Perl Search/Replace

remove all lines from a file:

```
perl -p -i.bak -e 's#^\*X-something.*\n$##g' $filename (remove *X-something.*\n)
perl -p -n -i.bak -e 's/^\*X-pol.*\n$//g' $filename
perl -p -n -i.bak -e 's/\r\n/\n/' $filename
awk '{print $3}' $filename
```

Regular Expressions

. matches an arbitrary character, but not a newline unless it is a single-line match (see m//s).
(…) groups a series of pattern elements to a single element.
^ matches the beginning of the target. In multiline mode (see m//m) also matches after every newline character.
$ matches the end of the line. In multiline mode also matches before every newline character.
[…] denotes a class of characters to match. [^ …] negates the class.
(… | … | …) matches one of the alternatives.
(?# TEXT) Comment.
(?: REGEXP) Like (REGEXP) but does not make back-references.
(?=REGEXP) Zero width positive look-ahead assertion.
(?! REGEXP) Zero width negative look-ahead assertion.
(? MODIFIER) Embedded pattern-match modifier. MODIFIER can be one or more of i, m, s, or x.

Quantified subpatterns match as many times as possible. When followed with a ? they match the minimum number of times.

Example: Url =~ /^http\:.{2}w{1,3}\.([a-c]{1}.+)\.(\w{2,3})\/$/
These are the quantifiers:
+ matches the preceding pattern element one or more times.
? matches zero or one times.
***** matches zero or more times.
{N,M} denotes the minimum N and maximum M match count. {N} means exactly N times; {N,} means at least N times.

\w matches alphanumeric, including _, \W matches non-alphanumeric.

\s matches whitespace, \S matches non-whitespace. \d matches numeric, \D matches non-numeric. \A matches the beginning of the string, \Z matches the end. \b matches word boundaries, \B matches non-boundaries. \G matches where the previous m//g search left off. \n, \r, \f, \t etc. have their usual meaning. \n neue Zeile (newline) \r Return \f neue Seite (form feed) \t horizontaler Tabulator	

\l	nächster Buchstabe klein
\u	nächster Buchstabe groß
\L	Buchstaben bis \E klein
\U	Buchstaben bis \E groß
\E	(beendet \L , \Q und \

\1 …\9 refer to matched subexpressions, grouped with (), inside the match.

\10 and up can also be used if the pattern matches that many subexpressions.

Each character matches itself, unless it is one of the special characters + ? . * ^ $ ( ) [ ] { } | \.

The special meaning of these characters can be escaped using a \.