

SUPSI

Ambienti Operativi: Espressioni Regolari

Amos Brocco, Ricercatore, DTI / ISIN

Vi ricordate 'grep' ?

- Abbiamo visto che il comando **grep** può essere utilizzato per “estrarre” le righe di un testo o di un input (con una pipe) che contengono una determinata parola

```
[X] bash
```

```
utente@host:~/Documenti/Privato$ grep mani testo.txt  
gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire.  
Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le  
ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.
```

... e se volessimo visualizzare tutte le righe che contengono solo la parola “mani” (e non Romani o domani) ?

Ricerca di una stringa di testo...

Cerco la parola "mani"...



Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.

Ricerca di una stringa di testo...

Cerco la parola "mani"...



Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.

```
T : String := "...testo...";
L : Integer := T'Length;

for I in Integer range 0 .. L-4 loop
    if T(I) = 'm' and
        T(I+1) = 'a' and
        T(I+2) = 'n' and
        T(I+3) = 'i' then
        -- Trovato!
    end if;
end loop;
```

Ricerca di una stringa di testo...

Cerco la parola "mani"...



Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le **mani**, stavo a sentire. Gli Egiziani trasmisero ai Rom**ani** le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma do**mani** possiamo esser poveri. E non si misero in cammino a **mani** vuote.

```
T : String := "...testo...";
L : Integer := T.Length;

for I in Integer range 0 .. L-4 loop
    if T(I) = 'm' and
        T(I+1) = 'a' and
        T(I+2) = 'n' and
        T(I+3) = 'i' then
        -- Trovato!
    end if;
end loop;
```

Non ci siamo!



Ricerca di una stringa di testo...

Cerco la parola "mani"...



Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a **mani** vuote.

```
T : String := "...testo...";
L : Integer := T'Length;

for I in Integer range 0 .. L-6 loop
    if T(I) = ' ' and
        T(I+1) = 'm' then
            T(I+2) = 'a' and
            T(I+3) = 'n' and
            T(I+4) = 'i' and
            T(I+5) = ' ' then
                -- Trovato!
            end if;
        end loop;
```

Non ci siamo!



Ricerca di una stringa di testo...

Cerco la parola "mani"...



Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le **mani**, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a **mani** vuote.

```
T : String := "...testo...";
L : Integer := T'Length;

for I in Integer range 0 .. L-6 loop
    if T(I) = ' ' and
       T(I+1) = 'm' then
       T(I+2) = 'a' and
       T(I+3) = 'n' and
       T(I+4) = 'i' and
       (T(I+5) = ' ' or T(I+5) = ',') then
           -- Trovato!
       end if;
end loop;
```

OK!



Ricerca di una stringa di testo...

Cerco la parola "mani"...



Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le **mani**, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a **mani** vuote.

```
T : String := "...testo...";
L : Integer := T'Length;

for I in Integer range 0 .. L-6 loop
  if T(I) = ' ' and
    T(I+1) = 'm' then
      T(I+2) = 'a' and
      T(I+3) = 'n' and
      T(I+4) = 'i' and
      (T(I+5) = ' ' or T(I+5) = ',') then
        -- Trovato!
      end if;
    end loop;
```

E se invece della virgola avessi '.' ?



Espressione regolare

Cerco la parola "mani"...



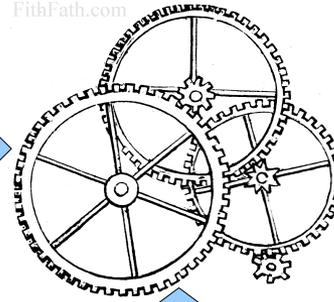
\bmani\b

Cos'è un'espressione regolare?

- È uno strumento per **ricercare sequenze di caratteri** (stringhe) **all'interno di un testo** (o flusso di caratteri)
- È un pattern o template per configurare un “motore delle espressioni regolari”

Motore regexp

FithFath.com



Lorem ipsum ea vis elitr alienum persequeris, pri autem corpora sensibus no. Minim congue definitiones ius eu, et laoreet invenire ilberavisse est, no cum viris eruditi euripidis. Vim illud commodo malestatis ne, elit quas dolore cu mel. Impedit fuisset laboramus id mea, mel ei tale harum, qui an habeo graecis. Has persecuti comprehensam signiferumque ne, id sit vide inani reprimique. Etiam omnesque te nam, sit diam legimus salutatus eu.

Mea denique mandamus consectetur in, sea ei diam justo tantas. Ut tale dolorum fastidii eum, te dicunt iuvaret principes has, omnis saepe offendit sit ei. Munere populo singulis ne eum, no iudico inciderint sadipscing qui, sea ex equidem voluptua assentior.

input

us consectetur in, sea ei diam justo tantas. Ut
um, te dicunt iuvaret principes has, omnis saene

match

`\bjusto\b`

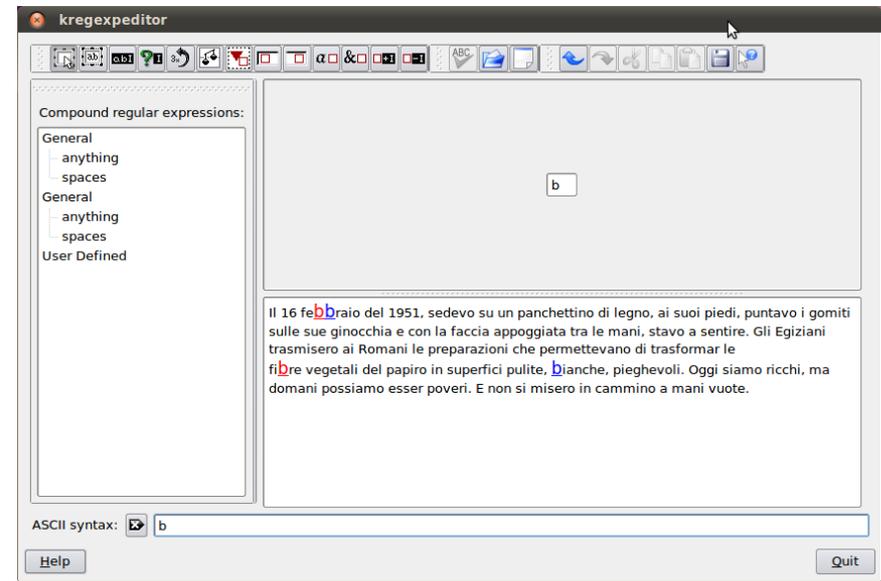
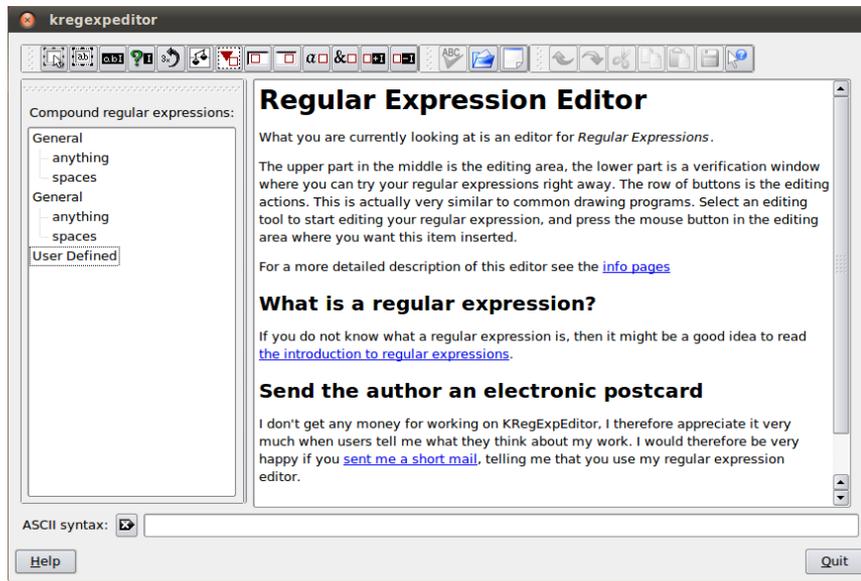
pattern

Qualche definizione

- **regexp**
 - **Reg**ular **exp**ression, espressione regolare
- **pattern**
 - Schema che descrive, utilizzando il linguaggio delle espressioni regolari, la stringa da ricercare
 - Solitamente si dice “Espressione regolare” per indicare il pattern
 - Un'espressione regolare è costituita da più **termini**, che determinano “cosa” vogliamo cercare
- **pattern matching**
 - Operazione per verificare se una stringa corrisponde a (o contiene) un pattern definito
- **pattern substitution**
 - Operazione di sostituzione di una stringa corrispondente a un pattern con un'altra stringa

kregexpeditor

- Tool grafico per sperimentare con le espressioni regolari



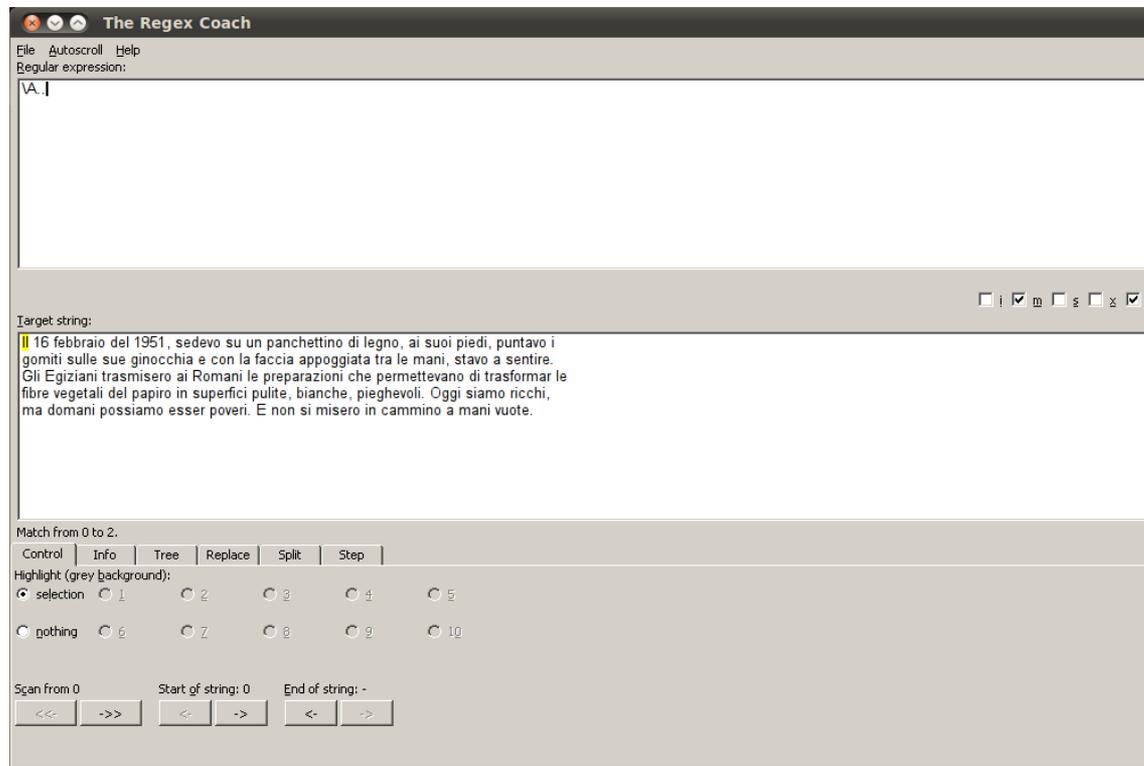
- Attenzione:
 - Il motore di kregexpeditor non è molto robusto e può generare errori con espressioni complesse
 - Non tutti i caratteri speciali sono supportati
 - le espressioni regolari riconosciute da kregexpeditor sono “case-insensitive” !

kregexpeditor

- Scaricate l'archivio **kregexpeditor.tar.gz** da Moodle, e il file **testo.txt**
- Estraiete il contenuto dell'archivio nella vostra directory personale
- Eseguite il file kregexpeditor.sh
- Incollate il contenuto di **testo.txt** nel pannello inferiore

The Regex Coach

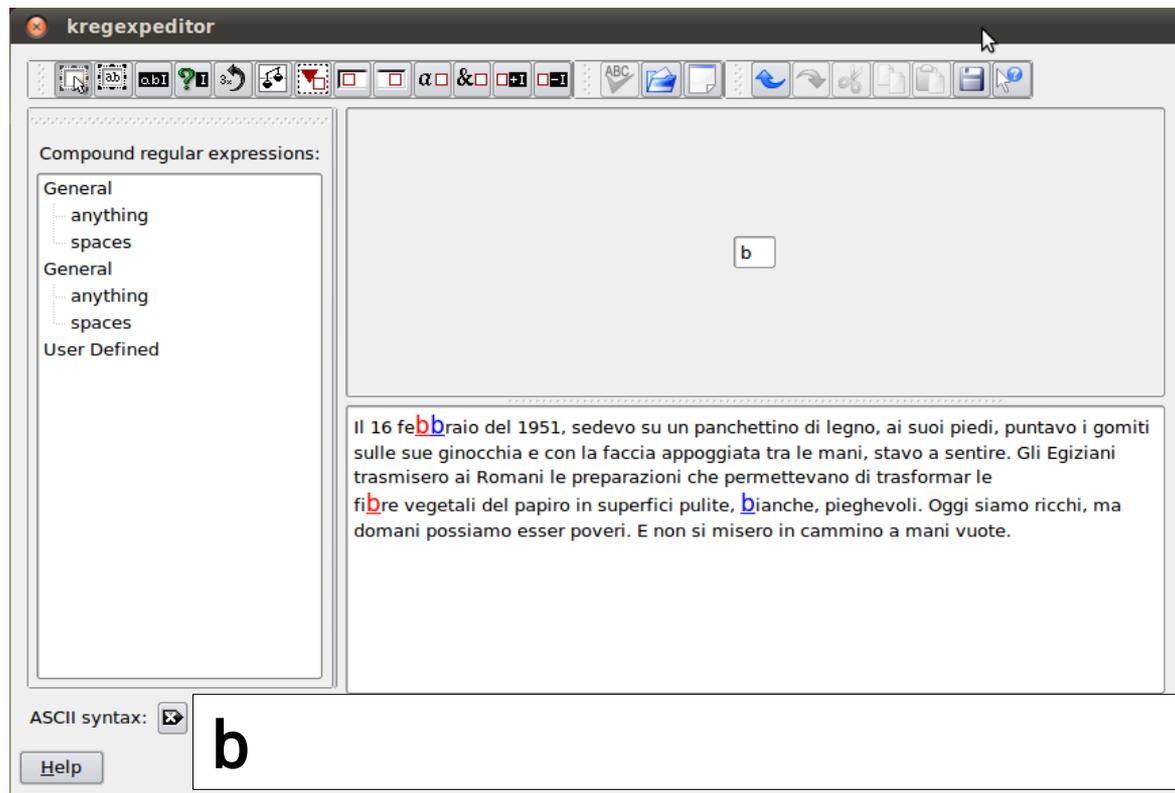
- <http://weitz.de/regex-coach/>
 - Download: <http://weitz.de/files/regex-coach.exe>
- Per Windows (funziona anche su GNU/Linux con Wine)



Pattern semplici: singoli caratteri

Cerco il carattere 'b'

L'espressione regolare è composta dal singolo carattere da ricercare

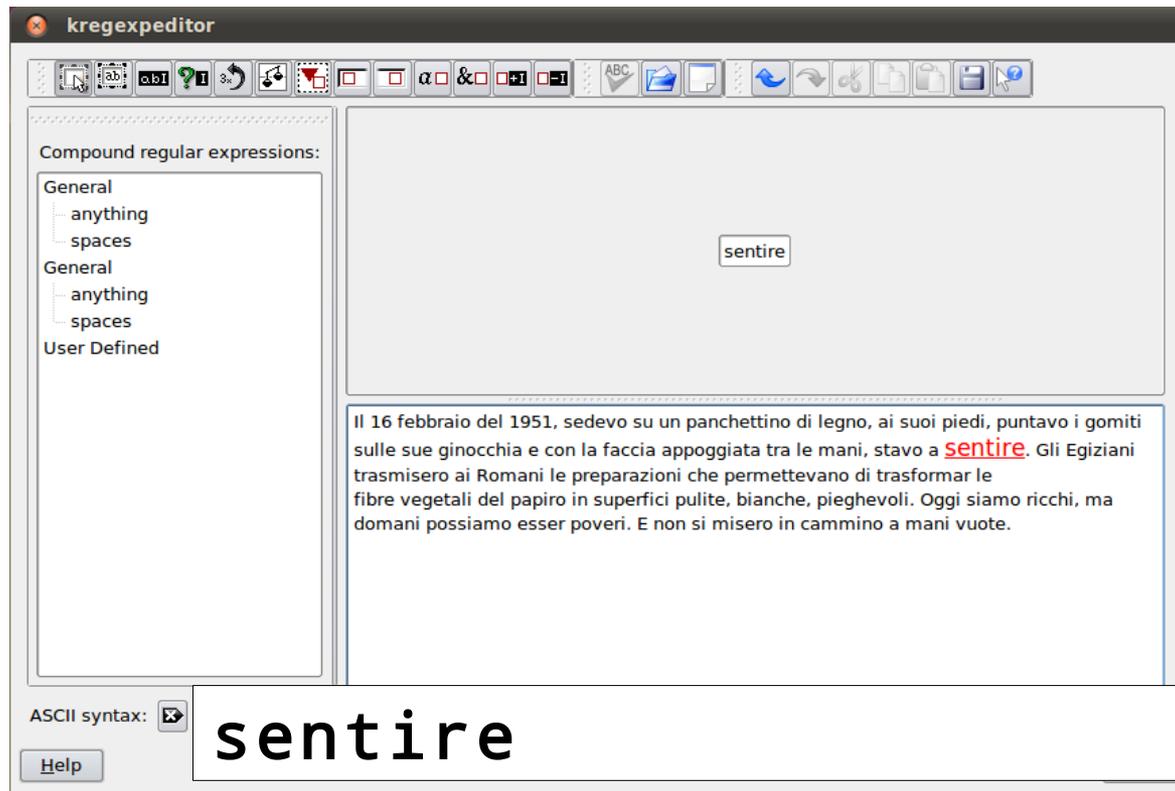


The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a tree view under 'Compound regular expressions:' with categories: 'General' (containing 'anything' and 'spaces'), 'General' (containing 'anything' and 'spaces'), and 'User Defined'. The main editing area contains the regular expression 'b'. Below the main area, there is a text preview showing a paragraph of Italian text with the character 'b' highlighted in red in the words 'febbraio', 'bianche', and 'fibre'. At the bottom, there is an 'ASCII syntax:' checkbox which is checked, and a large text field containing the character 'b'. A 'Help' button is located at the bottom left.

