIX_FSYNC",        SYSCONF,      NOFLAGS,
601          _SC_MAPPED_FILES,       "_POSIX_MAPPED_FILES", SYSCONF,      NOFLAGS,
602          _SC_MEMLOCK,            "_POSIX_MEMLOCK",      SYSCONF,      NOFLAGS,
603          _SC_MEMLOCK_RANGE,      "_POSIX_MEMLOCK_RANGE", SYSCONF,      NOFLAGS,
604          _SC_MEMORY_PROTECTION,  "_POSIX_MEMORY_PROTECTION", SYSCONF,      NOFLAGS,
605          _SC_MESSAGE_PASSING,    "_POSIX_MESSAGE_PASSING", SYSCONF,      NOFLAGS,
606          _SC_PRIORITIZED_IO,     "_POSIX_PRIORITIZED_IO", SYSCONF,      NOFLAGS,
607          _SC_PRIORITY_SCHEDULING, "_POSIX_PRIORITY_SCHEDULING", SYSCONF, NOFLAGS,
608          _SC_REALTIME_SIGNALS,   "_POSIX_REALTIME_SIGNALS", SYSCONF,      NOFLAGS,
609          _SC_SEMAPHORES,         "_POSIX_SEMAPHORES",   SYSCONF,      NOFLAGS,
```

```
  611          _SC_SHARED_MEMORY_OBJECTS, "_POSIX_SHARED_MEMORY_OBJECTS",
  612              SYSCONF,    NOFLAGS,

  614          _SC_SYNCHRONIZED_IO,    "_POSIX_SYNCHRONIZED_IO", SYSCONF,      NOFLAGS,
  615          _SC_TIMERS,             "_POSIX_TIMERS",        SYSCONF,        NOFLAGS,

  617          _SC_LOGIN_NAME_MAX,     "LOGIN_NAME_MAX",       SYSCONF,        NOFLAGS,

  619          _SC_THREAD_DESTRUCTOR_ITERATIONS, "PTHREAD_DESTRUCTOR_ITERATIONS",
  620              SYSCONF,    NOFLAGS,

  622          _SC_THREAD_KEYS_MAX,    "PTHREAD_KEYS_MAX",     SYSCONF,        NOFLAGS,
  623          _SC_THREAD_STACK_MIN,   "PTHREAD_STACK_MIN",    SYSCONF,        NOFLAGS,
  624          _SC_THREAD_THREADS_MAX, "PTHREAD_THREADS_MAX",  SYSCONF,        NOFLAGS,

  626          _SC_THREADS,            "_POSIX_THREADS",       SYSCONF,        NOFLAGS,

  628          _SC_THREAD_ATTR_STACKADDR, "POSIX_THREAD_ATTR_STACKADDR",
  629              SYSCONF,    NOFLAGS,
  630          _SC_THREAD_ATTR_STACKADDR, "_POSIX_THREAD_ATTR_STACKADDR",
  631              SYSCONF,    NOFLAGS,

  633          _SC_THREAD_ATTR_STACKSIZE, "POSIX_THREAD_ATTR_STACKSIZE",
  634              SYSCONF,    NOFLAGS,
  635          _SC_THREAD_ATTR_STACKSIZE, "_POSIX_THREAD_ATTR_STACKSIZE",
  636              SYSCONF,    NOFLAGS,

  638          _SC_THREAD_PRIORITY_SCHEDULING, "POSIX_THREAD_PRIORITY_SCHEDULING",
  639              SYSCONF,    NOFLAGS,
  640          _SC_THREAD_PRIORITY_SCHEDULING, "_POSIX_THREAD_PRIORITY_SCHEDULING",
  641              SYSCONF,    NOFLAGS,

  643          _SC_THREAD_PRIO_INHERIT, "POSIX_THREAD_PRIO_INHERIT",
  644              SYSCONF,    NOFLAGS,
  645          _SC_THREAD_PRIO_INHERIT, "_POSIX_THREAD_PRIO_INHERIT",
  646              SYSCONF,    NOFLAGS,

  648          _SC_THREAD_PRIO_PROTECT, "POSIX_THREAD_PRIO_PROTECT",
  649              SYSCONF,    NOFLAGS,
  650          _SC_THREAD_PRIO_PROTECT, "_POSIX_THREAD_PRIO_PROTECT",
  651              SYSCONF,    NOFLAGS,

