

```

{
    printf("Press Enter to drop the bomb:");
    getchar();
    dropBomb();
    printf("Yikes!\n");
    return(0);
}

void dropBomb()
{
    int x;

    for(x=20;x>1;x--)
    {
        puts("          *");
    }
    puts("          BOOM!");
}

```

Type the source code as listed here. In the `puts()` function in Line 20 are 10 spaces before the asterisk.) In Line 22 are 8 spaces before `BOOM!`.

Save the file to disk as `BOMBER.C`. Compile and run.

```

Press Enter to drop the bomb:
          *
          *
          *

```

And so on. . .

```

          *
          *
          BOOM!
Yikes!

```

Yeah, it happens a little too fast to build up the nerve-tingling anticipation of a true video game, but the point here is not dropping bombs; rather, the variable `x` is used in the `dropBomb()` function. It works just fine. Nothing quirky. Nothing new. That's how variables are used in functions.

✔ See how the `dropBomb()` function declares the variable `x`:

```
int x;
```

It works just like it does in the `main()` function: Declare the variables right up front. Refer to Chapter 8 for more information about declaring variables.

✔ The `dropBomb()` function is a void because it doesn't return any values. It has nothing to do with any values used *inside* the function.