Pattern semplici: sequenze di più caratteri (stringhe)

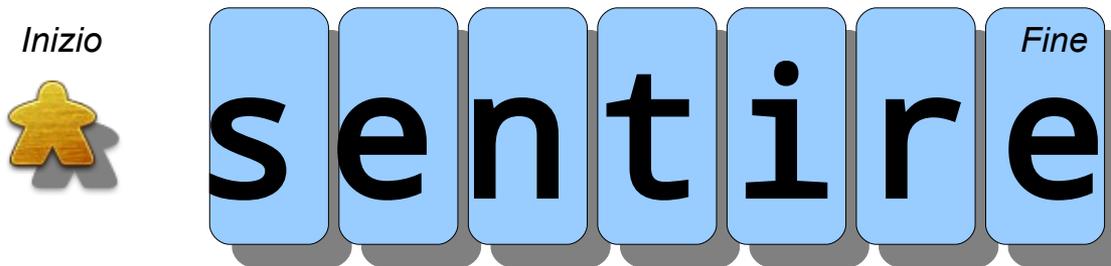
Cerco la parola 'sentire'

L'espressione regolare è composta dalla sequenza di caratteri da ricercare



Come funziona il matching

- Immaginiamo un gioco da tavolo,
 - Ogni termine dell'espressione regolare corrisponde a una casella
 - Invece di un dado, per spostarmi leggo il testo di input carattere per carattere:
 - se il carattere letto corrisponde alla casella successiva mi sposto in avanti
 - altrimenti torno all'inizio
- Lo scopo del gioco è ottenere una corrispondenza arrivando alla casella “Fine”



Come funziona il matching

a vo a sentire. Gli Egiziani trasmisero ai Romani le


Carattere corrente

Inizio


s e n t i r e

Fine

Come funziona il matching

V o a sentire. Gli Egiziani trasmisero ai Romani le


Carattere corrente

Inizio  **s e n t i r e** Fine

Come funziona il matching

O a sentire. Gli Egiziani trasmisero ai Romani le p

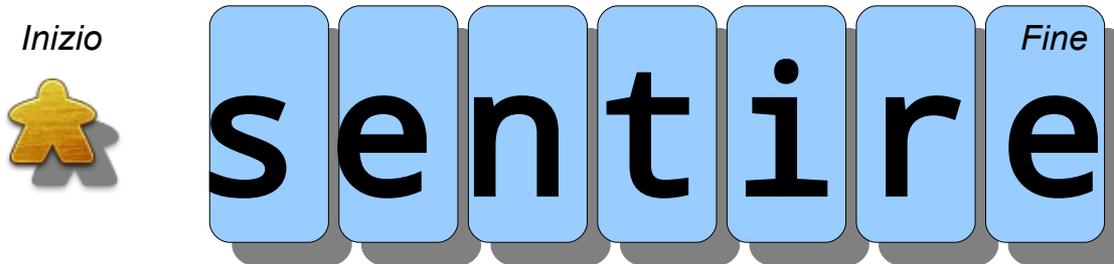

Carattere corrente

Inizio  **s e n t i r e** Fine

Come funziona il matching

a sentire. Gli Egiziani trasmisero ai Romani le pr


Carattere corrente



Come funziona il matching

a sentire. Gli Egiziani trasmisero ai Romani le pre

Carattere corrente

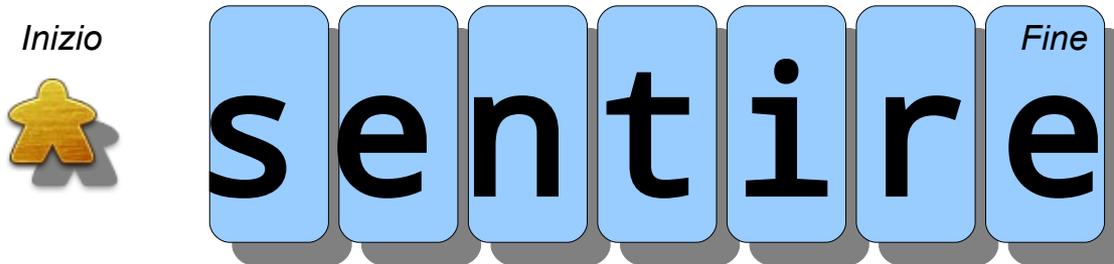
Inizio

 s e n t i r e Fine

Come funziona il matching

sentire. Gli Egiziani trasmisero ai Romani le prep


Carattere corrente

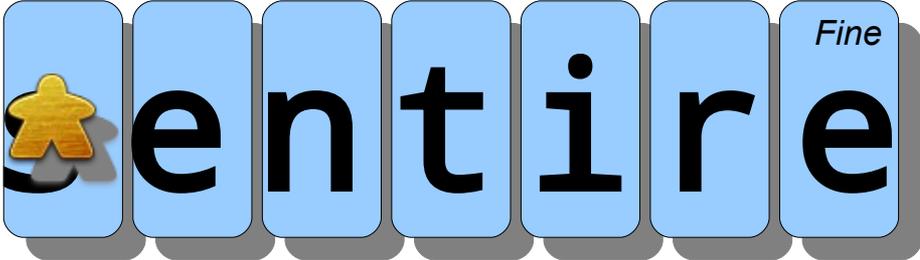


Come funziona il matching

Sentire. Gli Egiziani trasmisero ai Romani le prepa

Carattere corrente

Inizio

s e n t i r e

Fine

Come funziona il matching

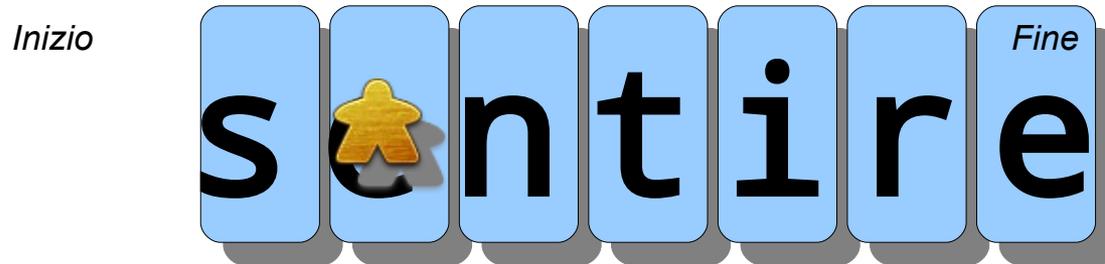
e
ntire. Gli Egiziani trasmisero ai Romani le prepar

Carattere corrente

Inizio

s e n t i r e

Fine



Come funziona il matching

n tire. Gli Egiziani trasmisero ai Romani le prepara


Carattere corrente

Inizio

s e n t i r e

Fine



Come funziona il matching

t ire. Gli Egiziani trasmisero ai Romani le preparaz

Carattere corrente

Inizio

s e n  i r e

Fine

Come funziona il matching

i re. Gli Egiziani trasmisero ai Romani le preparazi


Carattere corrente

Inizio

s e n t  r e

Fine

Come funziona il matching

r

e. Gli Egiziani trasmisero ai Romani le preparazio


Carattere corrente

Inizio

s e n t i  e

Fine

Come funziona il matching

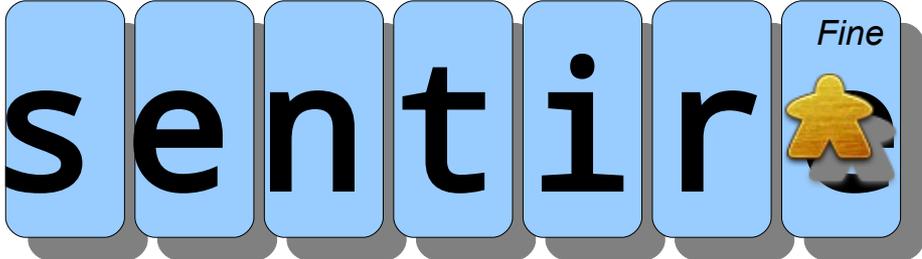
e . Gli Egiziani trasmisero ai Romani le preparazion

Carattere corrente

Inizio

s e n t i r o

Fine



Caratteri normali e speciali

- Alcuni caratteri assumono un significato speciale se preceduti da '\' (backslash)

\bmani\b

Delimitatore di
parola (spazi,
punteggiatura,...)

Altri caratteri speciali

<code>\b</code>	Delimitatore di una parola	} Non sono inclusi nel match!
<code>\B</code>	Inverso di <code>\b</code> (carattere che non è un delimitatore di parola)	
<code>\d</code>	Cifra	
<code>\D</code>	Inverso di <code>\d</code> (carattere che non è una cifra)	
<code>\s</code>	Spazio	
<code>\S</code>	Inverso di <code>\s</code> (tutto tranne uno spazio)	
<code>\w</code>	Carattere di una parola	
<code>\W</code>	Inverso di <code>\w</code>	
<code>\A</code>	Inizio del testo (!)	
<code>\Z</code>	Fine del testo (!)	

(!) non è riconosciuto da con kregexp

Delimitatore di parola '\b'

Cerco la parola 'mani'

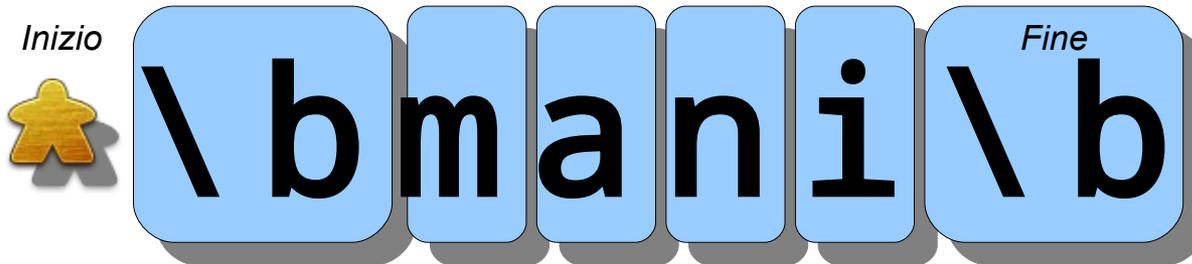


The screenshot shows the 'kregexpeditor' application window. The main text area contains a paragraph of Italian text with the word 'mani' highlighted in red. Above the text, three boxes are arranged: 'Word Boundary', 'mani', and 'Word Boundary'. The left sidebar shows 'Compound regular expressions:' with a tree view containing 'General', 'spaces', and 'User Defined'. At the bottom, the 'ASCII syntax:' field displays the regular expression `\bmani\b`. A 'Help' button is visible in the bottom left corner.

Come funziona il matching

t ra le mani, stavo a sentire. Gli Egiziani trasmisero

Carattere corrente

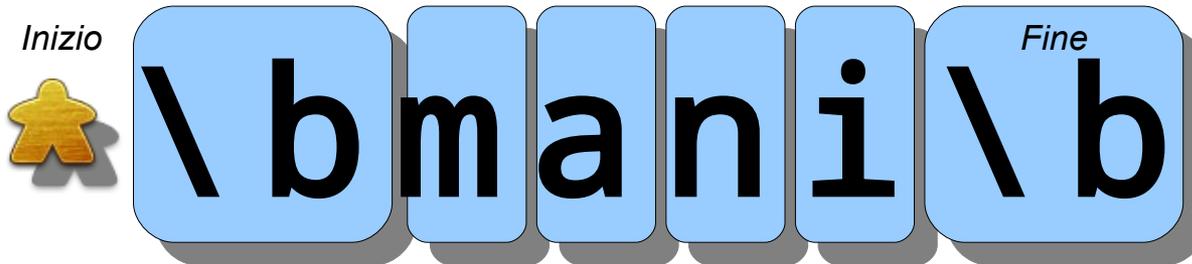


Come funziona il matching

r

a le mani, stavo a sentire. Gli Egiziani trasmisero

Carattere corrente

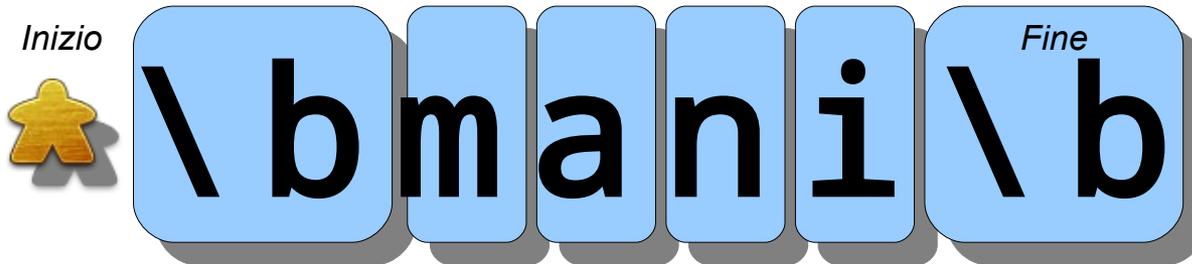


Come funziona il matching

a

le mani, stavo a sentire. Gli Egiziani trasmisero a


Carattere corrente



Come funziona il matching

le mani, stavo a sentire. Gli Egiziani trasmisero ai

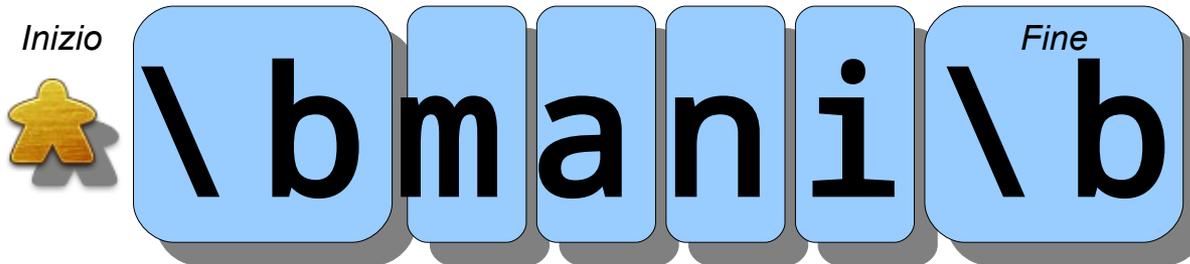

Carattere corrente



Come funziona il matching

l e mani, stavo a sentire. Gli Egiziani trasmisero ai

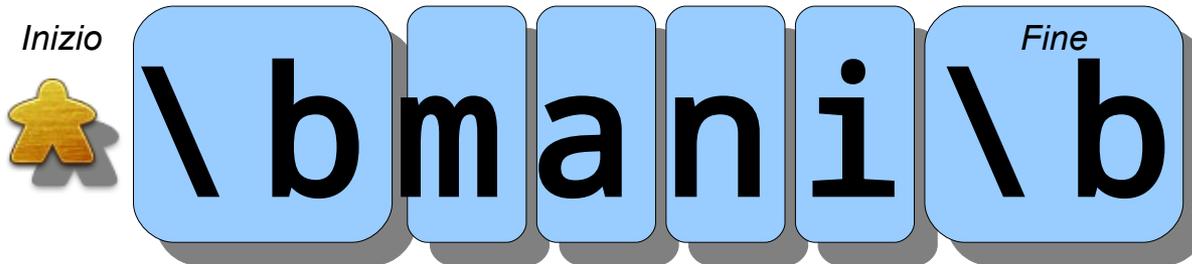

Carattere corrente



Come funziona il matching

e mani, stavo a sentire. Gli Egiziani trasmisero ai R

Carattere corrente



Come funziona il matching

mani, stavo a sentire. Gli Egiziani trasmisero ai Ro


Carattere corrente



Come funziona il matching

m

ani, stavo a sentire. Gli Egiziani trasmisero ai Rom

Carattere corrente

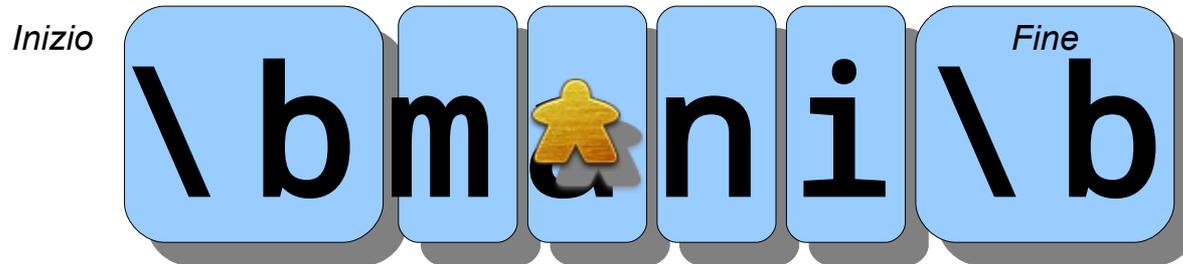


Come funziona il matching

a

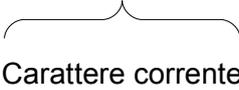
ni, stavo a sentire. Gli Egiziani trasmisero ai Roma

Carattere corrente



Come funziona il matching

n i, stavo a sentire. Gli Egiziani trasmisero ai Roman

 Carattere corrente



Come funziona il matching

i, stavo a sentire. Gli Egiziani trasmisero ai Romani

Carattere corrente



Come funziona il matching

 stavo a sentire. Gli Egiziani trasmisero ai Romani


Carattere corrente



Non-delimitatore di parola '\B'

Cerco le parole che finiscono con 'mani'



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a sidebar titled 'Compound regular expressions:' with three categories: 'General' (containing 'anything' and 'spaces'), 'General' (containing 'anything' and 'spaces'), and 'User Defined'. The main workspace contains a diagram illustrating the search pattern: a box labeled 'Non-word Boundary' is followed by the text 'mani', which is followed by a box labeled 'Word Boundary'. Below this diagram, a text block contains the following paragraph: 'Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai **Ro**mani**** le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma do**mani** possiamo esser poveri. E non si misero in cammino a mani vuote.' At the bottom of the window, there is a field labeled 'ASCII syntax:' containing the regular expression '\Bmani\b'. A 'Help' button is located in the bottom-left corner.

