11-6 String/Data Conversion

A common set of applications format data by either converting a sequence of characters into corresponding data types or vice versa. Two such applications are parsing and telecommunications.

Topics discussed in this section:

String to Data Conversion
Data to String Conversion

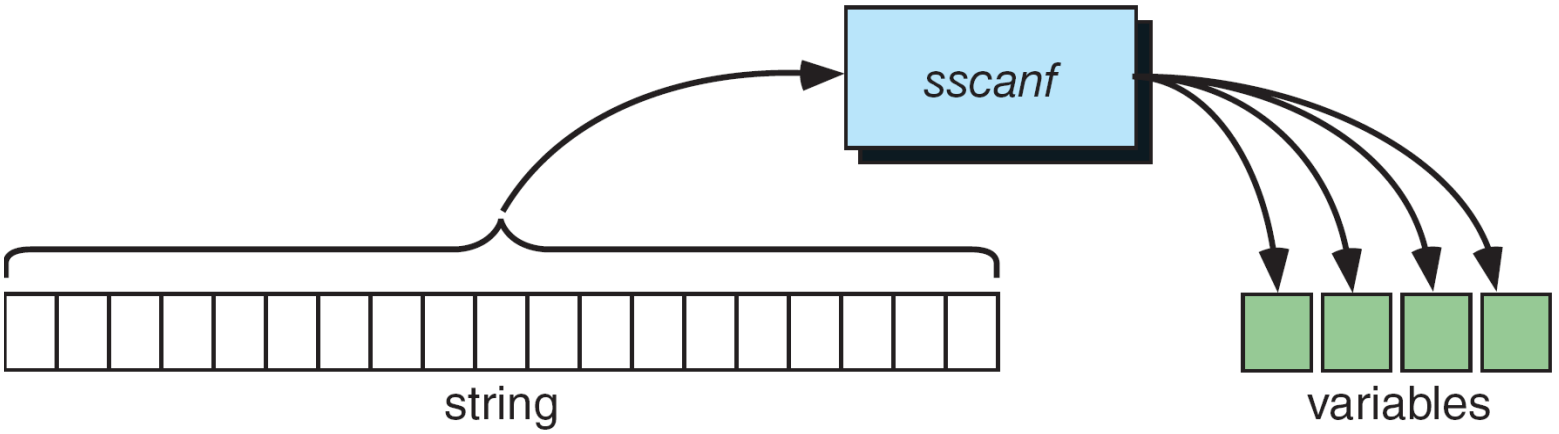

FIGURE 11-24 ***sscanf*** **Operation**

Note

sscanf is a one-to-many function. It splits one string into many variables.

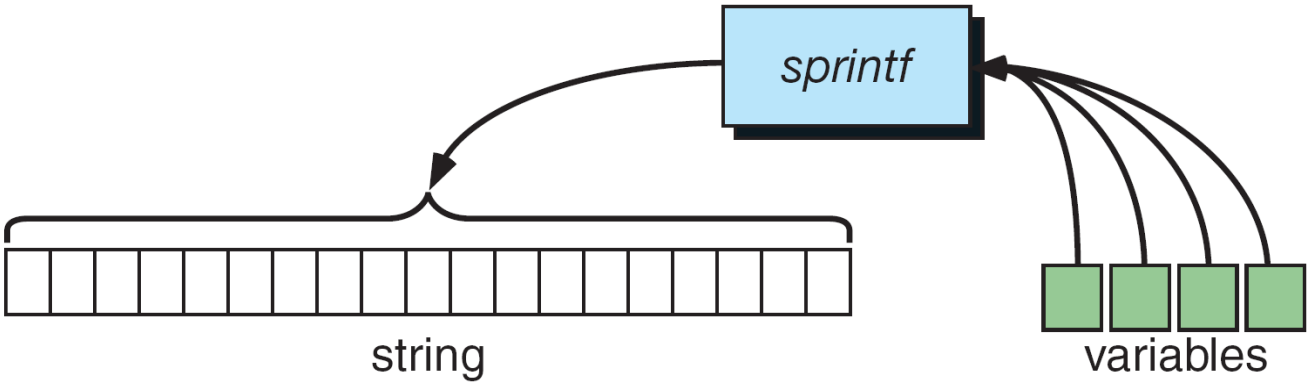

FIGURE 11-25 ***sprintf*** **Operation**

PROGRAM 11-16 Demonstrate Memory Formatting

```
1  /* Demonstrate memory formatting.
2        Written by:
3        Date:
4  */
5  #include <stdio.h>
6
7  int main (void)
8  {
9  // Local Declarations
10    char  strng[80] = "Einstein, Albert; 1234 97 A";
11    char  strngOut[80];
12    char  name[50];
13    char  id[5];
14    int   score;
15    char  grade;
16
17 // Statements
18    printf("String contains:  \"%s\"\n", strng);
```

PROGRAM 11-16 Demonstrate Memory Formatting

```
19
20      sscanf(strng, "%49[^;] %*c %4s %d %c",
21             name, id, &score, &grade);
22
23      printf("Reformatted data: \n");
24      printf("    Name:           \"%s\"\n", name);
25      printf("    id:             \"%s\"\n", id);
26      printf("    score:           %d\n", score);
27      printf("    grade:           %c\n", grade);
28
29      sprintf(strngOut, "%s %4s %3d %c",
30              name, id, score, grade);
31      printf("New string:         \"%s\"\n", strngOut);
32      return 0;
33  }   // main
```

```
Results:
String contains:  "Einstein, Albert; 1234 97 A"
Reformatted data:
    Name:           "Einstein, Albert"
    id:             "1234"
    score:           97
    grade:           A
New string:         "Einstein, Albert 1234  97 A"
```

PROGRAM 11-17 Testing for Open and Close Errors

```
 1  /* Given two Internet addresses, determine the number
 2     of unique addresses in their range.
 3        Written by:
 4        Date:
 5  */
 6  #include <stdio.h>
 7  #include <stdlib.h>
 8
 9  int main (void)
10  {
11  // Local Declarations
12     unsigned int   strt[4];
13     unsigned int   end[4];
14     unsigned long  add1 = 0;
15     unsigned long  add2 = 0;
16     unsigned long  range;
17
18     char addr1[15];
19     char addr2[15];
20
```

PROGRAM 11-17 Testing for Open and Close Errors

```
21  // Statements
22     printf ("Enter first address:  ");
23     fgets (addr1 , sizeof (addr1) , stdin);
24
25     printf ("Enter second address: ");
26     fgets (addr2 , sizeof (addr2) , stdin);
27
28     sscanf (addr1 , "%d %*c %d %*c %d %*c %d\n",
29             &strt[3], &strt[2], &strt[1], &strt[0]);
30     sscanf (addr2 , "%d %*c %d %*c %d %*c %d\n",
31             &end[3], &end[2], &end[1], &end[0]);
32
33     for (int i = 3 ; i >= 0  ; i--)
34         {
35           add1 = add1 * 256 + strt[i];
36           add2 = add2 * 256 + end[i];
37         } // for
38     range = abs (add1 - add2) + 1;
39
```

PROGRAM 11-17 Testing for Open and Close Errors

```
40      printf ("\nFirst Address: %s", addr1);
41      printf ("Second Address %s", addr2);
42      printf ("\nThe range: %ld\n", range);
43
44      return 0;
45  }   // main
```

```
Results:
Enter first address:  23.56.34.0
Enter second address: 23.56.32.255

First Address: 23.56.34.0
Second Address 23.56.32.255

The range: 258
```