

# Re: Regex issue

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-09/msg03742.html>

---

- *From:* Jesse Houwing <[jesse.houwing@xxxxxxxxxxxxxxxxxxxxx](mailto:jesse.houwing@xxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Sun, 30 Sep 2007 20:05:10 +0000 (UTC)
- 

Hello John,

I am trying to do a pretty simple pattern match using regex.

The pattern is `^(?:(<Item>.*?))@: @$`.

This should return a match for `test123@: @` but does not, instead it never returns when I call `Regex.Match`, I have to kill the thread.

The code is below, The Regulator (some will know it) returns the correct results, I cannot work out why.

```
RegexOptions options = RegexOptions.IgnoreCase |  
RegexOptions.IgnorePatternWhitespace | RegexOptions.Multiline; Regex  
matcher = new Regex(@"^(?:(<Item>.*?))@: @$", options); Match match =  
matcher.Match(@"test123@: @"); //Never returns from here.
```

I have tried this on two machines, it just doesnt work.

TIA

JB

`^(?:(<Item>.*?))@: @$`

I'd guess that the problem lies in the last `*`, but I'm unsure why it would hang the parser. I've seen these kinds of problems before where a lot of backtracking is possible.

a better pattern would be: `^(?<item>.*?)@: @$`

The extra `?` after the first `*` isn't really needed, unless there are more than one `@: @` signs in your text at the end of a line (which I doubt). But as you've not sent us the text you're trying to extract your pattern from I'd have to guess that.

Another pattern that would probably work would be:

Re: Regex issue

`^(?<item>[^\@]*)@:@$` but in there I assume that @ does not precede the @:@ at all.

I hope this helps. If not, please add some sample input.

Another possible problem might be that your input is simply too large. The engine isn't very good at reading multiple megabytes of text at once. As you're looking at the lines one by one, you could use that to your advantage and load a couple of lines, try to match and load the next couple of lines. A `StringReader` would be ideal for that purpose.

—  
Jesse Houwing  
jesse.houwing at sogeti.nl