Cifre '\d'

Cerco una sequenza di due cifre



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a 'Compound regular expressions:' panel with a tree view containing 'General', 'anything', 'spaces', and 'User Defined'. The main editing area contains two 'One of Following Characters' boxes, each with '- A digit character' selected. Below this, a text block contains a paragraph of Italian text: 'Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.' At the bottom, the 'ASCII syntax:' field contains the regular expression '\d\d'. A 'Help' button is visible in the bottom-left corner.

Non-cifre '\D'

Cerco i caratteri che non sono cifre



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a sidebar titled 'Compound regular expressions:' with a tree view containing 'General' (with sub-items 'anything' and 'spaces') and 'User Defined'. The main workspace displays a search result for the regular expression '\D'. A text box contains the following text: 'Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformare le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.' The text is color-coded: '16 febbraio del' is blue, '1951,' is red, 'sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire.' is blue, 'Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformare le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli.' is blue, 'Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.' is blue. At the bottom left, there is a 'Help' button and an 'ASCII syntax:' checkbox which is checked. To the right of the checkbox, the regular expression '\D' is displayed in a large font.

Spazi '\s'

Cerco tutti gli spazi



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a 'Compound regular expressions:' panel with a tree view containing 'General', 'anything', 'spaces', and 'User Defined'. The main editing area contains a text box with the regular expression '\s'. Below the text box, there is a preview of the search results on a sample text: 'Il_16_febbraio_del_1951,_sedevo_su_un_panchettino_di_legno,_ai_suoi_piedi,_puntavo_i_gomiti_sulle_sue_ginocchia_e_con_la_faccia_appoggiata_tra_le_mani,_stavo_a_sentire._Gli_Egiziani_trasmisero_ai_Romani_le_preparazioni_che_permettevano_di_trasformar_le_fibre_vegetali_del_papiro_in_superfici_pulite_bianche_pieghevoli._Oggi_siamo_ricchi_ma_domani_possiamo_esser_poveri._E_non_si_misero_in_cammino_a_mani_vuote.' The text is underlined where spaces were found. At the bottom left, there is an 'ASCII syntax:' checkbox and a 'Help' button.

Non-spazi '\S'

Cerco tutti i caratteri che non sono spazi



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a sidebar titled 'Compound regular expressions:' with a tree view containing 'General' (with sub-items 'anything' and 'spaces') and 'User Defined'. The main workspace is divided into two sections. The top section is titled 'One of Following Characters' and contains the text '- A non-space character'. The bottom section displays a sample text with several words highlighted in blue and red, representing the results of a search for non-space characters. At the bottom left, there is a field labeled 'ASCII syntax:' containing the regular expression '\S', and a 'Help' button.

Caratteri di una parola '\w'

Cerco tutti i caratteri di una parola



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a sidebar titled 'Compound regular expressions:' with a tree view containing 'General' (anything, spaces), 'General' (anything, spaces), and 'User Defined'. The main area displays a search result for the regular expression '\w', showing a box labeled 'One of Following Characters - A word character'. Below this, a sample text is shown with words highlighted in red and blue. At the bottom, the 'ASCII syntax:' field contains the regular expression '\w', and a 'Help' button is visible.

Compound regular expressions:

- General
 - anything
 - spaces
- General
 - anything
 - spaces
- User Defined

One of Following Characters
- A word character

Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.

ASCII syntax: **\w**

Help

Caratteri che non sono di una parola '\W'

Cerco tutti i caratteri che non fanno parte di una parola



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a sidebar titled 'Compound regular expressions:' with three categories: 'General', 'General', and 'User Defined'. The first 'General' category is expanded, showing 'anything' and 'spaces'. The main workspace contains a search result box with the text: 'One of Following Characters - A non-word character'. Below this, a sample text is displayed with red underlines indicating matches: 'Il_16_febbraio_del_1951_sedevo_su_un_panchettino_di_legno_ai_suoi_piedi_puntavo_i_gomiti_sulle_sue_ginocchia_e_con_la_faccia_appoggiata_tra_le_mani_stavo_a_sentire_Gli_Egiziani_trasmisero_ai_Romani_le_preparazioni_che_permettevano_di_trasformar_le_fibre_vegetali_del_papiro_in_superfici_pulite_bianche_pieghevoli_Oggi_siamo_ricchi_ma_domani_possiamo_esser_poveri_E_non_si_misero_in_cammino_a_mani_vuote_'. At the bottom, there is a field labeled 'ASCII syntax:' containing the regular expression '\W'. A 'Help' button is located in the bottom-left corner.

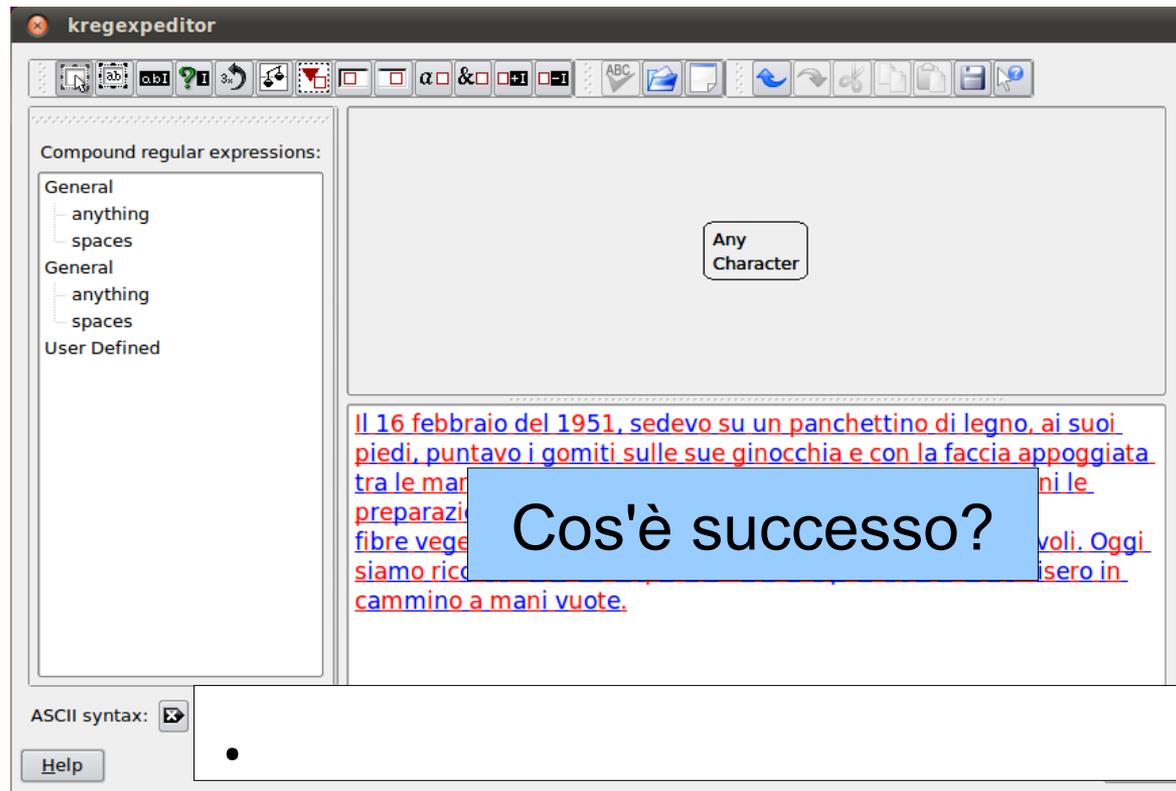
Ancora sul backslash

Cerco il carattere '.'



Ancora sul backslash

Cerco il carattere '.'



Caratteri speciali

- Alcuni caratteri sono “speciali” anche senza backslash
 - **Per poterli usare come caratteri normali bisogna anteporre un backslash '\'**

^	Inizio della riga
\$	Fine della riga
*	Zero o più (quantificatore)
+	Uno o più (quantificatore)
?	Zero o uno (quantificatore)
.	Qualsiasi carattere tranne ritorno a capo
()	Delimitatori di gruppo
{ }	Delimitatori per il numero di ripetizioni
[]	Insiemi di caratteri
	Alternativa

Attenzione! \$ non funziona correttamente in kregexpeditor

Inizio riga '^'

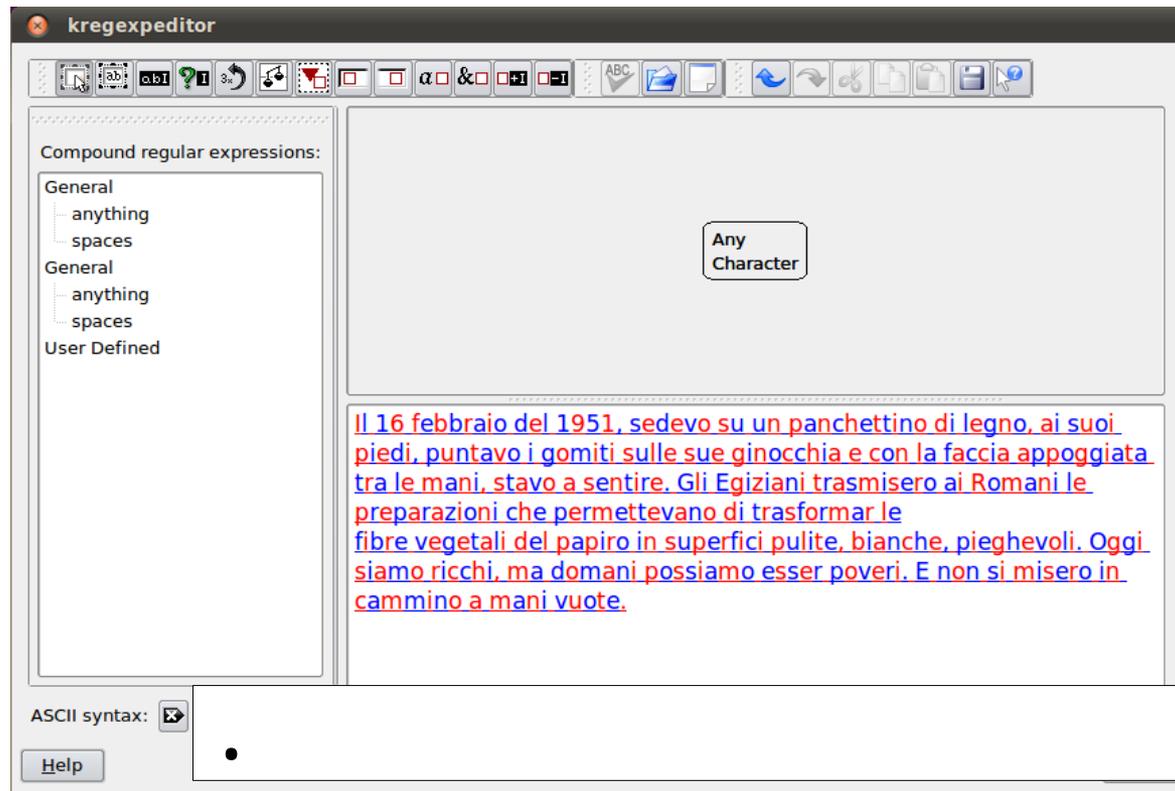
Cerco tutti i caratteri che sono all'inizio di una riga



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a sidebar titled 'Compound regular expressions:' with three categories: 'General' (containing 'anything' and 'spaces'), 'General' (containing 'anything' and 'spaces'), and 'User Defined'. The main area displays a search for '^' (Line Start) and a list of matches under the heading 'One of Following Characters - A word character'. The matches are: '16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire.', 'Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi siamo ricchi, ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.', and '16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire.' The bottom status bar shows 'ASCII syntax: ^ \w' and a 'Help' button.

Wildcard '.'

- Il carattere speciale/jolly '.' (punto) equivale a qualsiasi carattere (tranne il ritorno a capo '\n')



Wildcard '!

Cerco le parole di tre lettere che iniziano con 'c'

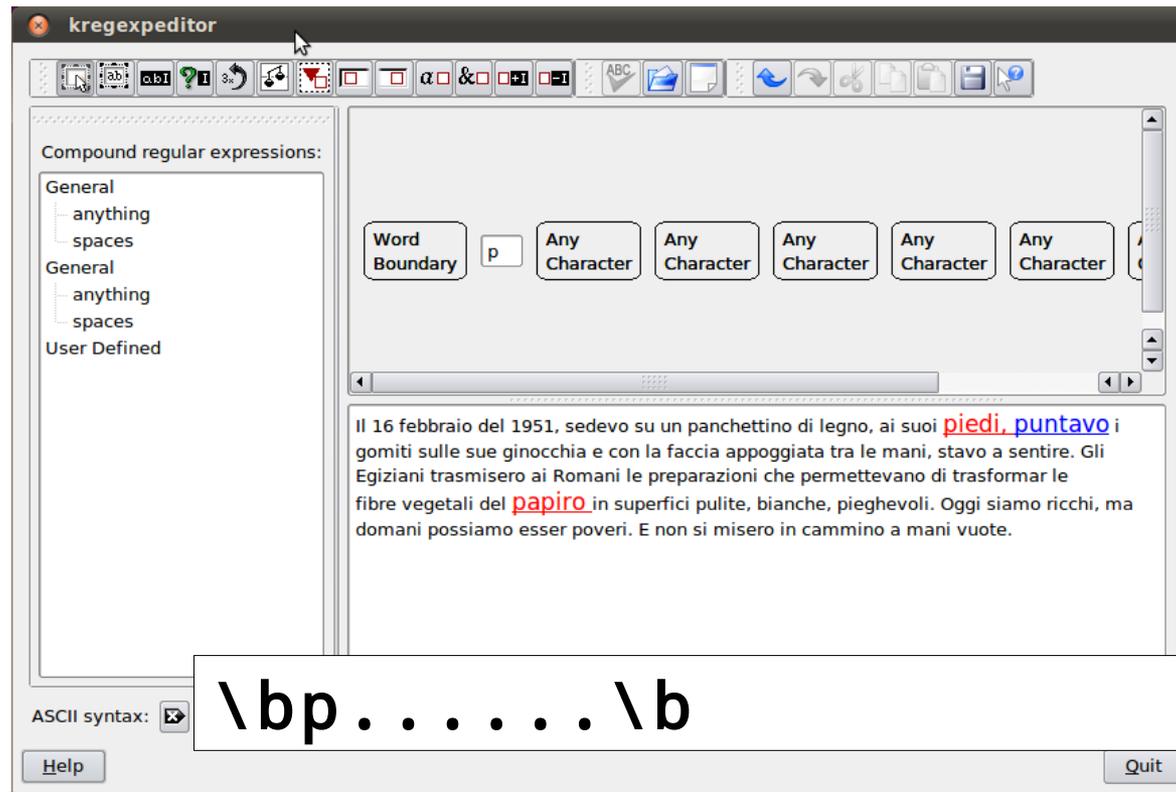


The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and searching. On the left, there is a 'Compound regular expressions:' panel with a tree view containing 'General', 'User Defined', and 'User Defined' sub-items. The main workspace displays a visual representation of the regular expression `\b c \b`, where `\b` is labeled 'Word Boundary' and `c` is labeled 'Any Character'. Below this, a text preview shows a paragraph of Italian text with the words 'con' and 'che' highlighted in red and blue respectively, demonstrating the search results. At the bottom, the 'ASCII syntax:' field shows the text `\bc.. \b`. A 'Help' button is located in the bottom-left corner.

