

Validation And Debugging



Testing And Debugging- Debugging

Debugging an application means actually fixing the errors (or bugs) identified by testing the app. It must then be tested again to make sure the fix hasn't caused more errors.



Debugging Terms

Let's about how debugging works, but first here's some terms:

- Debugging - the process of finding and removing errors from your code, using an IDE's Debugger tool



Fun Fact

The Term bug and debug comes from computers in the 1940's. The units were room sized and actual bugs would crawl into the moving parts. Programmers would have to physically open their machine and track down the rogue insect.



Debugging

- Breakpoint - this a line of code where you tell the debugger to stop.
- Step forward - this tells the debugger to move onto the next line



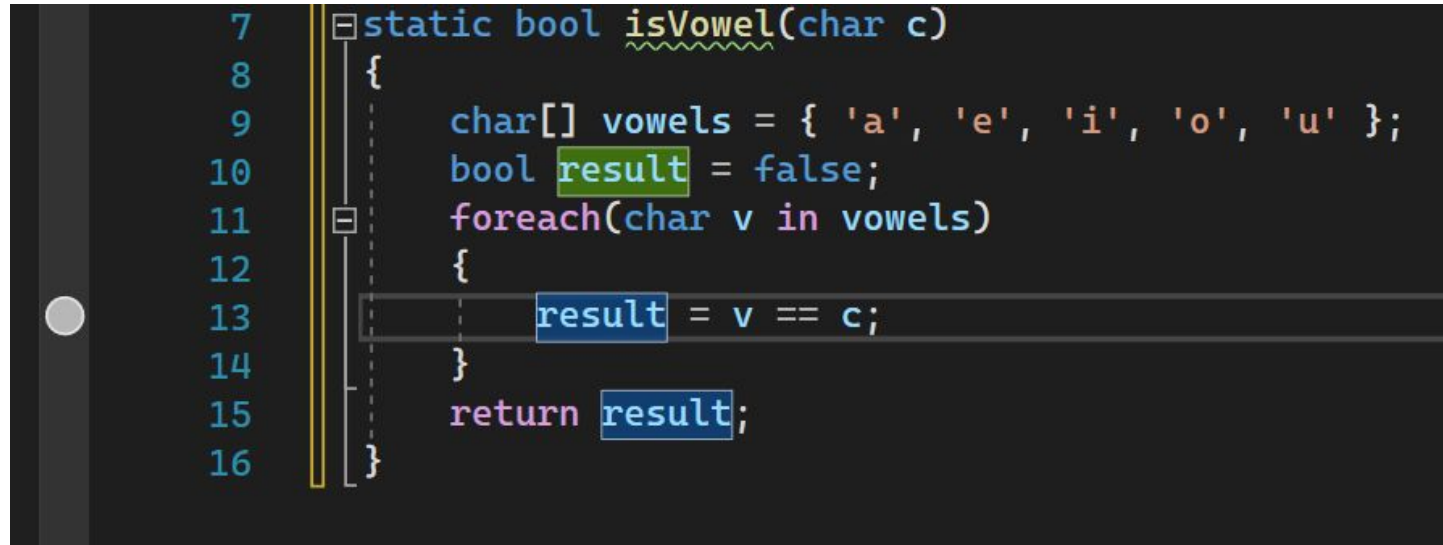
Debugging

Every debugger/IDE works a little differently, but the main idea is the same: set a break-point and see what the IDE sees at that line. Commonly, Debuggers will show you the current value of all variables as well as any methods currently running



Debugging in Visual Studio

First set a break-point, by clicking the far left of the coding panel in Visual Studio



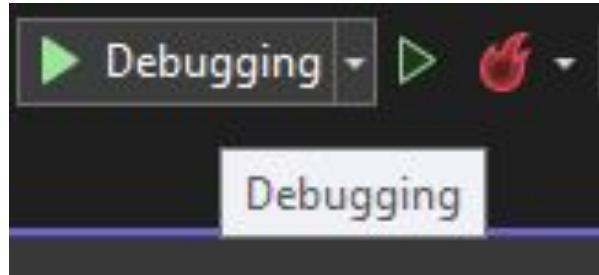
```
7  static bool isVowel(char c)
8  {
9      char[] vowels = { 'a', 'e', 'i', 'o', 'u' };
10     bool result = false;
11     foreach(char v in vowels)
12     {
13         result = v == c;
14     }
15     return result;
16 }
```

The image shows a screenshot of the Visual Studio code editor. On the left side, there is a vertical toolbar with a grey circle icon. The main area displays a C# method named `isVowel` (underlined with a wavy line). The method takes a `char c` parameter and returns a `bool`. It initializes a `char[] vowels` array with the characters 'a', 'e', 'i', 'o', and 'u'. A `bool result` variable is set to `false`. A `foreach` loop iterates over each character `v` in the `vowels` array. On line 13, inside the loop, the statement `result = v == c;` is highlighted with a blue selection box. A vertical dashed line, representing a break-point, is positioned to the left of this line. The line numbers 7 through 16 are visible on the left margin.



