Informatics 2A: Tutorial Sheet 2 - SOLUTIONS

John Longley

1. (a) Let $\underline{D}$ (for decimal digits) be the language:

$$0+1+2+3+4+5+6+7+8+9.$$

The required regular expression is then:

$$\underline{D}^*(\underline{D}\ .+.\ \underline{D})\underline{D}^*(\epsilon+((E+e)(\epsilon+\text{`}+\text{'}+\text{`}-\text{'})\underline{D}\ \underline{D}^*))$$

(many variations are possible)

(b) The egrep command is `egrep` *pattern* `foo.java`, where *pattern* is:

```
"[0-9]*(([0-9]\.)|(\.[0-9]))[0-9]*([Ee][+-]?[0-9]+)?"
```

In many versions of the machine syntax, `[0-9]` can be abbreviated e.g. to `[\d]` or `:digits:`

2. (a) Let $\underline{S}$ be the language of string characters, i.e. all input characters except `"`(double quote) and \(backslash).
Then the required regular expression is:

$$\texttt{"}\,(\underline{S}+\backslash\,(\,\texttt{"}+\backslash+\texttt{b}+\texttt{t}+\texttt{n}+\texttt{f}+\texttt{r}))^*\,\texttt{"}$$

(b) Translating this into machine syntax with suitable escapes for `"` and \, we get:

```
egrep "\"([^\"\\]|([\\][\"\\btnfr]))*\"" foo.java
```

(Note the two levels of escape sequences involved here.)

(c) A possible grep command is:

```
egrep "^[^\"]*\"[^\"]*(\"[^\"]*\"[^\"]*)*$" foo.java
```

(Note that `^` and `$` match the start and end of line respectively.)

3. In each of (a)-(e) below, let L denote the language in question.

(a) Not regular.
Given $k \geq 0$, consider $x = a^k$, $y = b^{k+1}$, $z = \epsilon$. Then $xyz \in L$.
Given any splitting of y as uvw where $v \neq \epsilon$, take $i = 0$.
Then $uv^iw = uw = b^l$ for some $l < k+1$. So $xuv^iwz = a^kb^l \notin L$ since $k \geq l$.

(b) Not regular.
Given $k \geq 0$, consider $x = \epsilon$, $y = a^k$, $z = ba^k$. Then $xyz \in L$.
Given any splitting of y as uvw where $v \neq \epsilon$, take $i = 0$.
Then $uv^iw = uw = a^l$ for some $l < k$, so $xuv^iwz = a^lba^k \notin L$.

(c) Not regular. Given $k \geq 0$, consider $x = \epsilon$, $y = a^h$, $z = b^h$ where $h = \max(k, 2)$, so $xyz \in L$ (no. substrings aa = no. substrings bb = $h - 1 > 0$) and $|y| \geq k$.
Given any splitting of y as uvw where $v \neq \epsilon$, take $i = 0$.
Then $uv^iw = uw = a^l$, for some $l < h$, which has at most $h - 2$ substrings aa. So $xuv^iwz = a^lb^h \notin L$.

(d) Regular. The trick is to note that the strings of L are exactly those that switch between a's and b's an even number of times, i.e. those that start and end with the same letter. This is because we get an ab every time we switch from a's to b's, and a ba every time we switch from b's to a's.
So L corresponds to the following regular expression:

$$\epsilon \;+\; a \;+\; b \;+\; a(a{+}b)^*a \;+\; b(a{+}b)^*b \;\;.$$

(e) Not regular.

Given $k \geq 0$, let $p \geq k$ be prime (using Euclid's theorem that there are infinitely many prime numbers).
Consider $x = \epsilon$, $y = a^p$, $z = \epsilon$. Then $xyz = a^p \in L$.
Given any splitting of y as uvw where $v \neq \epsilon$, take $i = p + 1$.
Note that $|xuwz| = p - |v|$ and $|v^i| = (p+1)|v|$, so $|xuv^iwz| = p - |v| + (p+1)|v| = p(|v|+1)$, which is not prime since both factors (p and $|v|+1$) are ≥ 2. Thus $xuv^iwz \notin L$.

Here is a more abstract version of the above proof, for those who know about arithmetic progressions.
Consider $x = \epsilon$, $y = a^p$, $z = \epsilon$, as above, and let uvw be any splitting of y with $|v| \geq 1$.
Suppose, for contradiction, that for every $i \geq 0$, we have $xuv^iwz \in L$, i.e., $|u| + i|v| + |w|$ is prime.
Then $(|u|{+}|w|{+}i|v|)_{i \geq 0}$ is an infinite arithmetic progression of primes.
But there are no infinite arithmetic progressions of prime numbers! (Proof: Consider an arithmetic progression $(a + ib)_{i \geq 0}$ where $b \geq 1$. If $a \leq 1$ then the $i = 0$ entry is not prime. Otherwise the $i = a$ entry is not prime since $a + ab = a(b+1)$ which has two factors ≥ 2.)
This contradiction means that, after all, there must be some $i \geq 0$ for which $xuv^iwz \notin L$.

(A side remark, for those interested in such things: the Green-Tao Theorem (2004), a celebrated result in number theory, states that, even though the prime numbers do not contain any infinite arithmetic progressions, they do contain arbitrarily long ones. So one cannot place any fixed upper bound on the i needed in the above proof in advance.)