

Table 25.14
Wide-String Copying
Functions and Their
<string.h>
Equivalents

<i><wchar.h> Function</i>	<i><string.h> Equivalent</i>
wscpy	strcpy
wscncpy	strncpy
wmemcpy	memcpy
wmemmove	memmove

Wide-String Concatenation Functions

```
wchar_t *wscat(wchar_t * restrict s1,  
               const wchar_t * restrict s2);  
wchar_t *wcsncat(wchar_t * restrict s1,  
                 const wchar_t * restrict s2,  
                 size_t n);
```

The functions in this group are wide-character versions of the string concatenation functions found in <string.h> and described in Section 23.6. The <wchar.h> functions have arguments of type wchar_t * instead of char *, but their behavior is mostly the same as the <string.h> functions. Table 25.15 shows the correspondence between the <string.h> functions and their wide-character counterparts.

Table 25.15
Wide-String Concatenation
Functions and Their
<string.h> Equivalents

<i><wchar.h> Function</i>	<i><string.h> Equivalent</i>
wscat	strcat
wcsncat	strncat

Wide-String Comparison Functions

```
int wcscmp(const wchar_t *s1, const wchar_t *s2);  
int wscoll(const wchar_t *s1, const wchar_t *s2);  
int wcsncmp(const wchar_t *s1, const wchar_t *s2,  
            size_t n);  
size_t wcsxfrm(wchar_t * restrict s1,  
               const wchar_t * restrict s2,  
               size_t n);  
int wmemcmp(const wchar_t * s1, const wchar_t * s2,  
            size_t n);
```

The functions in this group are wide-character versions of the string comparison functions found in <string.h> and described in Section 23.6. The <wchar.h> functions have arguments of type wchar_t * instead of char *, but their behavior is mostly the same as the <string.h> functions. Table 25.16 shows the correspondence between the <string.h> functions and their wide-character counterparts.