type wchar_t * that wcstok modifies to keep track of its progress. If s1 is a null pointer, this object must be the same one used in a previous call of wcstok; it determines which wide string is to be searched and where the search is to begin.

*Returns*     A pointer to the first wide character of the token. Returns a null pointer if no token could be found.                                                                                    *25.5*

---

**wcstol**     *Convert Wide String to Long Integer (C99)*          <wchar.h>

```
long int wcstol(const wchar_t * restrict nptr,
                wchar_t ** restrict endptr, int base);
```

Wide-character version of strtol.                                                                    *25.5*

---

**wcstold**     *Convert Wide String to Long Double (C99)*          <wchar.h>

```
long double wcstold(const wchar_t * restrict nptr,
                    wchar_t ** restrict endptr);
```

Wide-character version of strtold.                                                                  *25.5*

---

**wcstoll**     *Convert Wide String to Long Long Integer (C99)*          <wchar.h>

```
long long int wcstoll(const wchar_t * restrict nptr,
                      wchar_t ** restrict endptr,
                      int base);
```

Wide-character version of strtoll.                                                                  *25.5*

---

**wcstombs**     *Convert Wide String to Multibyte String*          <stdlib.h>

```
size_t wcstombs(char * restrict s,
                const wchar_t * restrict pwcs,
                size_t n);
```

Converts a sequence of wide characters into corresponding multibyte characters. pwcs points to an array containing the wide characters. The multibyte characters are stored in the array pointed to by s. Conversion ends if a null character is stored or if storing a multibyte character would exceed the limit of n bytes.

*Returns*     Number of bytes stored, not including the terminating null character, if any. Returns (size_t)(-1) if a wide character is encountered that doesn't correspond to a valid multibyte character.                                                                        *25.2*

---

**wcstoul**     *Convert Wide String to Unsigned Long Integer (C99)*          <wchar.h>

```
unsigned long int wcstoul(
    const wchar_t * restrict nptr,
    wchar_t ** restrict endptr, int base);
```

Wide-character version of strtoul.                                                                  *25.5*

---

**wcstoull**     *Convert Wide String to Unsigned Long Long Integer (C99)*          <wchar.h>

```
unsigned long long int wcstoull(
    const wchar_t * restrict nptr,
    wchar_t ** restrict endptr, int base);
```

Wide-character version of strtoull.                                                                *25.5*