Wildcard '!'

- In realtà l'esempio precedente non è molto robusto, es.

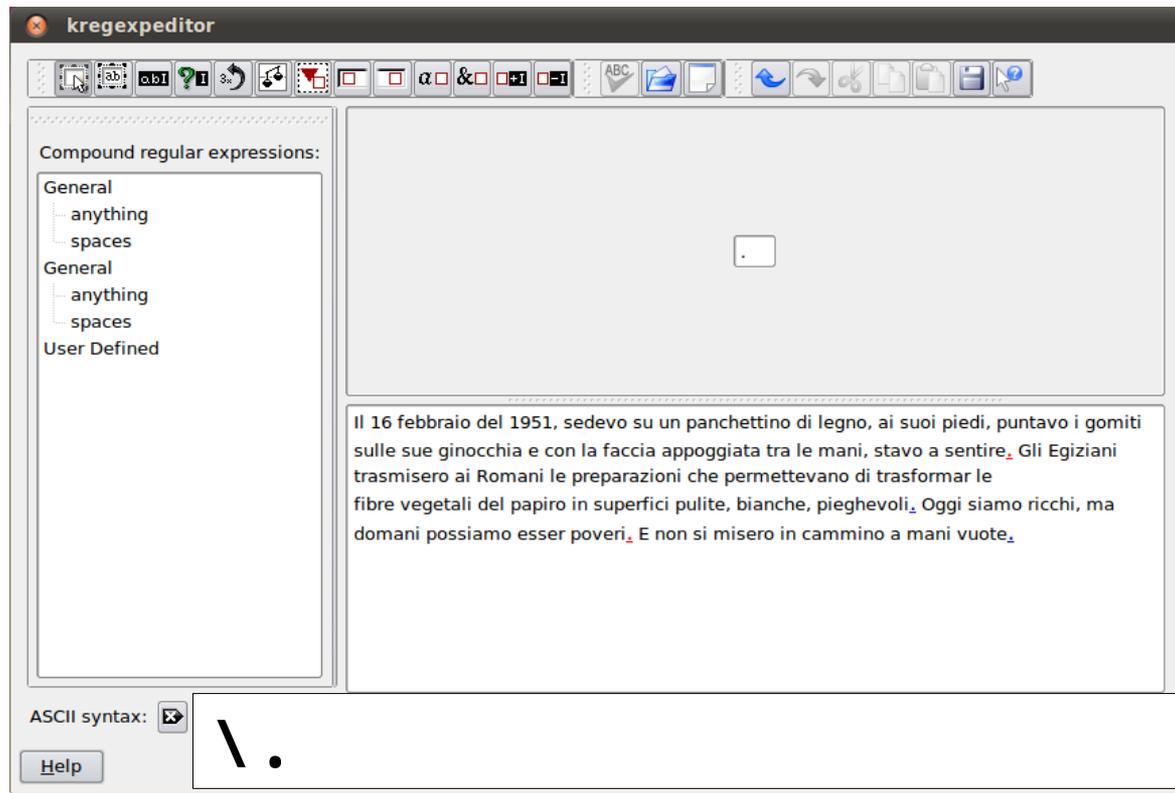
Cerco le parole di sette lettere che iniziano con 'p'



Wildcard '.'

Cerco il carattere '.'

Devo usare backslash perché '.' è un carattere speciale



Insiemi di caratteri []

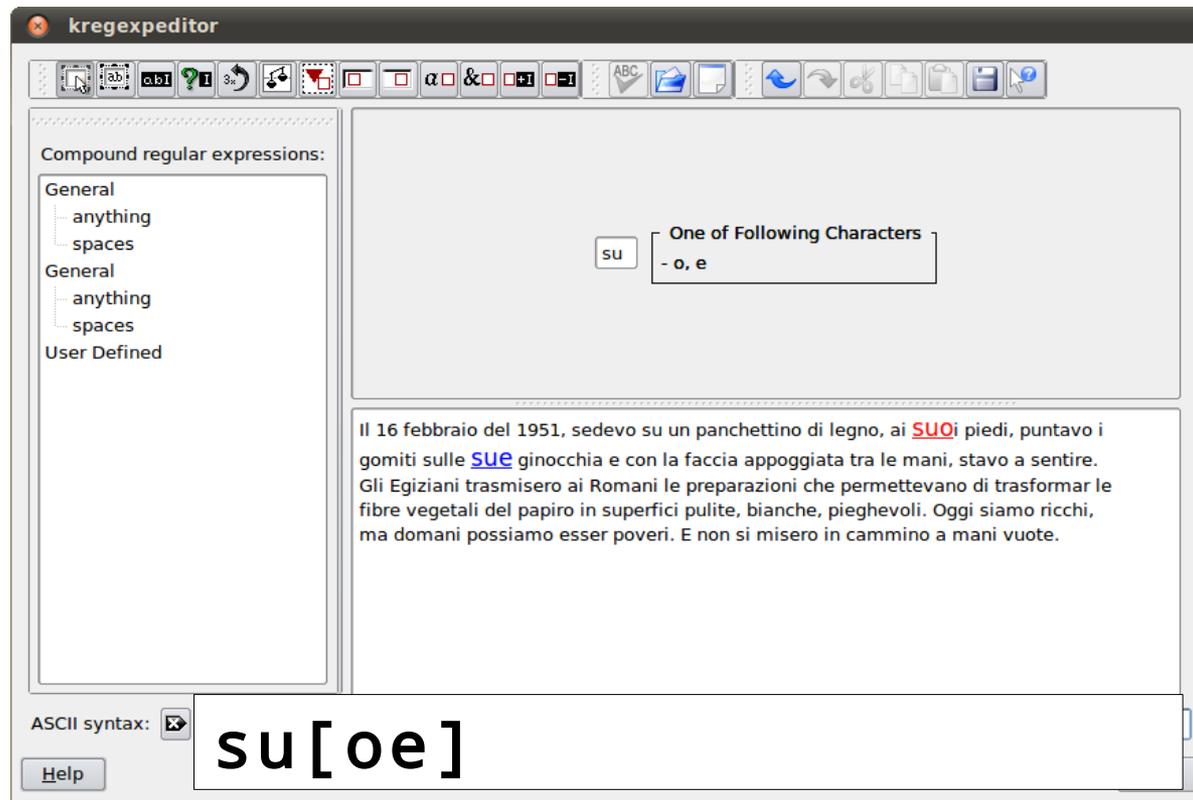
- Con **[]** possiamo specificare un carattere a scelta tra un insieme definito

su [oe]

Cerca la corrispondenza con 'suo' o 'sue'

Insiemi di caratteri []

- Con **[]** possiamo specificare un carattere a scelta tra un insieme definito



Insiemi di caratteri POSIX

- Alcuni insiemi di caratteri sono già definiti dallo standard POSIX
- Esempi:
 - **[[:alnum:]]** caratteri alfanumerici (equivale a [a-zA-Z0-9])
 - **[[:alpha:]]** caratteri alfabetici (equivale a [a-zA-Z])
 - **[[:digit:]]** cifre (equivale a [0-9])
 - **[[:blank:]]** spazi e tabulazioni
 - **[[:space:]]** tutti gli spazi e interruzioni di riga

Sequenza di caratteri [-]

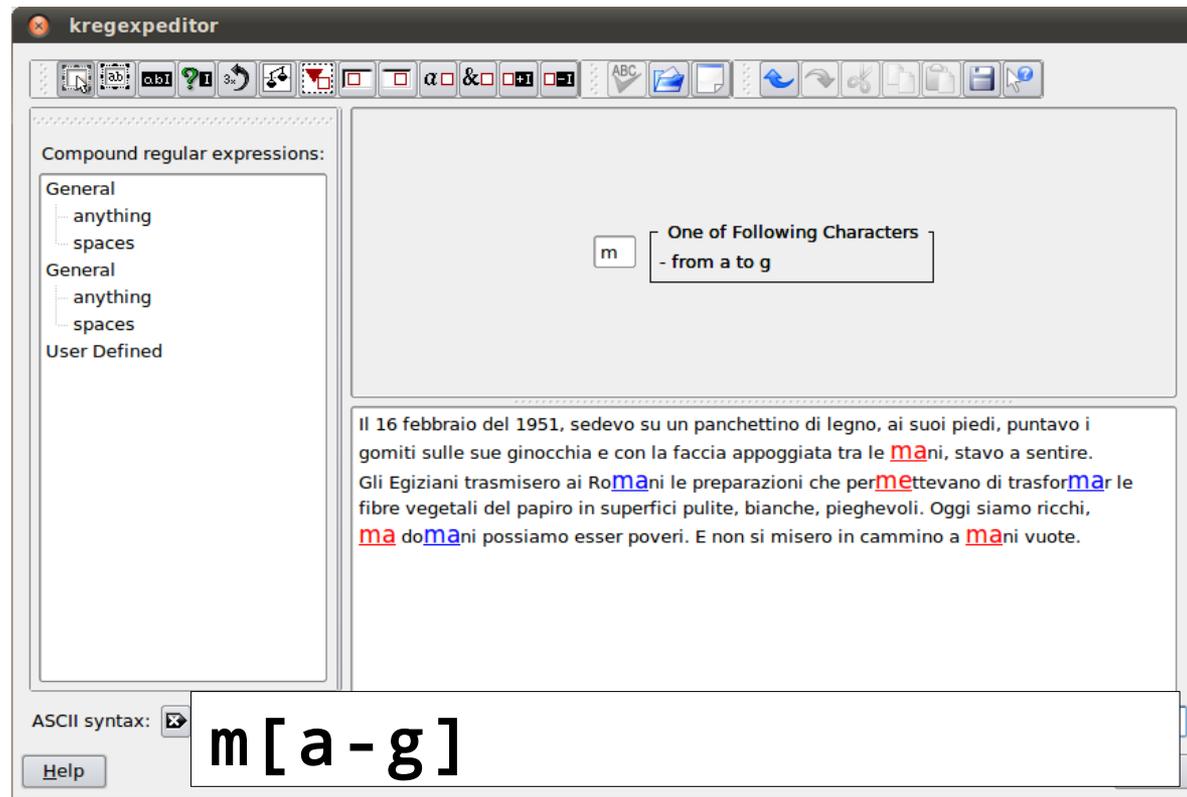
- Con **[-]** possiamo specificare un carattere a scelta tra una sequenza definita dal carattere iniziale e il carattere finale (secondo l'ordine ASCII, <http://en.wikipedia.org/wiki/ASCII>)

m [a - g]

Cerca la corrispondenza con 'm' seguita da un carattere da 'a' a 'g'

Sequenza di caratteri [-]

- Con **[-]** possiamo specificare un carattere a scelta tra una sequenza definita dal carattere iniziale e il carattere finale (secondo l'ordine ASCII, <http://en.wikipedia.org/wiki/ASCII>)



Nota: posso specificare anche più intervalli, p.es. [e-gs-z] (da 'e' a 'g' oppure da 's' a 'z')

Tutti i caratteri tranne... [^]

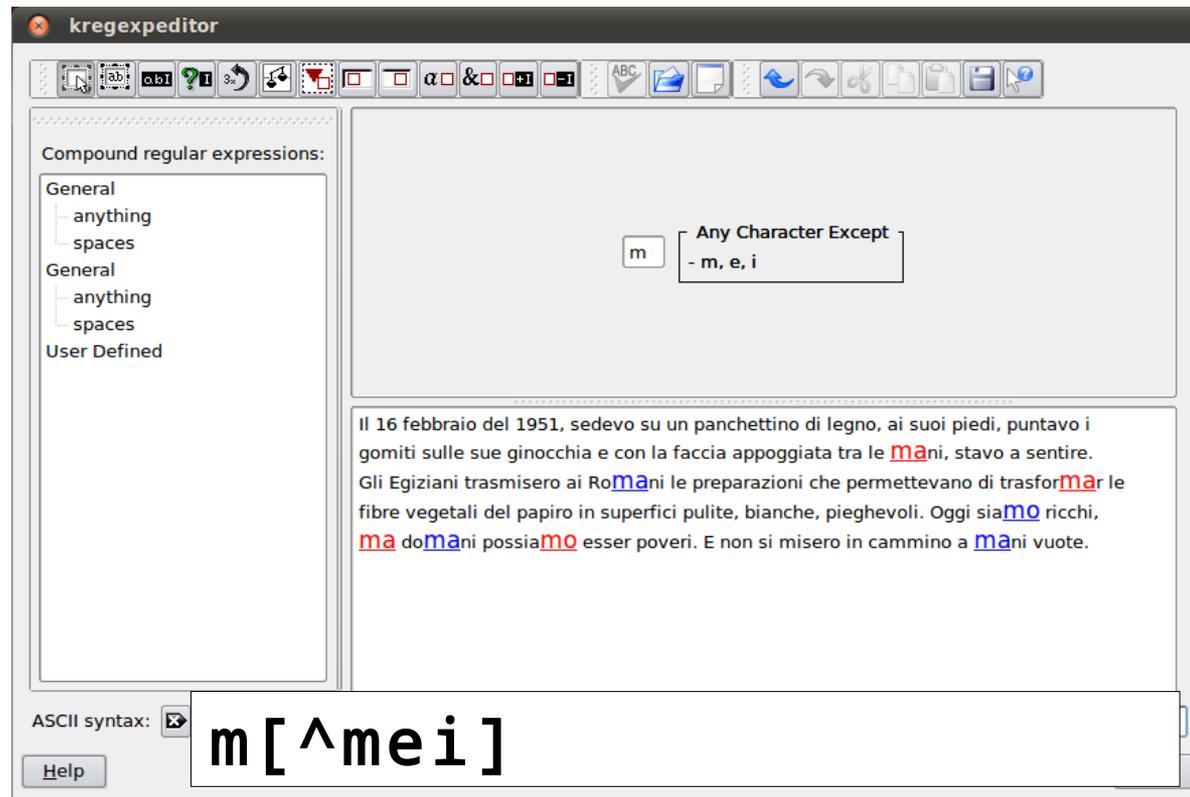
- Con [^] possiamo cercare la corrispondenza di tutti i caratteri escludendone alcuni

m [^ m e i]

Cerca la corrispondenza con 'm' seguita da un qualsiasi carattere tranne 'm', 'e', oppure 'i'

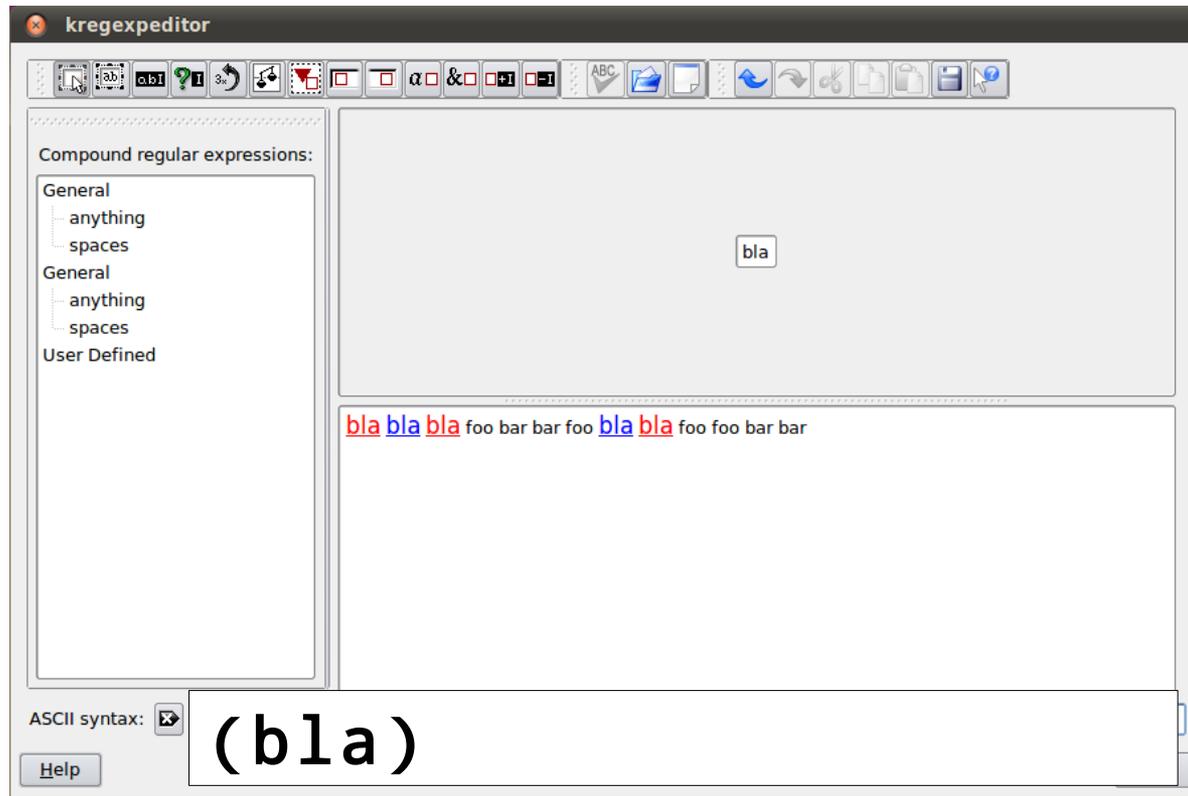
Tutti i caratteri tranne... [^]