  653          _SC_THREAD_PROCESS_SHARED, "POSIX_THREAD_PROCESS_SHARED",
  654              SYSCONF,    NOFLAGS,
  655          _SC_THREAD_PROCESS_SHARED, "_POSIX_THREAD_PROCESS_SHARED",
  656              SYSCONF,    NOFLAGS,

  658          _SC_THREAD_SAFE_FUNCTIONS, "POSIX_THREAD_SAFE_FUNCTIONS",
  659              SYSCONF,    NOFLAGS,
  660          _SC_THREAD_SAFE_FUNCTIONS, "_POSIX_THREAD_SAFE_FUNCTIONS",
  661              SYSCONF,    NOFLAGS,

  663          _SC_TTY_NAME_MAX,       "TTY_NAME_MAX",         SYSCONF,        NOFLAGS,
  664          _SC_XOPEN_REALTIME,     "_XOPEN_REALTIME",      SYSCONF,        NOFLAGS,
  665          _SC_XOPEN_REALTIME_THREADS, "_XOPEN_REALTIME_THREADS", SYSCONF, NOFLAGS,

  667          /*
  668           * Due to ambiguity in the standard we need
  669           * to recognize both the underscored name
  670           * and the non-underscored version in the
  671           * following variables.
  672           */
  673          _SC_XBS5_ILP32_OFF32,   "XBS5_ILP32_OFF32",     SYSCONF,        NOFLAGS,
  674          _SC_XBS5_ILP32_OFF32,   "_XBS5_ILP32_OFF32",    SYSCONF,        NOFLAGS,
```

```
  676          _SC_XBS5_ILP32_OFFBIG,  "XBS5_ILP32_OFFBIG",    SYSCONF,        NOFLAGS,
  677          _SC_XBS5_ILP32_OFFBIG,  "_XBS5_ILP32_OFFBIG",   SYSCONF,        NOFLAGS,

  679          _SC_XBS5_LP64_OFF64,    "XBS5_LP64_OFF64",      SYSCONF,        NOFLAGS,
  680          _SC_XBS5_LP64_OFF64,    "_XBS5_LP64_OFF64",     SYSCONF,        NOFLAGS,

  682          _SC_XBS5_LPBIG_OFFBIG,  "XBS5_LPBIG_OFFBIG",    SYSCONF,        NOFLAGS,
  683          _SC_XBS5_LPBIG_OFFBIG,  "_XBS5_LPBIG_OFFBIG",   SYSCONF,        NOFLAGS,

  685 /*
  686  * UNIX 03 additions for sysconf() names, and names that are not in interfaces
  687  */
  688          _SC_REGEXP,             "_POSIX_REGEXP",        SYSCONF,        NOFLAGS,
  689          _SC_SHELL,              "_POSIX_SHELL",         SYSCONF,        NOFLAGS,
  690          _SC_HOST_NAME_MAX,      "HOST_NAME_MAX",        SYSCONF,        NOFLAGS,
  691          _SC_READER_WRITER_LOCKS, "_POSIX_READER_WRITER_LOCKS", SYSCONF, NOFLAGS,
  692          _SC_IPV6,               "_POSIX_IPV6",          SYSCONF,        NOFLAGS,
  693          _SC_RAW_SOCKETS,        "_POSIX_RAW_SOCKETS",   SYSCONF,        NOFLAGS,
  694          _SC_XOPEN_STREAMS,      "_XOPEN_STREAMS",       SYSCONF,        NOFLAGS,
  695          _SC_SYMLOOP_MAX,        "SYMLOOP_MAX",          SYSCONF,        NOFLAGS,
  696          _SC_2_PBS,              "_POSIX2_PBS",          SYSCONF,        NOFLAGS,
  697          _SC_2_PBS_ACCOUNTING,   "_POSIX2_PBS_ACCOUNTING", SYSCONF,      NOFLAGS,
  698          _SC_2_PBS_CHECKPOINT,   "_POSIX2_PBS_CHECKPOINT", SYSCONF,      NOFLAGS,
  699          _SC_2_PBS_LOCATE,       "_POSIX2_PBS_LOCATE",   SYSCONF,        NOFLAGS,
  700          _SC_2_PBS_MESSAGE,      "_POSIX2_PBS_MESSAGE",  SYSCONF,        NOFLAGS,
  701          _SC_2_PBS_TRACK,        "_POSIX2_PBS_TRACK",    SYSCONF,        NOFLAGS,