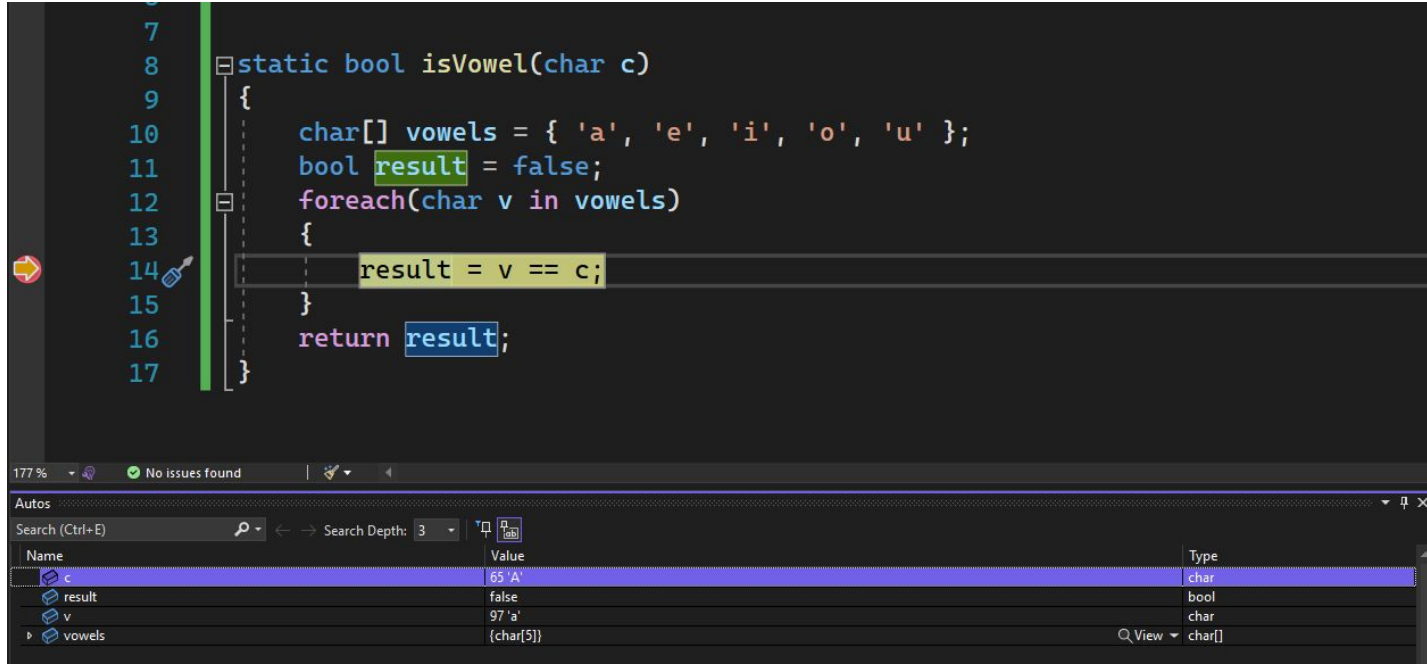
Debugging in Visual Studio

Next to run the debugger click the filled in run button. This will run your code until the first breakpoint is hit.



Debugging in Visual Studio

If the debugger ran it should look like so:



```
7
8 static bool isVowel(char c)
9 {
10     char[] vowels = { 'a', 'e', 'i', 'o', 'u' };
11     bool result = false;
12     foreach(char v in vowels)
13     {
14         result = v == c;
15     }
16     return result;
17 }
```

177 % | No issues found

Autos

Search (Ctrl+E) | Search Depth: 3

Name	Value	Type
c	65 'A'	char
result	false	bool
v	97 'a'	char
vowels	{char[5]}	char[]



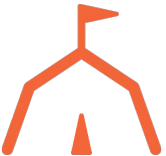
Debugging in Visual Studio

Notice on the bottom, the current values of each variable are shown. As you step through the program line by line these will update.



Debugging in Visual Studio

For stepping through your code, you have 4 different options (and continue). Experiment with each one. What do they do?



Debugging in Visual Studio

Go back to the code screenshot and see if you can figure out the two problems with it.
Discuss it with each other.