- Con [^] possiamo cercare la corrispondenza di tutti i caratteri escludendone alcuni



Gruppi

- Con le parentesi () possiamo raggruppare più termini
 - Vedremo più avanti come recuperare i gruppi dopo aver trovato le corrispondenze



Alternative

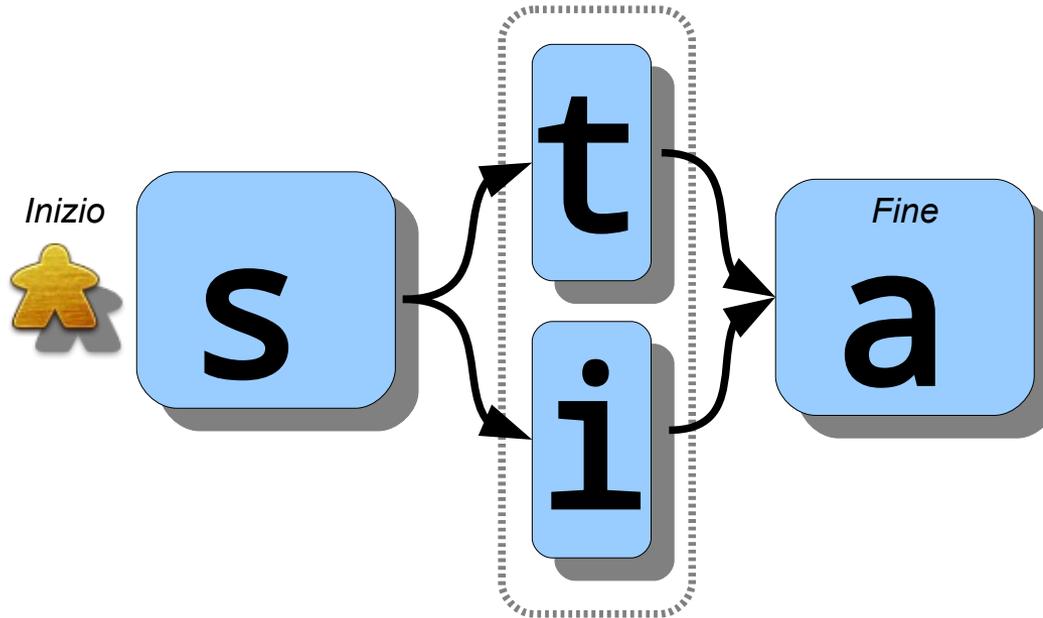
- Il carattere `|` ci permette di definire delle alternative

s (t | i) a

Cerca la corrispondenza con 's' seguita da 't'
oppure 'i', e poi da 'a'

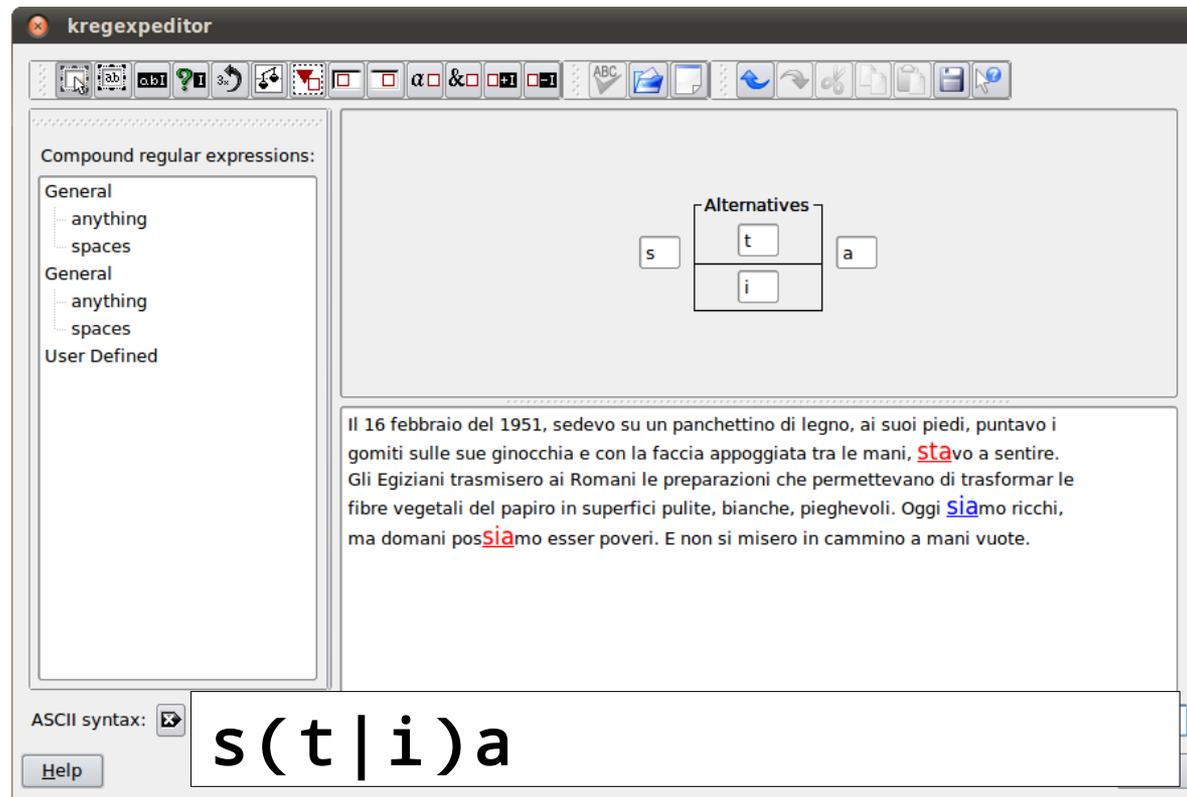
Alternative

- Il carattere | ci permette di definire delle alternative



Alternative

- Il carattere | ci permette di definire delle alternative



Compound regular expressions:

General

- anything
- spaces

General

- anything
- spaces

User Defined

Alternatives

s t a

i

Il 16 febbraio del 1951, sedevo su un panchettino di legno, ai suoi piedi, puntavo i gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, **st**avo a sentire. Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le fibre vegetali del papiro in superfici pulite, bianche, pieghevoli. Oggi **sia**mo ricchi, ma domani **pos**siamo esser poveri. E non si misero in cammino a mani vuote.

ASCII syntax: `s(t|i)a`

Help

Alternative

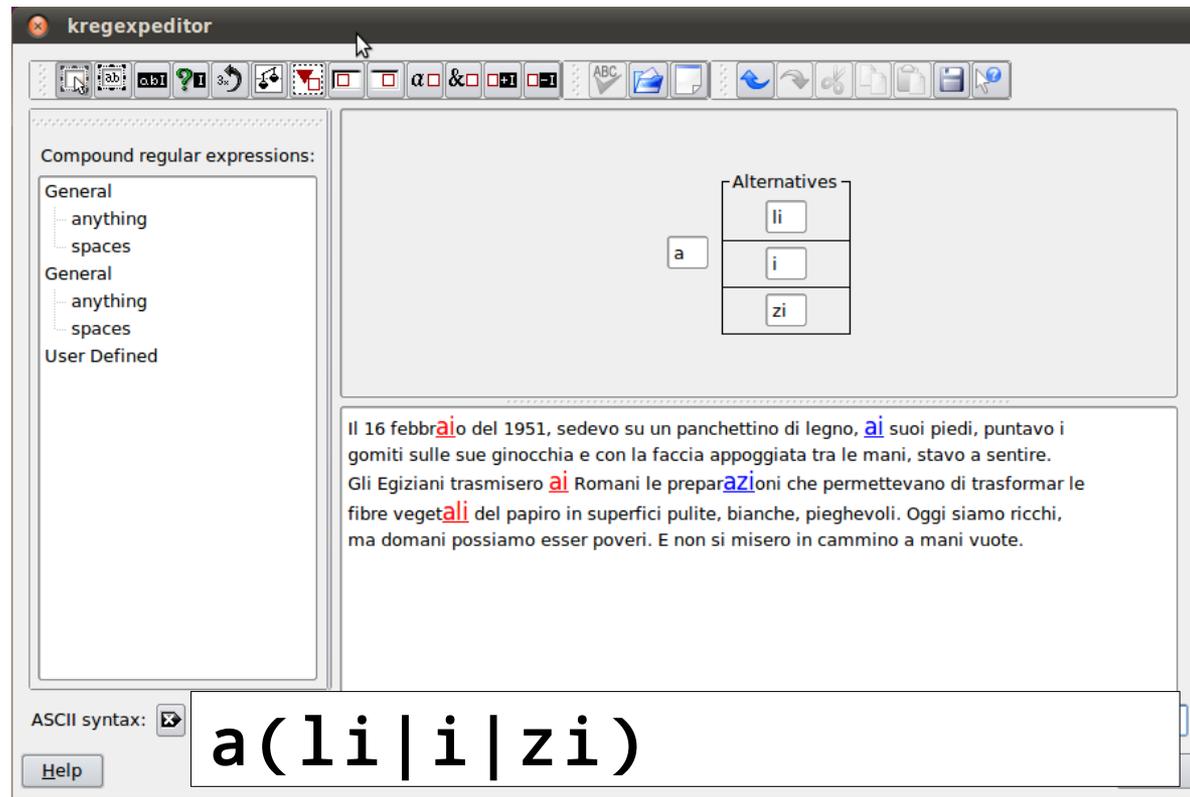
- Possiamo avere più alternative possibili

a (l i | i | z i)

Cerca la corrispondenza con 'a' seguita da 'l' e 'i'
oppure 'i' oppure 'z' e 'i'

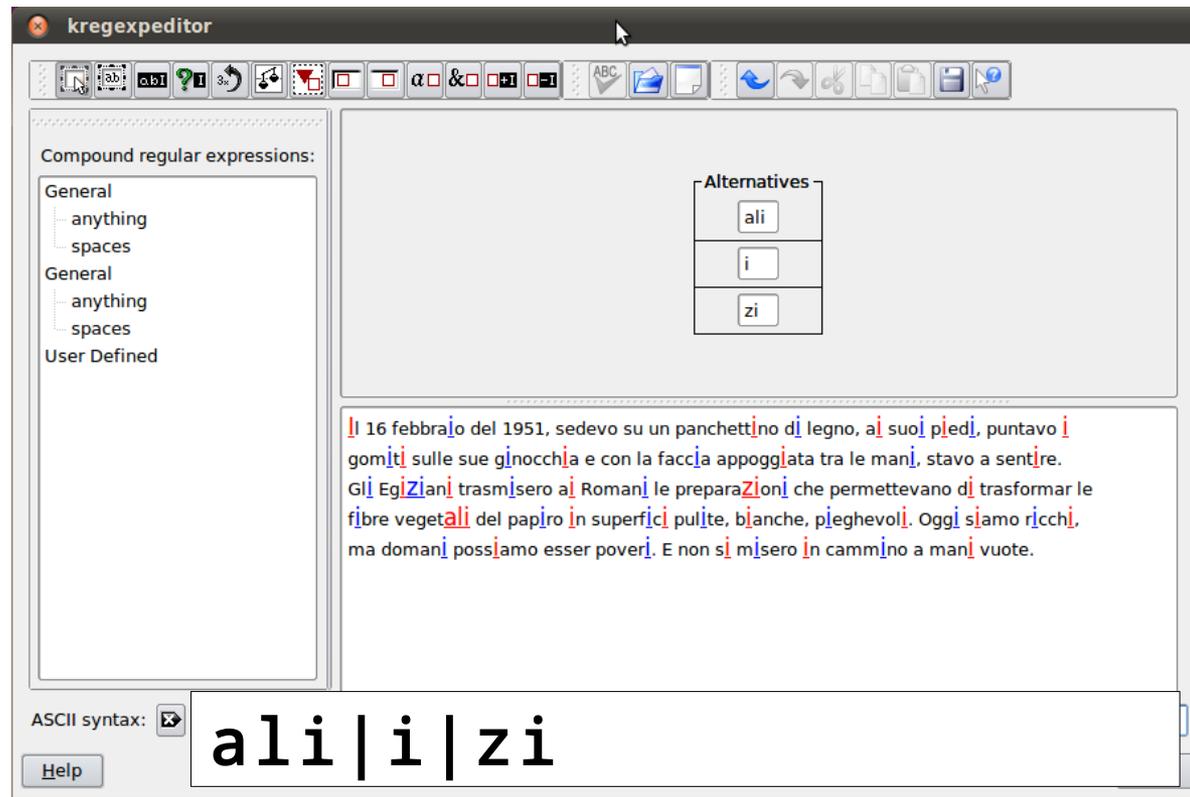
Alternative

- Possiamo avere più alternative possibili



Alternative

- Attenzione! Con le alternative bisogna utilizzare correttamente i gruppi!



Zero o più (*)

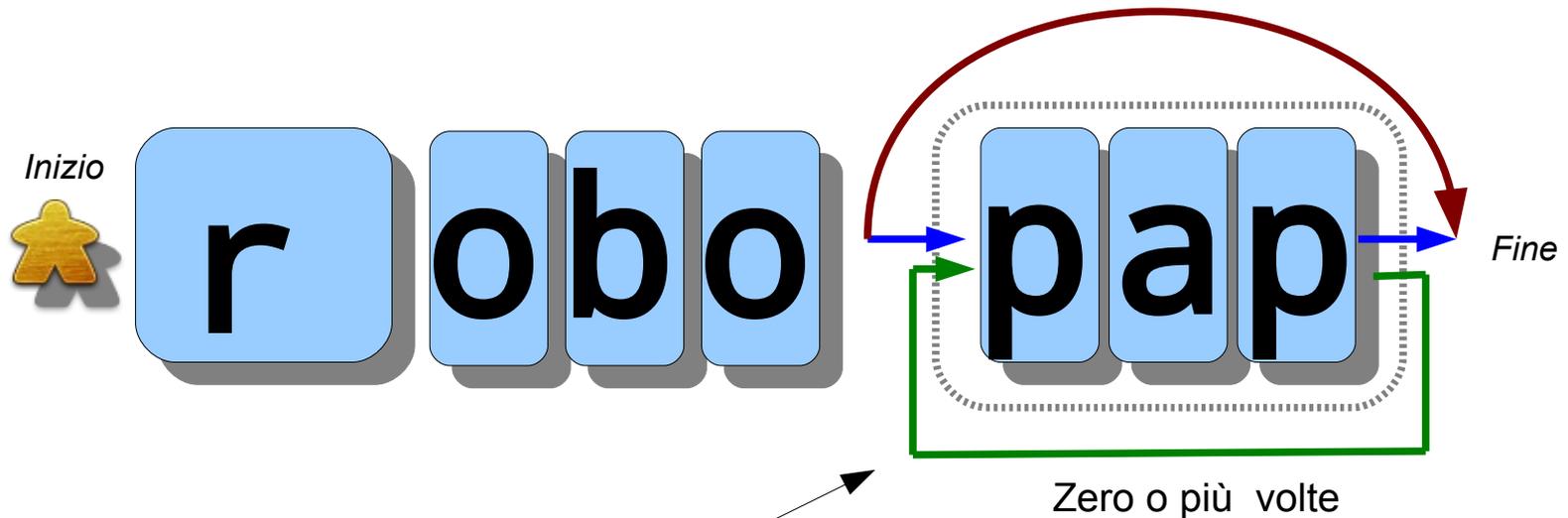
- Aggiungendo il quantificatore '*' possiamo specificare zero o più ripetizioni del termine (o gruppo) che precede

robo (pap)*

Cerca la corrispondenza le sequenze che contengono 'robo' seguito da zero o più ripetizioni di 'pap'

Zero o più (*)