  703          _SC_ADVISORY_INFO,      "_POSIX_ADVISORY_INFO", SYSCONF,        NOFLAGS,
  704          _SC_BARRIERS,           "_POSIX_BARRIERS",      SYSCONF,        NOFLAGS,
  705          _SC_CLOCK_SELECTION,    "_POSIX_CLOCK_SELECTION", SYSCONF,      NOFLAGS,
  706          _SC_CPUTIME,            "_POSIX_CPUTIME",       SYSCONF,        NOFLAGS,
  707          _SC_MONOTONIC_CLOCK,    "_POSIX_MONOTONIC_CLOCK", SYSCONF,      NOFLAGS,
  708          _SC_SPAWN,              "_POSIX_SPAWN",         SYSCONF,        NOFLAGS,
  709          _SC_SPIN_LOCKS,         "_POSIX_SPIN_LOCKS",    SYSCONF,        NOFLAGS,
  710          _SC_SPORADIC_SERVER,    "_POSIX_SPORADIC_SERVER", SYSCONF,      NOFLAGS,
  711          _SC_SS_REPL_MAX,        "_POSIX_SS_REPL_MAX",   SYSCONF,        NOFLAGS,
  712          _SC_THREAD_CPUTIME,     "_POSIX_THREAD_CPUTIME", SYSCONF,       NOFLAGS,
  713          _SC_THREAD_SPORADIC_SERVER, "_POSIX_THREAD_SPORADIC_SERVER",    SYSCONF,
  714              NOFLAGS,
  715          _SC_TIMEOUTS,           "_POSIX_TIMEOUTS",      SYSCONF,        NOFLAGS,
  716          _SC_TRACE,              "_POSIX_TRACE",         SYSCONF,        NOFLAGS,
  717          _SC_TRACE_EVENT_FILTER, "_POSIX_TRACE_EVENT_FILTER", SYSCONF,   NOFLAGS,
  718          _SC_TRACE_EVENT_NAME_MAX, "_POSIX_TRACE_EVENT_NAME_MAX", SYSCONF,
  719                                                                          NOFLAGS,
  720          _SC_TRACE_INHERIT,      "_POSIX_TRACE_INHERIT", SYSCONF,        NOFLAGS,
  721          _SC_TRACE_LOG,          "_POSIX_TRACE_LOG",     SYSCONF,        NOFLAGS,
  722          _SC_TRACE_NAME_MAX,     "_POSIX_TRACE_NAME_MAX", SYSCONF,       NOFLAGS,
  723          _SC_TRACE_SYS_MAX,      "_POSIX_TRACE_SYS_MAX", SYSCONF,        NOFLAGS,
  724          _SC_TRACE_USER_EVENT_MAX, "_POSIX_TRACE_USER_EVENT_MAX", SYSCONF,
  725                                                                          NOFLAGS,
  726          _SC_TYPED_MEMORY_OBJECTS, "_POSIX_TYPED_MEMORY_OBJECTS",        SYSCONF,
  727              NOFLAGS,

  729          /*
  730           * Due to an ambiguity in the specification, we need to recognize both
  731           * the underscored POSIX name and the non-underscored name. In addition,
  732           * we need to permit the use of the _V6_* environment names specified
  733           * in unistd.h.
  734           */
  735          _SC_V6_ILP32_OFF32,     "_V6_ILP32_OFF32",      SYSCONF,        NOFLAGS,
  736          _SC_V6_ILP32_OFF32,     "_POSIX_V6_ILP32_OFF32", SYSCONF,       NOFLAGS,
  737          _SC_V6_ILP32_OFF32,     "POSIX_V6_ILP32_OFF32", SYSCONF,        NOFLAGS,

  739          _SC_V6_ILP32_OFFBIG,    "_V6_ILP32_OFFBIG",     SYSCONF,        NOFLAGS,
  740          _SC_V6_ILP32_OFFBIG,    "_POSIX_V6_ILP32_OFFBIG", SYSCONF,      NOFLAGS,
  741          _SC_V6_ILP32_OFFBIG,    "POSIX_V6_ILP32_OFFBIG", SYSCONF,       NOFLAGS,
```

