

## scanf Ignores White Space Typed by User

Example: 

```
int A;
int B;
scanf ("%d%d", &A, &B);
```

The user can separate the two numbers with spaces, tabs, and/or linefeeds, such as ...

```
5 42      /* A is 5, B is 42 */
5 /* two lines -> same result */
42
```

The user must push <Enter> when done.

## Other Characters in Format Must be Typed Exactly

If format includes characters

- other than format specifiers and white space
- user must **type them exactly** with no extra spaces. **Rarely useful.**

Example: 

```
int A; int B;
scanf ("%d<>%d", &A, &B);
```

Type "5<>42" and A==5, B==42.

But type "5 <>42" and A==5, while B is unchanged (no initializer, so B contains bits).

## Conversion Specifiers Similar to printf

Format Specifier	Interpretation
<b>%c</b>	store one <b>ASCII</b> character (as <b>char</b> )
<b>%d</b>	convert decimal integer to <b>int</b>
<b>%f</b>	convert decimal real number to <b>float</b>
<b>%lf</b>	convert decimal real number to <b>double</b>

## Conversion Specifiers Similar to printf

Format Specifier	Interpretation
<b>%u</b>	convert decimal integer to <b>unsigned int</b>
<b>%x</b> or <b>%X</b>	convert hexadecimal integer to <b>unsigned int</b>