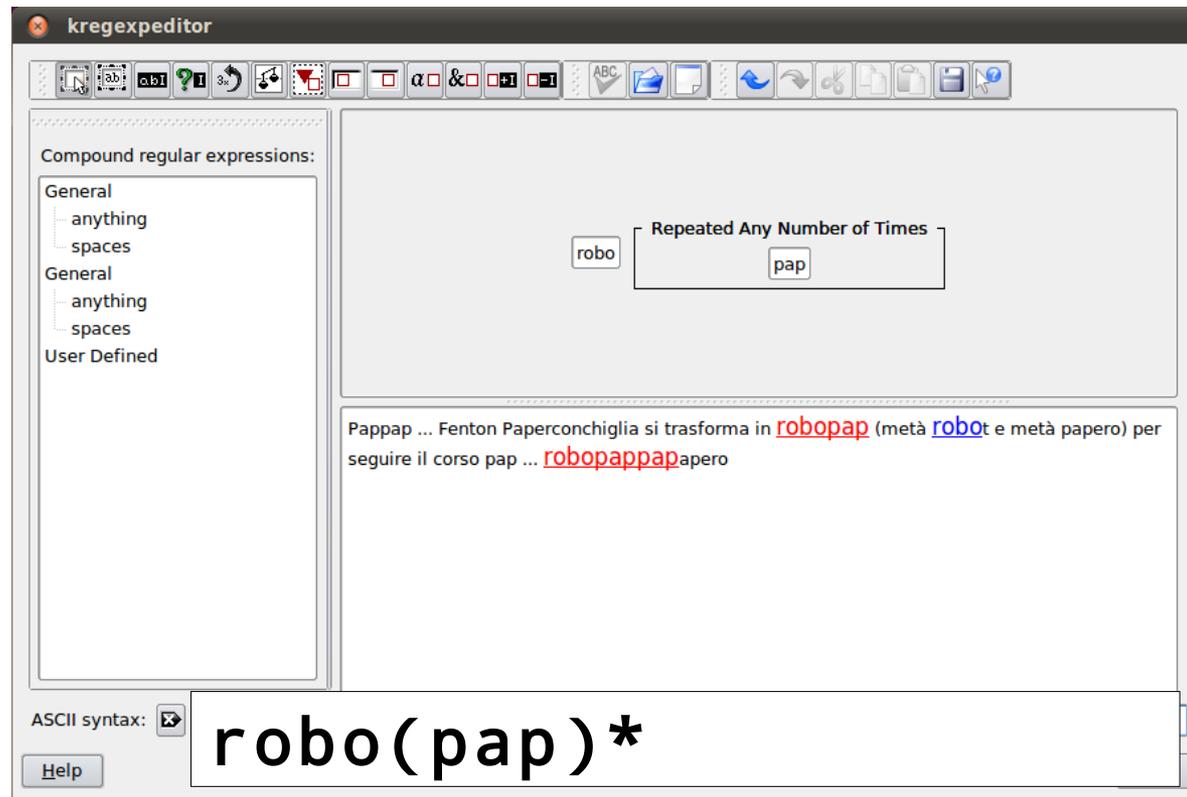
- Aggiungendo il quantificatore '*' possiamo specificare zero o più ripetizioni del termine (o gruppo) che precede



Se non riesco a completare una "ripetizioni", il match si ferma all'ultima volta che siamo arrivati alla Fine

Zero o più (*)

- Aggiungendo il quantificatore '*' possiamo specificare zero o più ripetizioni del termine (o gruppo) che precede



Uno o più (+)

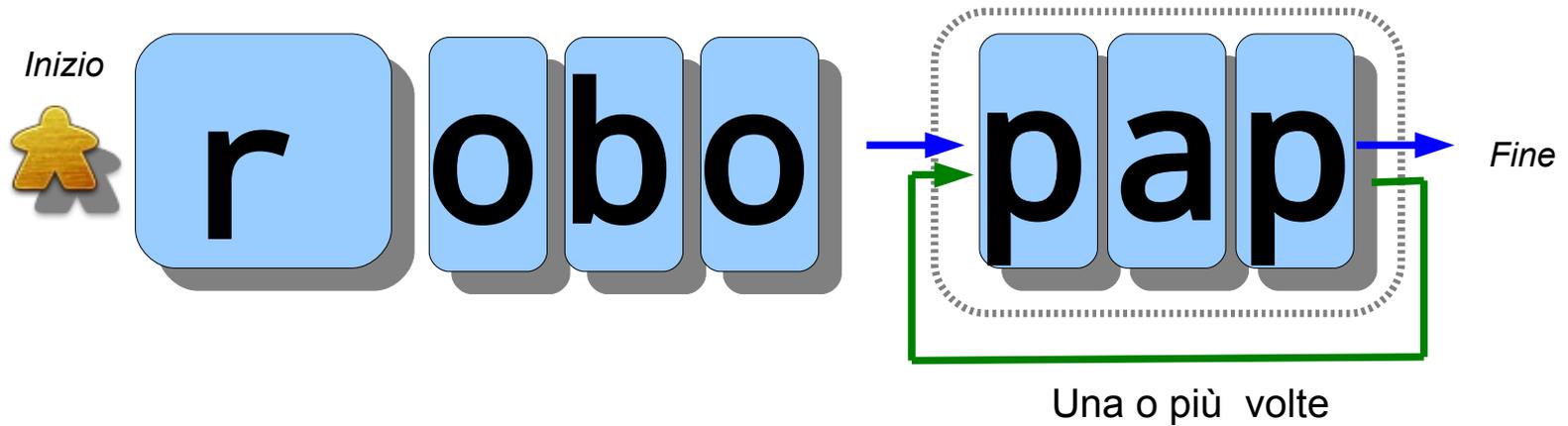
- Aggiungendo il quantificatore '+' possiamo specificare una o più ripetizioni del termine (o gruppo) che precede

robo (pap) +

Cerca la corrispondenza le sequenze che contengono 'robo' seguito da una o più ripetizioni di 'pap'

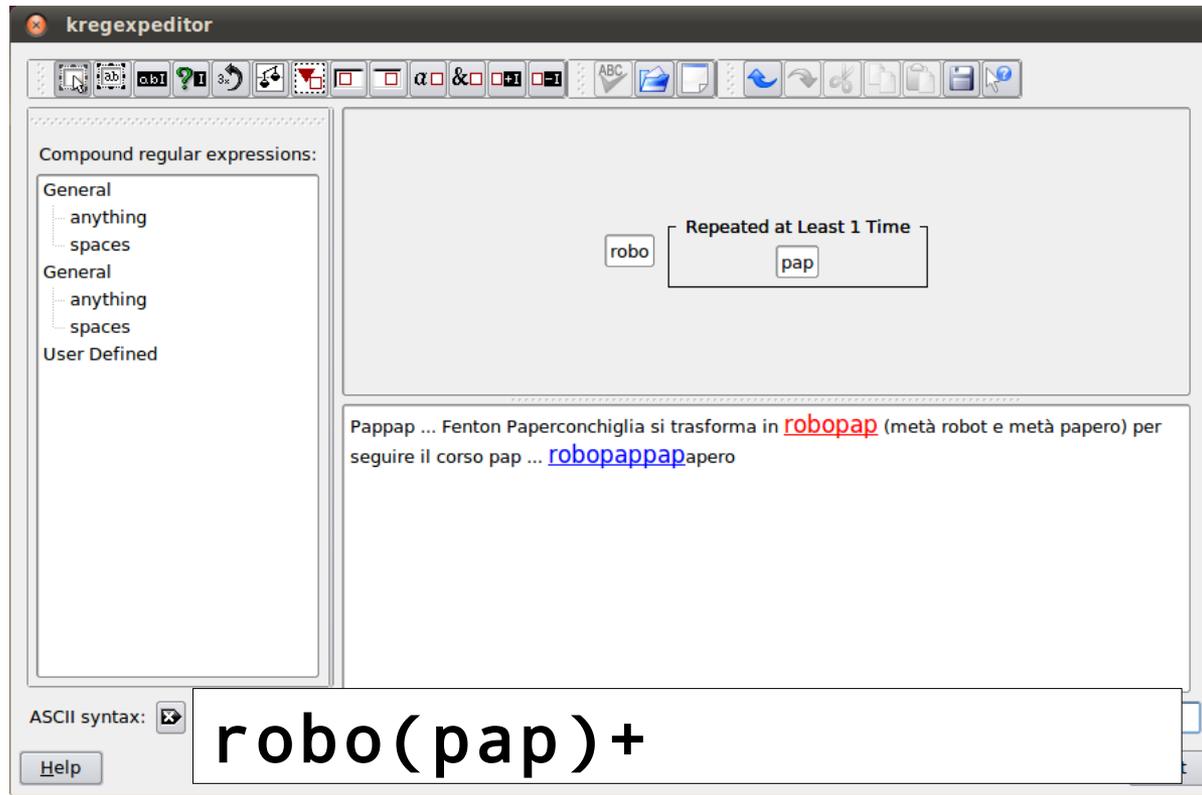
Uno o più (+)

- Aggiungendo il quantificatore '+' possiamo specificare una o più ripetizioni del termine (o gruppo) che precede



Uno o più (+)

- Aggiungendo il quantificatore '+' possiamo specificare una o più ripetizioni del termine (o gruppo) che precede



The screenshot shows the 'kregexpeditor' application window. The title bar reads 'kregexpeditor'. The interface includes a toolbar with various icons for editing and navigating. On the left, there is a sidebar titled 'Compound regular expressions:' with three categories: 'General' (containing 'anything' and 'spaces'), 'General' (containing 'anything' and 'spaces'), and 'User Defined'. The main workspace contains a visual representation of a regular expression: a box labeled 'robo' is followed by a box labeled 'Repeated at Least 1 Time' which contains a box labeled 'pap'. Below this workspace, there is a text area with the example sentence: 'Pappap ... Fenton Paperconchiglia si trasforma in **robopap** (metà robot e metà papero) per seguire il corso pap ... [robopappap](#)apero'. At the bottom, there is a text input field with the ASCII syntax: `robo(pap)+` and a 'Help' button.

Zero o uno (?)

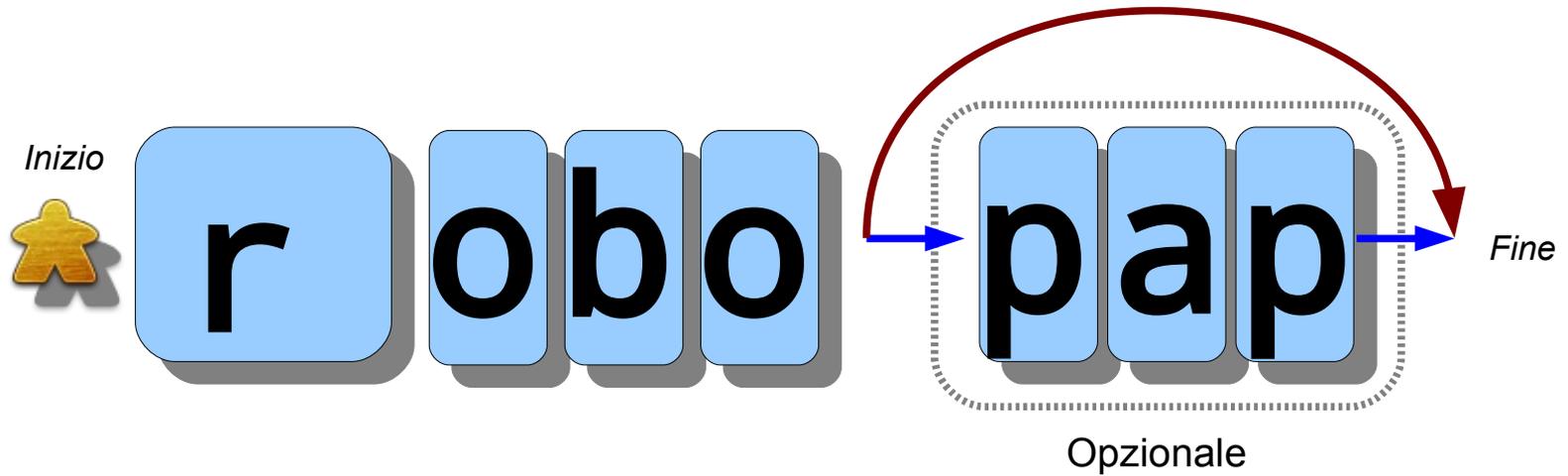
- Aggiungendo il quantificatore '?' possiamo specificare che il termine (o gruppo) che precede è opzionale

robo (pap) ?

Cerca la corrispondenza le sequenze che contengono 'robo' seguito opzionalmente da 'pap'

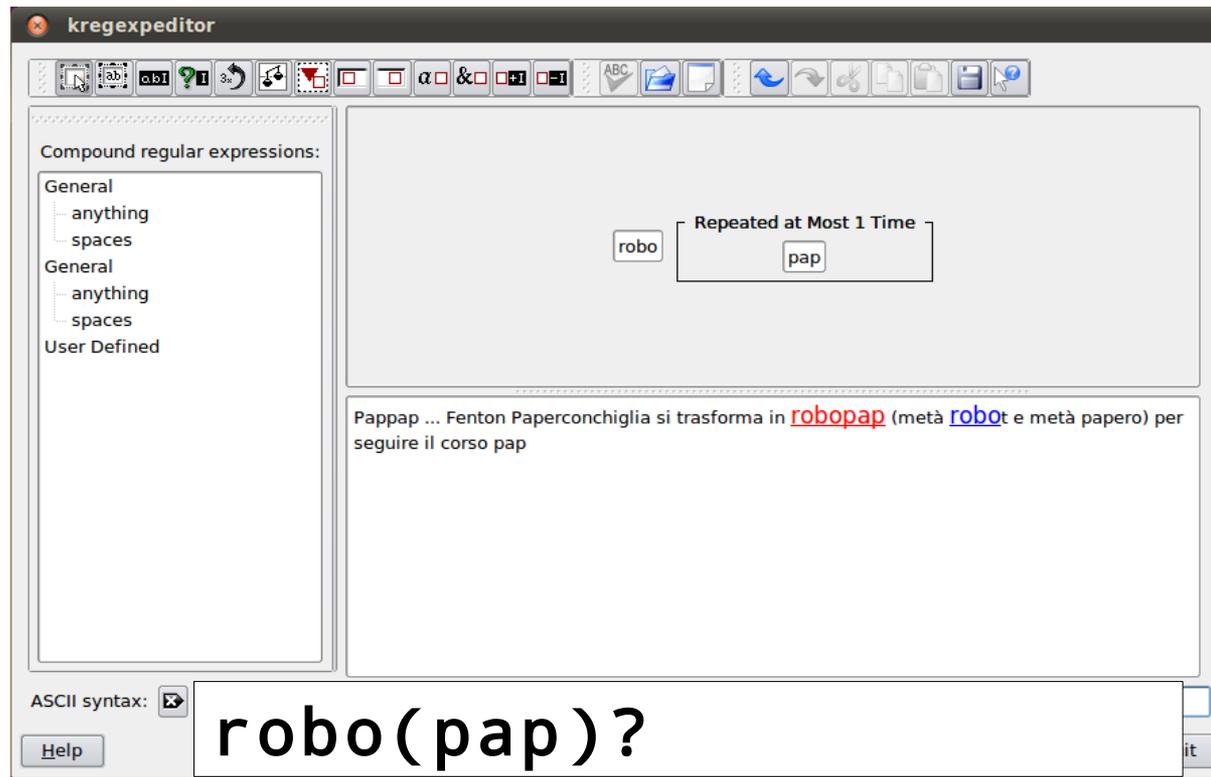
Zero o uno (?)

- Aggiungendo il quantificatore '?' possiamo specificare che il termine (o gruppo) che precede è opzionale



Zero o uno (?)

- Aggiungendo il quantificatore '?' possiamo specificare che il termine (o gruppo) che precede è opzionale



Numero di ripetizioni '{n}', '{n,m}', {n,}

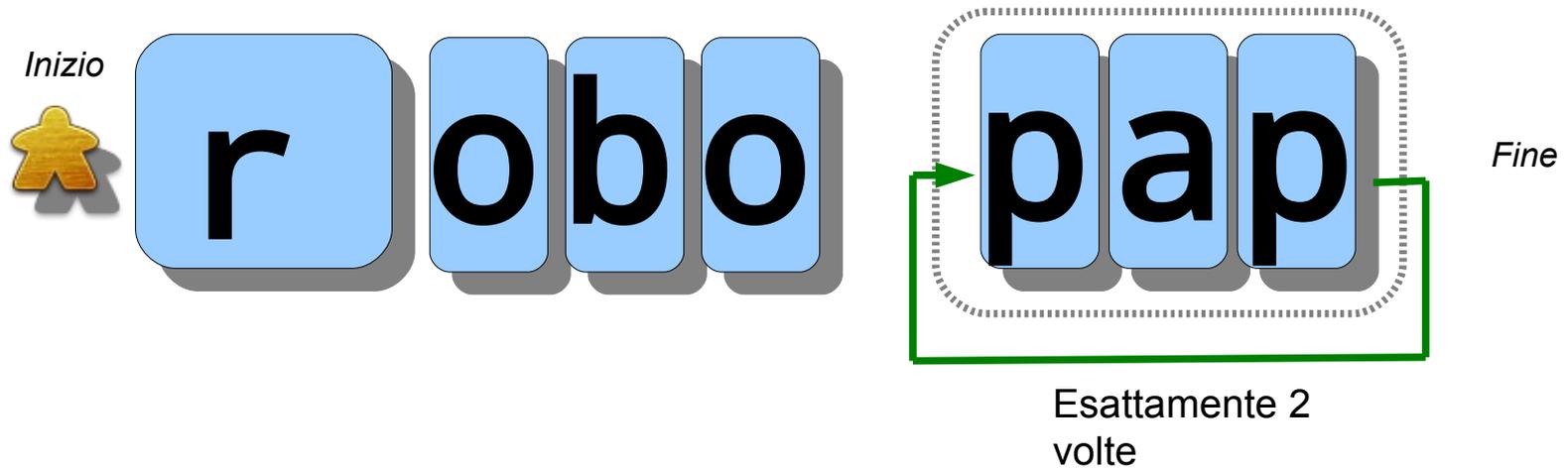
- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'

robo (pap) {2}

Cerca la corrispondenza le sequenze che contengono 'robo' seguito esattamente due ripetizioni di 'pap'

