

C Programming

Keyboard Input (Standard Input)

The **scanf()** statement is used to input from standard input

You must place the **&** character before the variable name in a scanf() statement

Let's look at the following code sample:

```
/* Program to demonstrate simple standard input
*/

#include <stdio.h>

int main(void)
{
    int var = 0;

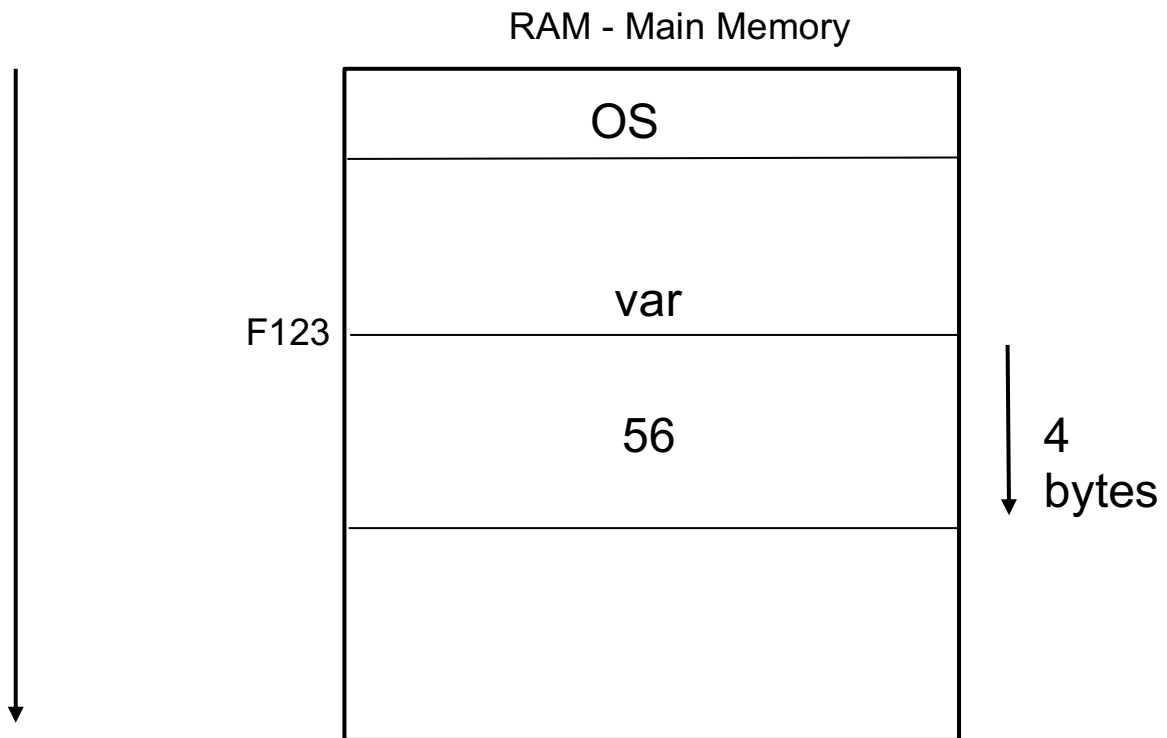
    printf("Enter any whole number\n");

    // Ensure you include the & character before the variable name
    scanf("%d", &var);
    printf("var contains %d", var);
    return 0;
}
```

Repl 3.1: <https://replit.com/@michaelTUDublin/31-Simple-Standard-Input#main.c>

Let's understand why we need the **&** character before the variable name in a scanf() statement. Take a look at this memory map diagram:

0...9, A ... F, 10 ... 19, 1A ...1F, 20 ...29. 2A ... 2F, etc,...
Hexadecimal format



```
/* Program to demonstrate entering more than 1 data item
*/
```

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int var1 = 0;
```

```
    int var2 = 0;
```

```
    int sum = 0;
```

```
    printf("Enter first number\n");
```

```
    scanf("%d", &var1);
```

```
    printf("Enter second number\n");
```

```
    scanf("%d", &var2);
```

```
    // Get the sum of the numbers entered
```

```
    sum = var1 + var2;
```

```

    printf("sum of var1 and var2 is %d", sum);
    return 0;
}

```

Repl 3.2: <https://replit.com/@michaelTUDublin/32-Entering-multiple-data-items-p1#main.c>

```

/* Program to demonstrate entering more than 1 data item
*/

#include <stdio.h>

int main(void)
{
    int var = 0;
    char my_char = ' ';

    printf("Enter a single number and a single character\n");
    scanf("%d", &var);

    while(getchar() != '\n');

    scanf("%c", &my_char);

    printf("You entered %d and %c", var, my_char);

    return 0;
}

```

Repl 3.3: <https://replit.com/@michaelTUDublin/33-Entering-multiple-data-items-p2#main.c>

Precision specification in a printf() statement

To specify a floating-point number to x decimal places, you can do the following:

```

/* Program to demonstrate entering more than 1 data item
*/

```

```

#include <stdio.h>

int main()
{
    float var = 0;

    printf("Enter any float number\n");
    scanf("%f", &var);

    // Display the number entered correct to 2 decimal places
    printf("You entered %.2f", var);

    return 0;
}

```

Repl 3.4: <https://replit.com/@michaelTUDublin/34-Precision-specification-p1#main.c>

By changing the number, e.g., .2 inside the delimiter %f (see above code), this specifies how many decimal places the float number will be displayed to standard output

```

/* Program to demonstrate entering more than 1 data item
*/

```

```

#include <stdio.h>

int main(void)
{
    float var = 0;

    printf("Enter any float number\n");
    scanf("%f", &var);
    printf("You entered %15.2f", var);

    return 0;
}

```

Repl 3.5: <https://replit.com/@michaelTUDublin/35-Precision-specification-p2#main.c>

Alternative (Quick) way to enter and display **character** data into a character variable

This involves 2 built-in C functions called **getchar()** and **putchar()**

Let's have a look at sample code:

```
/* Program to introduce getchar() and putchar()
*/

#include <stdio.h>

int main(void)
{
    char my_char = ' ';
    printf("Enter any character\n");

    // use the getchar() function
    //scanf("%c", &my_char);

    // same as above
    my_char = getchar();

    //printf("my_char contains %c", my_char);
    printf("\nYou entered ");
    putchar(my_char);

    return 0;
}
```

Repl 3.6: <https://replit.com/@michaelTUDublin/36-getchar-and-putchar#main.c>

Programming Pitfall

1. Don't forget to include the & in the scanf() statement
2. Regarding precision formatting, only the printf() statement allows this to be done. e.g., printf("you entered %.2f", num). You cannot place the precision formatting inside a scanf() statement. e.g., the following is **wrong**: `scanf("%.2f", &num);`