```
 743          _SC_V6_LP64_OFF64,          "_V6_LP64_OFF64",         SYSCONF,        NOFLAGS,
 744          _SC_V6_LP64_OFF64,          "_POSIX_V6_LP64_OFF64", SYSCONF,          NOFLAGS,
 745          _SC_V6_LP64_OFF64,          "POSIX_V6_LP64_OFF64",  SYSCONF,          NOFLAGS,

 747          _SC_V6_LPBIG_OFFBIG,        "_V6_LPBIG_OFFBIG",       SYSCONF,        NOFLAGS,
 748          _SC_V6_LPBIG_OFFBIG,        "_POSIX_V6_LPBIG_OFFBIG", SYSCONF,        NOFLAGS,
 749          _SC_V6_LPBIG_OFFBIG,        "POSIX_V6_LPBIG_OFFBIG", SYSCONF,         NOFLAGS,
 750  /*
 751   * System V Interface Definition
 752   */
 753          _SC_NPROCESSORS_CONF,       "NPROCESSORS_CONF",       SYSCONF,        NOFLAGS,
 754          _SC_NPROCESSORS_MAX,        "NPROCESSORS_MAX",        SYSCONF,        NOFLAGS,
 755          _SC_NPROCESSORS_ONLN,       "NPROCESSORS_ONLN",       SYSCONF,        NOFLAGS,
 756  #endif /* ! codereview */
 757          0,                  NULL,   0,      0               /* end of table */
 758  };


 761  /*
 762   * Print usage message.
 763   */
 764  static int
 765  usage()
 766  {

 768          (void) fputs(gettext(
 769              "usage:\tgetconf [-v specification] system_var\n"
 770              "\tgetconf [-v specification] path_var pathname\n"
 771              "\tgetconf -a\n"), stderr);
 772          return (2);
 773  }


 776  static int
 777  namecmp(const void *a, const void *b)
 778  {
 779          return (strcoll(((const struct sctab *)a)->name,
 780              ((const struct sctab *)b)->name));
 781  }


 784  static int
 785  getconf(struct sctab *scp, int argc, char *name, char *file)
 786  {
 787          register size_t len;
 788          register char *buffer;
 789          long value;

 791          switch (scp->type) {
 792          case SELFCONF:
 793                  if (argc > 2)
 794                          return (usage());
 795                  if (scp->flag & UNDEFINED ||
 796                      (!(scp->flag&UNSIGNED_VALUE) && scp->value == -1)) {
 797                          /* DO NOT TRANSLATE */
 798                          (void) printf("undefined\n");
 799                  } else
 800                          if (scp->flag & UNSIGNED_VALUE)
 801                                  (void) printf("%lu\n", scp->value);
 802                          else
 803                                  (void) printf("%ld\n", scp->value);
 804                  break;

 806          case SYSCONF:
 807                  if (argc > 2)
```

```
 808                          return (usage());
 809                  errno = 0;
 810                  if (scp->flag & UNDEFINED ||
 811                      (value = sysconf((int)scp->value)) == -1) {
 812                          if (errno == EINVAL) {
 813                                  (void) fprintf(stderr,
 814                                      gettext(INVAL_ARG), name);
 815                                  return (1);
 816                          } else {
 817                                  /* DO NOT TRANSLATE */
 818                                  (void) printf("undefined\n");
 819                                  return (0);
 820                          }
 821                  } else
 822                          (void) printf("%ld\n", value);
 823                  break;

 825          case CONFSTR:
 826                  if (argc > 2)
 827                          return (usage());
 828                  errno = 0;
 829                  len = confstr((int)scp->value, NULL, (size_t)0);
 830                  if (len == 0)
 831                          if (errno == EINVAL) {
 832                                  (void) fprintf(stderr, gettext(INVAL_ARG),
 833                                      name);
 834                                  return (1);
 835                          } else {
 836                                  /* DO NOT TRANSLATE */
 837                                  (void) printf("undefined\n");
 838                                  return (0);
 839                          }
 840                  /*
 841                   * allocate space to store result of constr() into
 842                   */
 843                  if ((buffer = malloc(len)) == NULL) {
 844                          (void) fprintf(stderr,
 845                              gettext("insufficient memory available"));
 846                          return (1);
 847                  }