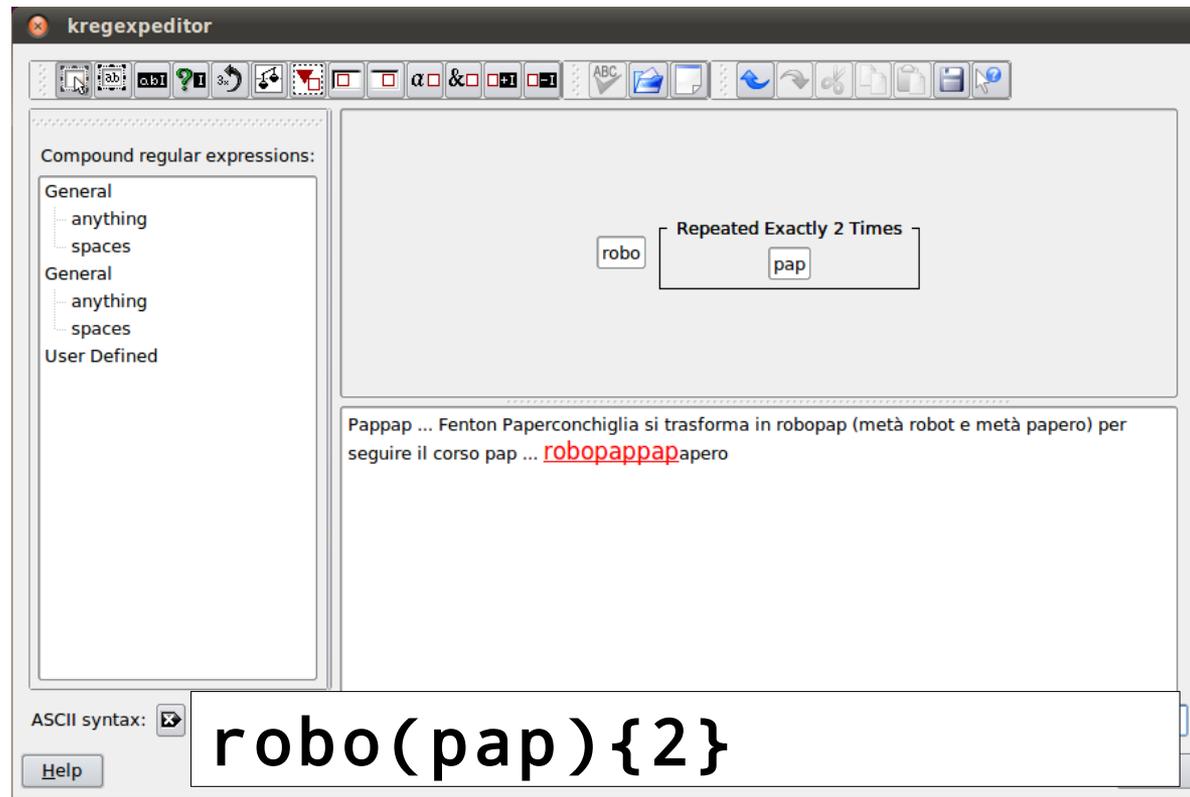
Numero di ripetizioni '{n}', '{n,m}', {n,}

- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'



Numero di ripetizioni '{n}', '{n,m}', {n,}

- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'



Numero di ripetizioni '{n}', '{n,m}', {n,}

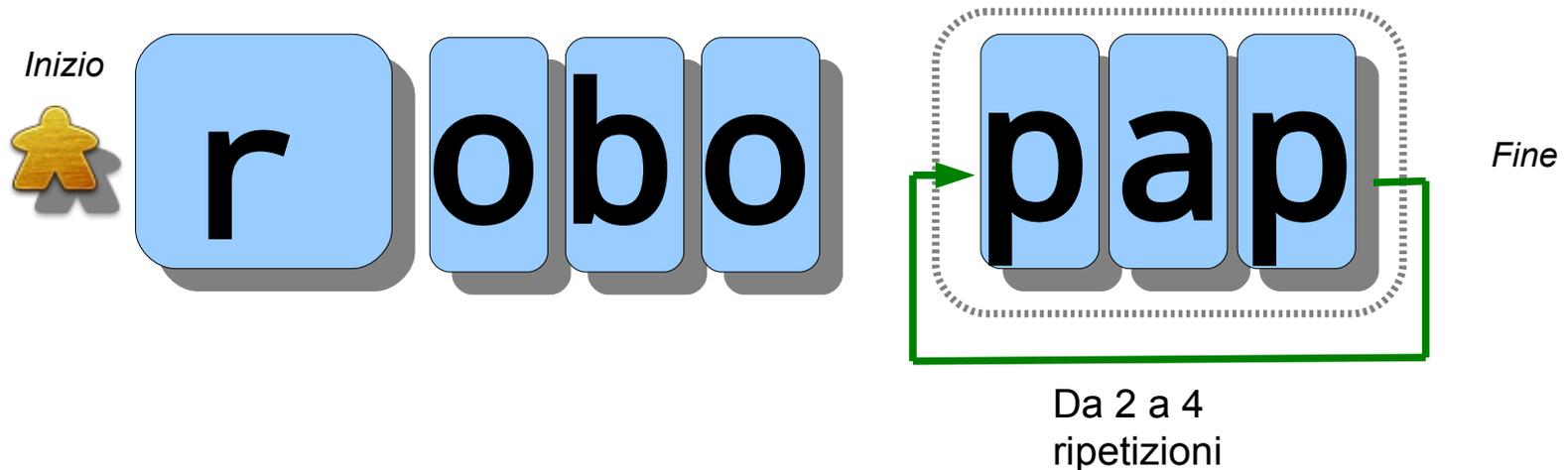
- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'

robo (pap) {2,4}

Cerca la corrispondenza le sequenze che contengono 'robo' seguito da 2 a 4 ripetizioni di 'pap'

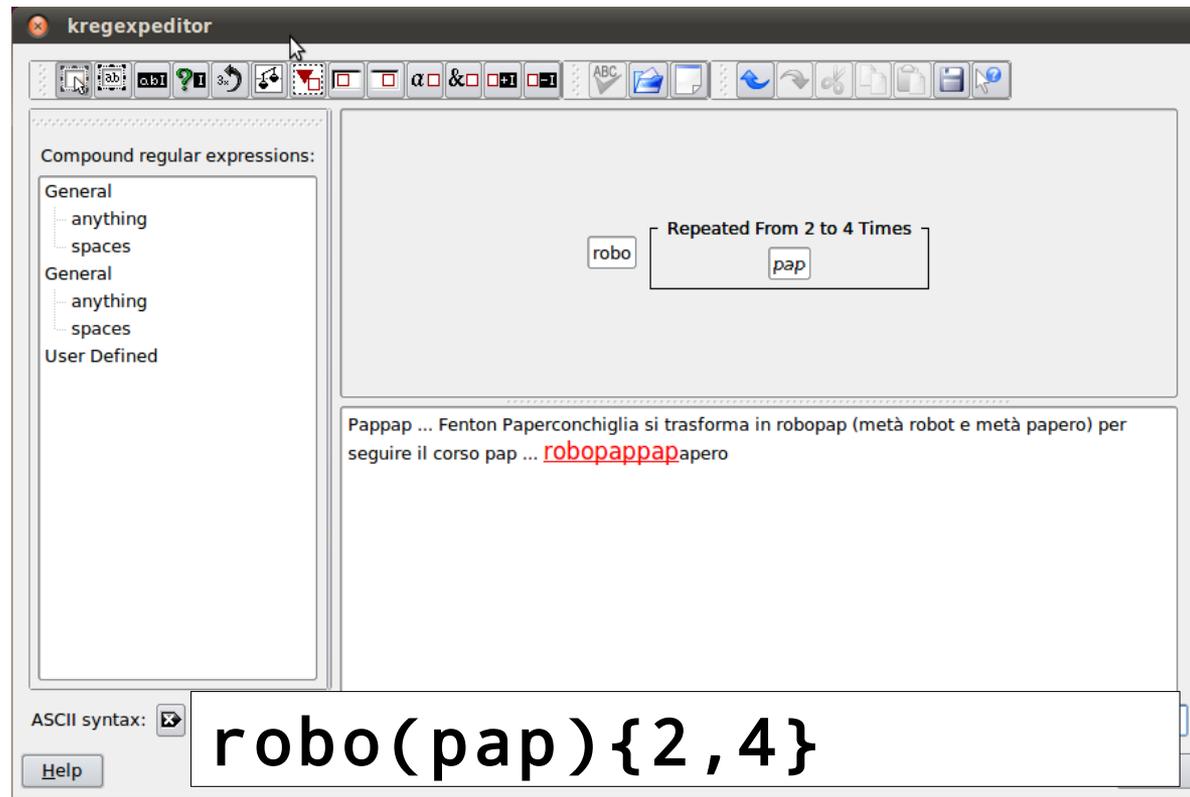
Numero di ripetizioni '{n}', '{n,m}', {n,}

- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'



Numero di ripetizioni '{n}', '{n,m}', {n,}

- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'



Numero di ripetizioni '{n}', '{n,m}', {n,}

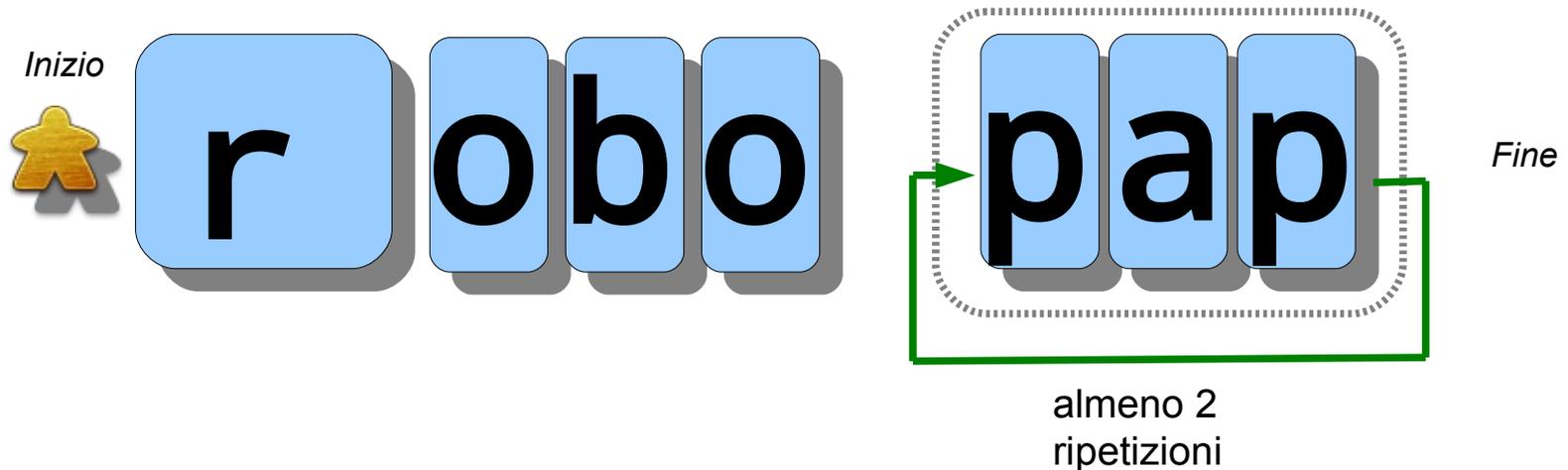
- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'

robo(pap){2,}

Cerca la corrispondenza le sequenze che contengono 'robo' seguito da almeno 2 ripetizioni di 'pap'

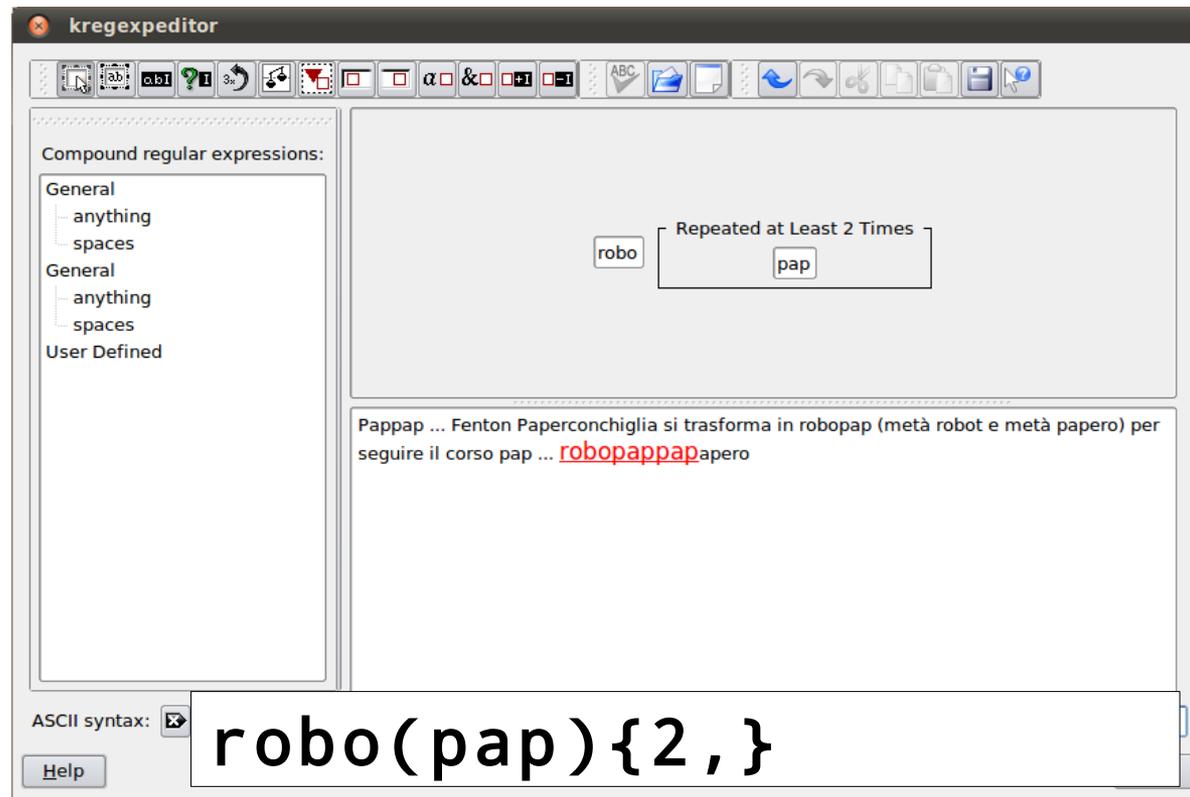
Numero di ripetizioni '{n}', '{n,m}', {n,}

- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'



Numero di ripetizioni '{n}', '{n,m}', {n,}

- Posso anche specificare un numero esatto di ripetizioni '{n}', un intervallo '{n,m}', o ripetizioni di almeno un numero specificato di volte '{n,}' o fino a un massimo di volte '{,n}'



Greediness

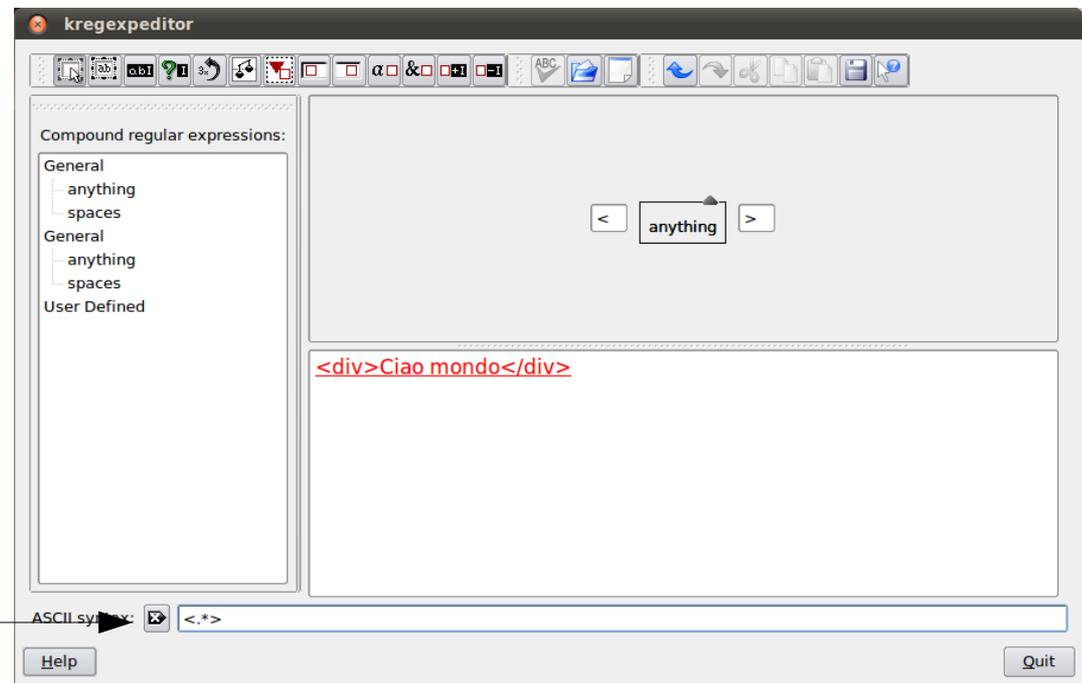
- Quando usiamo i quantificatori il motore di ricerca cerca di ripetere il maggior numero di volte possibile
 - Modalità **greedy** (ingorda)
- Questo può portare a degli errori, per esempio:



Voglio trovare tutti i tag <...>

Il termine '.'
ripetuto si
“mangia” tutto
quello che trova

< .* >



Laziness e back-tracking

- Se il motore delle regex lo supporta, è possibile limitare le ripetizioni al numero minimo possibile, aggiungendo '?' dopo il quantificatore
 - Questo meccanismo rallenta sensibilmente il processo di matching, perché il motore deve fare *back-tracking*: ritornare indietro e provare un'altro cammino

<.*?>

Voglio trovare tutti i tag <...>

The Regex Coach

File Autoscroll Help

Regular expression:

<.*?>

Target string:

<div>Ciao mondo</div>

Match from 0 to 5.

