Reading and Writing Single Characters

This book introduces you to two C language functions used to read text from the keyboard: `scanf()` and `gets()`. Both read text from the keyboard, usually as full sentences. However, `scanf()` can be used to read in single characters, which is part of its charm — and flexibility:

```
scanf("%c",&key);
```

In this format, `scanf()` reads the keyboard for a single character, specified by the placeholder `%c`. That value is stored in the `char` variable `key`, which you can assume was already declared earlier in the program. To wit:

```
#include <stdio.h>

int main()
{
    char key;

    puts("Type your favorite keyboard character:");
    scanf("%c",&key);
    printf("Your favorite character is %c!\n",key);
    return(0);
}
```

Carefully enter this source code into your editor. Save it to disk as FAVKEY1.C.

Compile and run. Here's the sample output:

```
Type your favorite keyboard character:
```

Press a key, such as **m** (or whatever your favorite is), and then press the Enter key. You see this:

```
M
Your favorite character is M!
```

The M key is read from the keyboard, stored in the `char` variable `key`, and then displayed by `printf()`.

- Yes, you have to press the Enter key to finish your input — that's the way the `scanf()` function works.
- You can type a whole string of text, but only the first character that's typed is read by `scanf()` as the favorite key.