 849                  assert(confstr((int)scp->value, buffer, len) != 0);

 851                  (void) printf("%s\n", buffer);
 852                  free(buffer);
 853                  break;

 855          case PATHCONF:
 856                  if (argc != 3)
 857                          return (usage());
 858                  errno = 0;
 859                  if ((value = pathconf(file, (int)scp->value)) == -1) {
 860                          if (errno == EINVAL || errno == 0) {
 861                                  /* Does not support the association */
 862                                  /* DO NOT TRANSLATE */
 863                                  (void) printf("undefined\n");
 864                                  return (0);
 865                          } else {
 866                                  /* Probably problems with the pathname itself */
 867                                  if (aflag) {
 868                                          (void) printf("%s\n", file);
 869                                  } else {
 870                                          (void) printf("%s", file);
 871                                  }
 872                                  return (1);
 873                          }
```

```
874                   }
875                   (void) printf("%ld\n", value);
876                   break;
877           }
878           return (0);
879 }

881 int
882 main(int argc, char **argv)
883 {
884           register struct sctab *scp;
885           int c;
886           int exstat = 0;

888           (void) setlocale(LC_ALL, "");
889 #if      !defined(TEXT_DOMAIN)   /* Should be defined by cc -D */
890 #define TEXT_DOMAIN "SYS_TEST"
891 #endif
892           (void) textdomain(TEXT_DOMAIN);

894           while ((c = getopt(argc, argv, "av:")) != -1)
895                   switch (c) {
896                   case 'a':
897                           aflag = 1;
898                           break;
899                   case 'v':
900                           /*
901                            * Unix98 adds the -v option to allow
902                            * programming 'specifications' to be
903                            * indicated, for present purposes
904                            * the specification isn't really
905                            * doing anything of value, so for
906                            * the moment getopt just processes the
907                            * option value and argv[1] is adjusted.
908                            *
909                            * At some later date we might want to
910                            * do specification processing at this
911                            * point.
912                            */
913                           break;
914                   case '?':
915                           return (usage());
916                   }
917           argc -= optind-1;
918           argv += optind-1;

920           if ((aflag && argc >= 2) || (!aflag && argc < 2))
921                   return (usage());
922           if (aflag) {

924 #define TabStop        8
925 #define RightColumn    32
926 #define DefPathName    "."

928                   /*
929                    * sort the table by the "name" field
930                    * so we print it in sorted order
931                    */
932           qsort(&sctab[0], (sizeof (sctab)/sizeof (struct sctab))-1,
933               sizeof (struct sctab), namecmp);

935                   /*
936                    * print all the known symbols and their values
937                    */
938                   for (scp = &sctab[0]; scp->name != NULL; ++scp) {
939                           int stat;
```

```
941                           /*
942                            * create 2 columns:
943                            *    variable name in the left column,
944                            *    value in the right column.
945                            * The right column starts at a tab stop.
946                            */
947                           (void) printf("%s:\t", scp->name);

949                           c = strlen(scp->name) + 1;
950                           c = (c+TabStop) / TabStop, c *= TabStop;
951                           for (; c < RightColumn; c += TabStop)
952                                   (void) putchar('\t');

954                           /*
955                            * for pathconf() related variables that require
956                            * a pathname, use "."
957                            */
958                           stat = getconf(scp, scp->type == PATHCONF ? 3 : 2,
959                               scp->name, DefPathName);

961                           exstat |= stat;

963                           /*
964                            * if stat != 0, then an error message was already
965                            * printed in getconf(),
966                            * so don't need to print one here
967                            */
968                   }
969           } else {

971                   /*
972                    * find the name of the specified variable (argv[1])
973                    * and print its value.
974                    */
975                   for (scp = &sctab[0]; scp->name != NULL; ++scp)
976                           if (strcmp(argv[1], scp->name) == 0) {
977                                   exstat = getconf(scp, argc, argv[1], argv[2]);
978                                   break;
979                           }

981                   /*
982                    * if at last entry in table, then the user specified
983                    * variable is invalid
984                    */
985                   if (scp->name == NULL) {
986                           errno = EINVAL;
987                           (void) fprintf(stderr, gettext(INVAL_ARG), argv[1]);
988                           return (1);
989                   }
990           }
991           return (exstat);
992 }
```