

Uokviriti i šrafirati poklapanja sa odgovarajućim *regexima*. U slučaju više uzastopnih poklapanja, jasno naznačiti razliku. U slučaju da je razmak (*whitespace*) obuhvaćen na početku ili na kraju poklapanja, jasno naznačiti i to. Pretpostaviti da se simbol . ne poklapa sa znakom za novi red. Pretpostaviti da nema razmaka na kraju reda (ni kod regexa, ni kod ulaznih linija).

1.

<code>" .* "</code>
<code>"Pozdrav, ", rekla je ona, "kako ste danas?" "Da li ste došli gdje ste trebali?", upitao sam. Previše pitanja. "Znate li", interesuje se, "koliko je sada časova u Njujorku?" " .* " {Red,"White","Blue"} string result = "We've achieved " + points + "points!";</code>

2.

<code>" .*? "</code>
<code>"Pozdrav, ", rekli smo im, "kako ste danas?" "Da li sam došao gdje sam trebao?", upitao sam se. Puno pitanja. "Znate li", interesuje se, "koliko je sada časova u Parizu?" " .* " .*? " {"Blue","White","Red"} string result = "We've achieved " + points + "points!";</code>

3.

<code>[a-z0-9_ -]{3,16}\$</code>	<code>.?[j-t]..[a-z][2-3e-j()]{0,2} (?:f.)?(.?) (.)\2\1</code>
<code>markoMarkovic marko_markovic95 vvv_www0-9{3,16     kristinal23456 a-z0-9_ - 222a-z0-vvvv0 _____ -----_--_ xv peropetarovic1234 =____</code>	<code>jadajadajada123 asdf bbbb qwerty uiop _!=poziv Da li znaš g()funkcije() f==ZZ</code>

4.

"[a-zA_Z, ]+" "\\..*?" "k[a-z ]*(\\)??"
".*?" "Pozdrav, ", rekli smo im, "kako ste danas?" "Da li sam došao", upitao sam se, "gdje sam trebao?". Previše pitanja. "Znate li", interesuje se, "koliko je sada časova u Moskvi?" "\\..*"\\..*?" { "White", "Blue", "Red" } string result = "We've achieved " + points + "points!"; "[a-zA_Z, ]+" "\\..*?" "k[a-z ]*(\\)??"

5.

[abcdefghijklmnopqrstuvwxyz][abcdefghijklmnopqrstuvwxyz]*
abcdefghijklmnopqrstuvwxyz [a-z] [A-Z] [:alpha]

6.

asdf asdfbcer	asdfbcer asdf	(.{1,3}).\1	((a b) c).\2.{1,2}\1
asdf asdfbcer asdfbcer asdf asdebcerasdf	asdf asdfbcer asdfbcer asdf asdebcerasdf	abab abcc abcababca bbaaaabb aaaaaacba abcbacabcac	ababca bcb baca bbab abcbacba aaba babab

7.

<code>\&lt;([<sup>0-9</sup>&gt;] 4)* \\" [<sup>0-9\s</sup>]*&gt;\$ \b[a-z]+\b</code>
<code>&lt;state id="3" name="q3"&gt;#13;   &lt;final/&gt;#13; &lt;/state&gt;#13; &lt;state id="4" name="q4"&gt;#13; &lt;/state&gt;#13; &lt;state id="7" name="q7"&gt;   &lt;x&gt;922.0&lt;/x&gt;   &lt;y&gt;110.0&lt;/y&gt;   &lt;final/&gt; &lt;/state&gt;</code>

8.

<code>[a-h]..? ^(.)\1\1 /?\$</code>	<code>-?(0x 0b 0)?[1-9A-F][0-9A-F]+.[0-9A-F]*</code>
wo i know. iii know! do you? abababce cab cab cab? [[[func	1 -1 5250130 0x10 0x-15 0x5F+037=-512 0b100001000 0b001010000 -256.371 -1F1.02 0b11+0b110 0b1FFF 0b 1000 0000 0xAF00B352 102571

9.

<code>[e-w]{2,3} [[:alpha]]\d\.\$</code>	<code>(\b\w+?\b)\W(\b\w+?\b)\W.*?\W\2\W\1</code>
!XP23koq-0dkQCw44i 04-czh8 _ _c:1=wXzzz[]4. ><Xkp1-`1-`-+w+9-x	Suspendisse tempor condimentum efficitur. Vivamus semper eget sapien ac tristique, condimentum ac sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mauris, a suscipit nulla. Praesent suscipit accumsan. A posuere felis tristique at. Donec varius, sit amet posuere augue. Pellentesque ut metus justo, justo metus ut pellentesque. Sed lorem a, pretium eleifend mauris. Sed lorem dui feugiat.

10.

**(https?:\/\/\/)?(www\.)?[a-z0-9@:.\+=]\*\.[A-Za-z]{2,6}\b[p-u?\/]\***

www.google.com  
https://etf.unibl.org/  
etfunibl.org  
.NET  
!%.net  
\_etf.netcom  
google.com/?q=\search=hello%20world  
#.com?q=:80.io

11.

**-?(0x[0-9A-F]+\.?[0-9A-F]+)|(0b[01]+\.?[01]{,})|(0[0-7]+\.?[0-7]\*)|[0-9]+\.?[0-9]+**

1  
-1  
5250130  
0x10  
0x-15  
0x5F+037=-512  
0b100001000  
0b001010000  
-256.371  
-1F1.02  
0b11+0b110  
0b1FFF  
0b 1000 0000  
0xAF00B352  
102571

12.