Control Info Tree Replace Split Step

Highlight (grey background):

selection 1 2 3 4 5

nothing 6 7 8 9 10

Scan from 0 Start of string: 0 End of string: -

<< >> < >

Attenzione! Questa funzione non è supportata da kregexpeditor

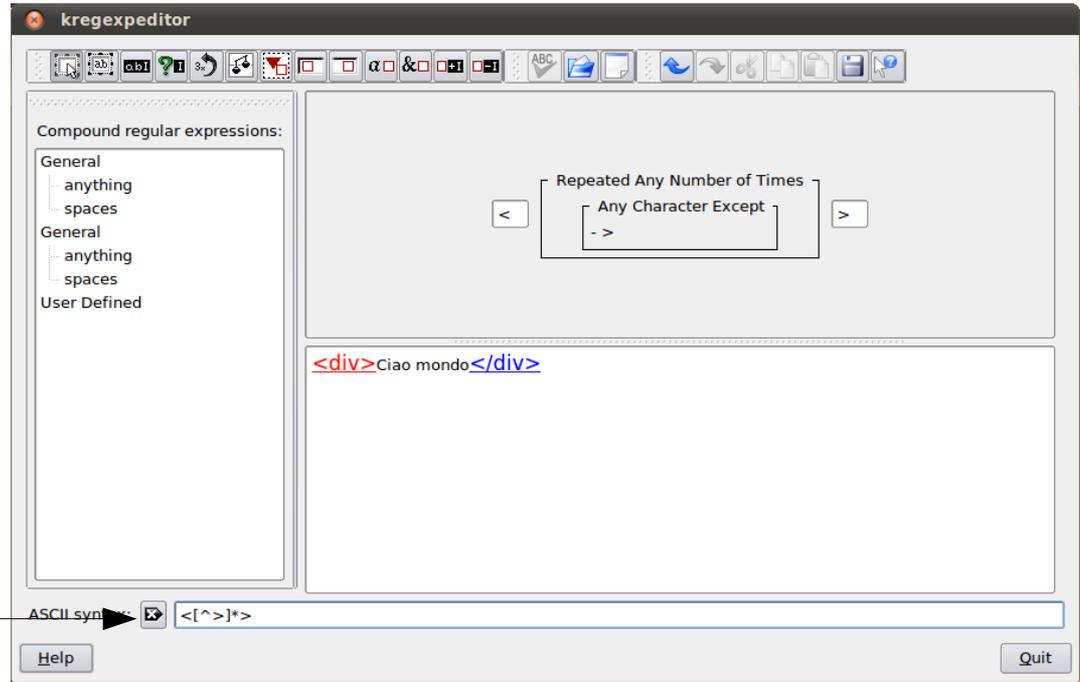
Alternativa a laziness...

- Per evitare i problemi di performance, o per ovviare alla mancanza della “laziness” nel motore utilizzato a volte è possibile utilizzare degli insiemi di negazione



Voglio trovare tutti i tag <...>

< [^ >] * >



Un esempio pratico

Voglio estrarre gli indirizzi email...



Transport of email across the Internet uses the Simple Mail Transfer Protocol (SMTP), which is defined in Internet standards RFC 5321 and RFC 5322, while mailboxes are most often accessed with the Post Office Protocol (POP) and the Internet Message Access Protocol (IMAP).

Email addresses, such as `jsmith@example.org`, have two parts. The part before the `@` sign is the local-part of the address, often the username of the recipient (`jsmith`), and the part after the `@` sign is a domain name to which the email message will be sent (`example.org`).

For more information contact `neo@matrix.eu` or `jeff.jeffty@jeff.co.uk`

Un esempio pratico

Voglio estrarre gli indirizzi email...

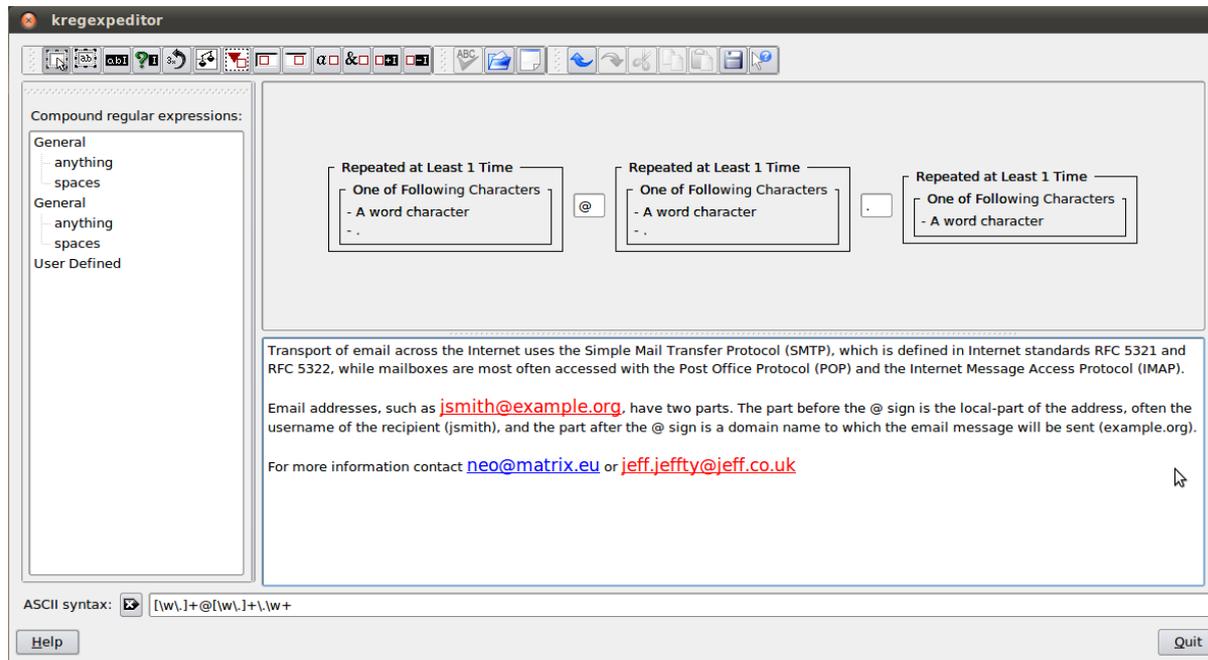


[\w \.] + @ [\w \.] + \. \w +

Un esempio pratico

Voglio estrarre gli indirizzi email...

[\w \.] + @ [\w \.]



Questa soluzione è corretta... ma obbliga il motore a fare back-tracking... sapete individuare dove?

Un esempio pratico

Voglio estrarre gli indirizzi email...



[\w \ .] + @ [\w \ .] + \ . \w +

A regular expression is shown in bold black text: `[\w \ .] + @ [\w \ .] + \ . \w +`. The second part, `[\w \ .] + \ .`, is highlighted in red. A white curly bracket is drawn underneath this red portion, pointing downwards.

Se dopo la ripetizione di `[\w \ .] +` non viene trovato il carattere '.', il motore torna indietro di un carattere nell'input e riprova. Se ancora non funziona, si torna indietro ancora di un carattere, e così via,...

Un esempio pratico

Voglio estrarre gli indirizzi email...



Soluzione senza back-tracking

[\w \ .] + @ [\w \ .] +

Un esempio pratico

Voglio estrarre gli indirizzi email...



Soluzione senza back-tracking

Compound regular expressions:

- General
 - anything
 - spaces
- General
 - anything
 - spaces
- User Defined

Repeated at Least 1 Time

One of Following Characters

- A word character
- .

@

Repeated at Least 1 Time

One of Following Characters

- A word character
- .

Transport of email across the Internet uses the Simple Mail Transfer Protocol (SMTP), which is defined in Internet standards RFC 5321 and RFC 5322, while mailboxes are most often accessed with the Post Office Protocol (POP) and the Internet Message Access Protocol (IMAP).

Email addresses, such as jsmith@example.org, have two parts. The part before the @ sign is the local-part of the address, often the username of the recipient (jsmith), and the part after the @ sign is a domain name to which the email message will be sent (example.org).

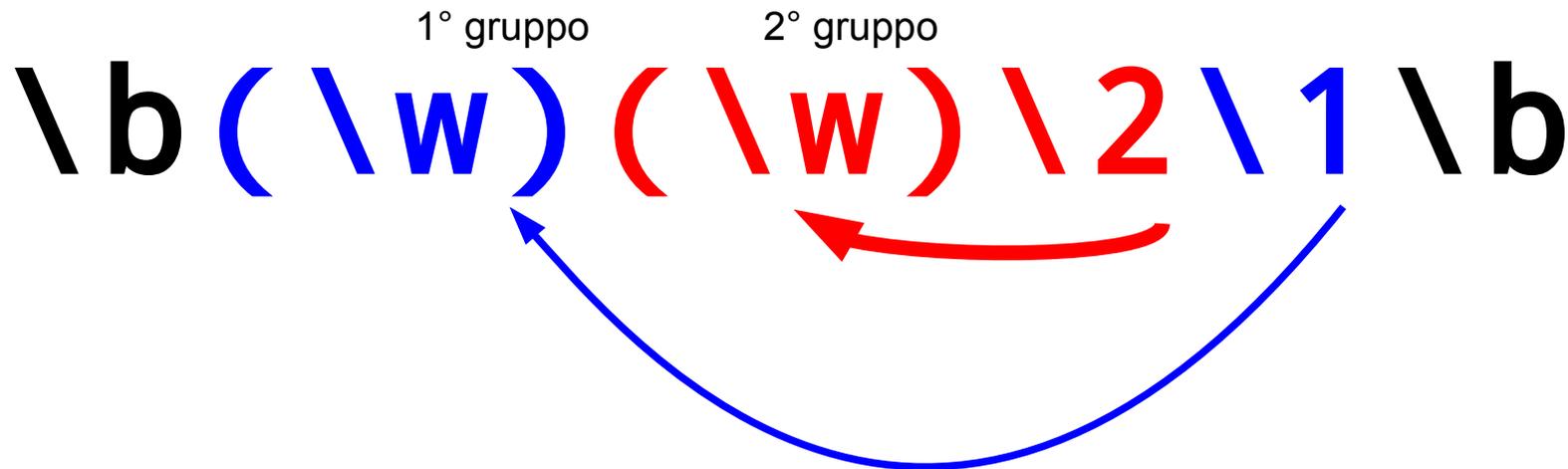
For more information contact neo@matrix.eu or jeff.jeffty@jeff.co.uk

ASCII syntax: `[\\w\\.]+@[\\w\\.]+`

Help

Back-reference (referenze all'indietro) \n

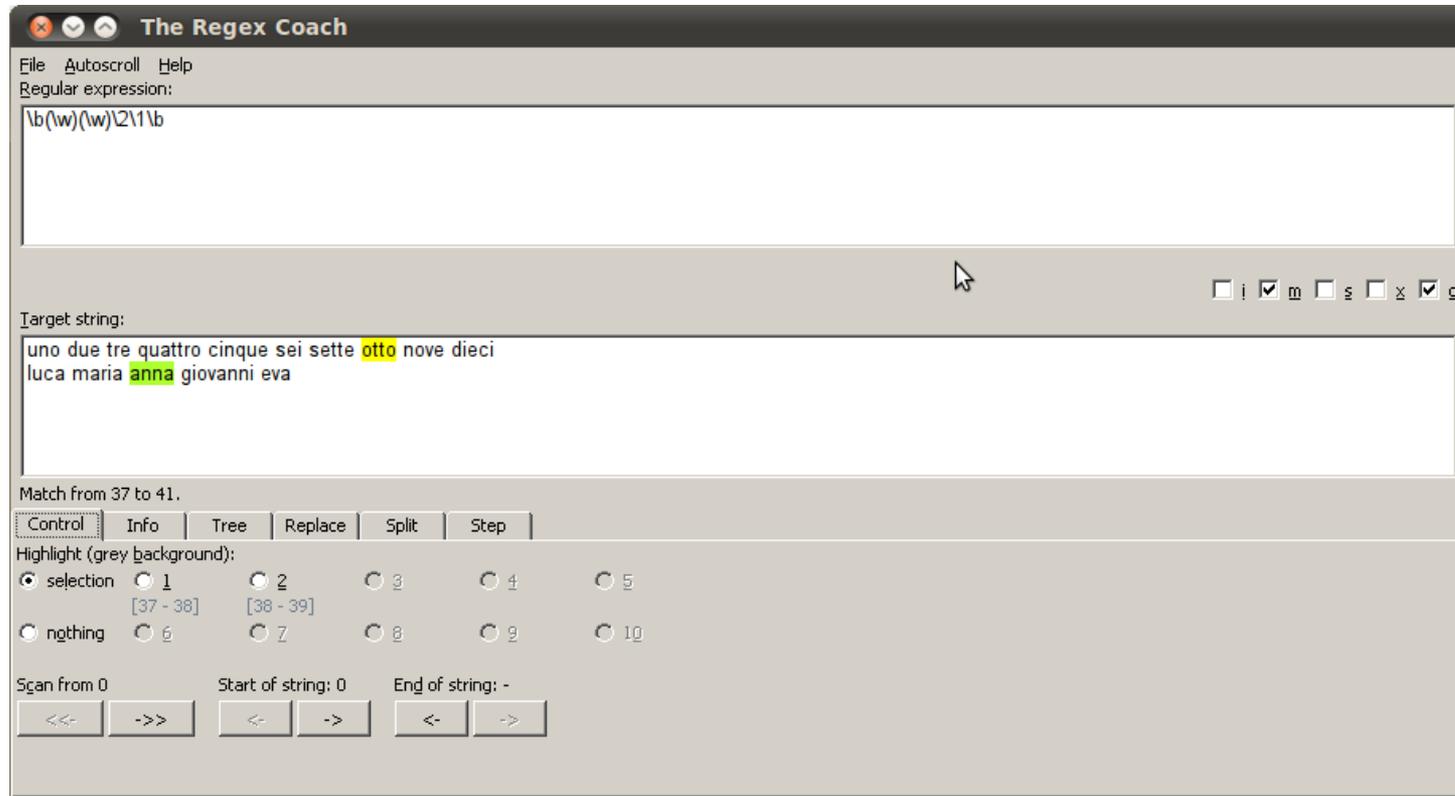
- Quando si utilizzano i gruppi () è possibile referenziarli successivamente nell'espressione regolare, specificando '\' (backslash) poi il loro indice (nell'ordine di definizione)



Questa espressione regolare trova i palindromi di 4 lettere

Back-reference (referenze all'indietro) \n

- Quando si utilizzano i gruppi () è possibile referenziarli successivamente nell'espressione regolare, specificando '\n' (backslash) poi il loro indice (nell'ordine di definizione)



Precedenze

- Quando l'espressione viene valutata, valgono le seguenti precedenze:
 - Alternative “|” (più forte)
 - Concatenazione di caratteri
 - Quantificatori (ripetizioni) (più debole)
- Raggruppando i termini in gruppi posso ridefinire le precedenze

grep

- grep può filtrare le righe in input in base a un'espressione regolare
 - Se viene trovato almeno una corrispondenza, la linea viene stampata
 - Noi utilizzeremo grep con l'opzione -E (extended regular expression) che supporta i quantificatori `?`, `+`, `{n}`, `{n,m}`, e `{n,}`
- Attenzione, le parentesi `{}` `()`, nonché `|`, `?`, e `+` devono essere preceduti da `\` (backslash), se non viene usata l'opzione -E

```
[X] bash
```

```
utente@host:~$ grep -E "\bmani\b" testo.txt
gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire.
ma domani possiamo esser poveri. E non si misero in cammino a mani vuote.
```

```
utente@host:~$ grep -E "\bl[aei]\b" testo.txt
gomiti sulle sue ginocchia e con la faccia appoggiata tra le mani, stavo a sentire.
Gli Egiziani trasmisero ai Romani le preparazioni che permettevano di trasformar le
```

Bash regex =~ / BASH_REMATCH

- Bash include un motore di espressioni regolari
 - Per verificare se una corrispondenza si trova all'interno di una stringa si utilizza `==~` e le doppie parentesi quadre `[[]]`

```
[X] bash
```

```
utente@host:~$ [[ "The dog is under the aBple" =~ [^d]*dog([^\t]*)(.*) ]]
utente@host:~$ echo $?
0
```

- Quando ho un match posso recuperare le stringhe corrispondenti ai vari gruppi accedendo all'array `BASH_REMATCH`

```
[X] bash
```

```
utente@host:~$ if [[ "The dog is under the aBple" =~ [^d]*dog([^\t]*)(.*) ]]; then echo ${BASH_REMATCH[2]}; fi
the aBple
```

- Posso anche simulare grep:

```
[X] bash
```

```
utente@host:~$ cat testo.txt | while read i; do if [[ $i =~ (^..) ]]; then echo $i; fi; done
```

Cheatsheet

- <http://www.addedbytes.com/cheat-sheets/regular-expressions-cheat-sheet/>