



main.c

Output



```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char str[100], temp;
6      int i, j, len;
7
8      printf("Enter a string: ");
9      fgets(str, sizeof(str), stdin);
10         // ✓ safer than gets()
11
12         // Remove newline if present
13         (because fgets adds '\n')
14
15         len = strlen(str);
16
17         if (str[len - 1] == '\n') {
18             str[len - 1] = '\0';
19             len--;
20         }
21
22         // Sorting logic
23         for (i = 0; i < len - 1; i++) {
24             for (j = i + 1; j < len; j++)
25                 {
26                     if (str[i] > str[j]) {
27                         temp = str[i];
28                         str[i] = str[j];
29                         str[j] = temp;
30                     }
31                 }
32         }
33     }
```


Run



main.c

Output



```
//  safer than gets()

10
11     // Remove newline if present
        (because fgets adds '\n')
12     len = strlen(str);
13     if (str[len - 1] == '\n') {
14         str[len - 1] = '\0';
15         len--;
16     }
17
18     // Sorting logic
19     for (i = 0; i < len - 1; i++) {
20         for (j = i + 1; j < len; j++)
21             {
22                 if (str[i] > str[j]) {
23                     temp = str[i];
24                     str[i] = str[j];
25                     str[j] = temp;
26                 }
27             }
28
29     printf("Sorted string: %s\n", str
30         );
31     return 0;
32 }
33
```

Run



main.c

Output



Enter a string: zebra
Sorted string: aberz

=== Code Execution Successful ===