**\b[a-z]{2,3}[n-t]|String|^[NnDd]\*[uUiI]+|Integer|[fF]loat|(a-z)[A-Z]\*.)+?**

Duis nec elit et sapien porta dignissim.  
fFloat x = new String(new Integer())  
Ffloat y = new Integer(new String())  
Integer vitae leo id enim imperdiet dignissim.  
Nulla eget odio nec ante convallis fringilla vel sit amet enim.  
[a-z]A-Z(:alpha-z)D\\d\.\$

13.

**\b.f[0-9]+\b|(?<=\(.)epsilon(=?<=))**

```
q0( a ) = q0
q5(b ) =q5
q6(\epsilon)= q4
q8(c ) = q8
q1(\epsilon ) = q2
q2(a) = qf3
qf3(b) = q0
q0( b)= q1
q6( c ) = qf7
q2( a )= q4
qf9(\epsilon) = q1
q2(b) = q4
qf9 (\epsilon)= qf7
q0 (\epsilon) = q5
qf3(\epsilon)= q6
q6 ( \epsilon ) = q1
q0( c) = q8
q5(\epsilon) = q2
```

14.

**(\D) (\<|=) (.\*)\[.\*\]| \[...\]| ([-azC-V]{2,3}).\*\4**

```
"[a-zA_Z, ]+"|"\..*?"|"k[a-z ]*(\)??"
\b[a-z]{2,3}[n-t]|String|Integer|[fF]loat|(a-z)[A-Z]*.+.+?
\<([ ^0-9>]|4)*|\"[ ^0-9\s]*>$|\b[a-z]+\b
(https?:\//\/?)(www\.)?[a-z0-9@:.\+=]*\.[A-Za-z]{2,6}\b[p-u?\//]*
(?<=BEGIN) (?: (.|\s)* ) (?=ERROR) | (?<=PASSWORD) ( (.|\s)* ) END|^.*HACKER|DETECTED.*?$
```

15.

**(\b\w+\b) (?.\*\b\1\b)**

Uokviriti i šrafirati poklapanja sa odgovarajućim regularnim izrazima (regex). U slučaju više uzastopnih poklapanja, jasno naznačiti razliku. U slučaju da je razmak (whitespace) obuhvaćen na početku ili na kraju poklapanja, jasno naznačiti i to. Pretpostaviti da se simbol . ne poklapa sa znakom za novi red. Pretpostaviti da nema razmaka na kraju redova (ni kod regexa, ni kod ulaznih linija), osim ako nije drugačije naglašeno.

16.

[1-1]0\..*+ 172\.(?:1[6789] 2\d 3[0-1]) .*+ 192\.168\..*
1079.198.111 192.168.15.74 73.230.228.599.112 56.116.181.171 18.161.167 10..168.110.35 10.207.126.59 172.19.232.220 34.129.194.168 54.600.10.3692 192.168.20.209.175 10.250.210.96 192.168.161.29 129.3.81.198 172.26..... 23228.223.173 10.168.4451 173.245.237.97

17.

(?<=BEGIN) (?: (.  \s) *) (?=ERROR)   (?<=PASSWORD) ( (.  \s) *) END   ^. *HACKER DETECTED. *?\$
nZ`Ca7n\$\$680#4,7-1lc2q) / ) n`k-\$`3`:3zLC`) 0E;aW6BEGIN5a* ooo5S<*;5@11S2n#S)+nyl,KPC[p3W-, /2,m) "-1o7,D~~2Z4Wxl:,ERROR~Z#0 D95533;D;_d"e`L0-m DlO@@k'2@V`HACKERn~4/31-h3;6-288 p3W-, /2,m) "-: ,LarA`+t4WP`624Z4i6RL\$S /lLrL@42ZDETECTEDD-D@`O 'q59c_',Z7k\$-2-,4-/PASSWORDpLZx-<--690[;4WWEND- nePR,['4379v\$*)_ _3"/LP\$`03_+~X1- !5`g8Z55Zjt:56`@@la"4*ehP5a* 90sDvvCEND?

```
(public|private)? (int|float|double|bool|char|string|byte|var) (this\.?)?w*( *| = [ \w.(!=+]*)
```

```
class State
{
    private string id;
    public bool    isFinal;
    bool    isInitial;
    private string name;
    State parentState;
    public State(string id, string name, bool isFinal, bool isInitial)
    {
        this.id = id;
        this.name = name;
        this.isFinal = isFinal;
        this.isInitial = isInitial;
    }
}

static void Main(string[] args)
{
    var automaton = Xdocument.Parse(File.ReadAllText(args[0]))
        .Element("structure").Element("automaton");
    // States data.
    var statesElements = automaton.Elements("state");
    var states = new Dictionary<string, State>();
    foreach (var element in statesElements)
    {
        var id          = element.Attribute("id").Value;
        var name         = element.Attribute("name").Value;
        var isFinal      = element.Element("final")    != null;
        var isInitial    = element.Element("initial")  != null;
        if (isInitial && id != "0")
            throw new Exception("Initial state should have var id = 0");
        states.Add(id, new State(id, name, isFinal, isInitial));
    }

    var transitions = automaton.Elements("transition");
    var sb = new StringBuilder();

    foreach (var transition in transitions)
    {
        var from = states[transition.Element("from").Value];
        var to   = states[transition.Element("to").Value];
        var read = transition.Element("read").Value;
        if (string.IsNullOrEmpty(read)) read = "\\epsilon";
        string fromState = "q" + (from.isFinal ? "f" : "") + from.id;
        string toState   = "q" + (to.isFinal ? "f" : "") + to.id;
        sb.Append($"\\item[]  ${fromState}({read})={toState}$"
            + Environment.NewLine);
    }
    Console.WriteLine(sb.ToString());
    Console.WriteLine(1);}

```