

Tutorial

Yudit: un editor Unicode che “parla” (anche) L^AT_EX

Gianluca Pignalberi

Sommario

Gli utenti L^AT_EX hanno a disposizione un’ampia gamma di editor tra cui scegliere il preferito: dai più spartani editor di testo privi di funzioni specifiche ai più complessi IDE da cui lanciare le compilazioni, verificare la coerenza e correttezza del codice e ottenere aiuti alla scrittura grazie alle scorciatoie. Tra le varie possibili scelte c’è Yudit, un editor Unicode leggero scritto agli albori dello standard Unicode su Linux.

Abstract

L^AT_EX users can pick their favorite editor among a wide range of such programs: from the most basic text editors without specific functions to the most complex IDEs to compile documents, verify the code coherence and correctness and obtain typing shortcuts from. Among the several choices we have Yudit, a lightweight Unicode editor written during the very first days of the Unicode standard on Linux.

1 Introduzione

Da diversi anni, ormai, gli utenti L^AT_EX possono beneficiare del supporto Unicode durante la composizione dei propri documenti. La parte “difficile”, eventualmente, è inserire determinati caratteri in maniera efficiente.

Tutti i moderni sistemi operativi offrono un ottimo supporto multilingua e mettono a disposizione degli utenti diverse configurazioni di tastiera. Purtroppo anche questo può essere limitante perché non tutti gli utenti hanno diverse tastiere da collegare al computer¹ e, in ogni caso, nessuna tastiera normale avrà mai tutti i possibili caratteri.²

Esistono diversi programmi di utilità (*utility*) che mostrano una mappa dei caratteri selezionando i quali si compone una stringa che può poi essere copiata e incollata nei propri documenti. Questa soluzione è molto versatile ma può essere scomoda se la quantità di stringhe da scrivere supera le poche unità.

1. Non è necessario per un computer avere la tastiera russa per permettere all’utente di digitare il russo perché il sistema riconfigura i tasti mappando, per esempio, la Q in Ъ. L’utilità di avere la perfetta corrispondenza mappa-tastiera è per l’utente che avrà più facilità a digitare il testo.

2. In realtà non credo neanche che esista un font con tutti i glifi a disposizione.

Nei casi in cui le precedenti soluzioni non siano praticabili è sempre possibile ricorrere a editor di testo³ specializzati, verosimilmente “residui” di quando l’immissione di testo multilingue era ancor più complicata. Emacs, per esempio, ha un ottimo supporto multilingue ma è tutto tranne che un editor leggero: è più un coltellino svizzero, delle dimensioni di un *machete*, capace di moltissime funzioni. Un editor leggero ma molto versatile dal punto di vista dell’immissione di testo multilingue è Yudit, tra le cui capacità c’è quella di far inserire direttamente dei caratteri Unicode anche se non mappati da alcuna tastiera e che offre un moderato supporto a L^AT_EX permettendo all’utente di scrivere dei glifi ricorrendo proprio ai comandi del nostro compositore preferito.

Questo tutorial costituisce un breve avvio all’uso di Yudit (nella versione per Linux) con l’auspicio che soprattutto i linguisti possano trarre beneficio dalle sue funzionalità.

2 Installazione

Sul sito ufficiale del programma troviamo gli eseguibili per Windows (32 e 64 bit) e per Linux, quest’ultimo in un comodo file RPM (comodo solo per gli utenti di distribuzioni derivate da Red Hat). Oltre a questi, abbiamo a disposizione i sorgenti (distribuiti sotto forma di *tarball* e RPM), le istruzioni per compilare l’eseguibile per Linux e Windows e per realizzare un pacchetto *.deb* per Debian e derivate. Sui repository, ufficiali e non, delle varie distribuzioni troviamo le versioni pacchettizzate.

L’installazione non comporta difficoltà e pure la compilazione non è molto difficoltosa perché Yudit è interamente basato sulla libreria standard del C e su un toolkit a finestre generico, non dipende da altro; questo è il motivo del suo aspetto “peculiare”. La versione stabile corrente è la 2.9.6 e la beta della 2.9.7 è già disponibile.

3 Uso e configurazione

Una volta avviato, Yudit appare come uno spartano editor di testo, con una barra di informazioni

3. La locuzione *editor di testo* ha un’accezione piuttosto limitata per gli informatici formati negli anni ’70-’80: è un programma che permette di scrivere un semplice testo, modificarlo, salvarlo in formato *.txt* e poco altro. Sono editor di testo *edit* di MS DOS e Blocco note di Windows il quale permette pure la ricerca all’interno del testo; Word non lo è: è un *elaboratore* di testo.

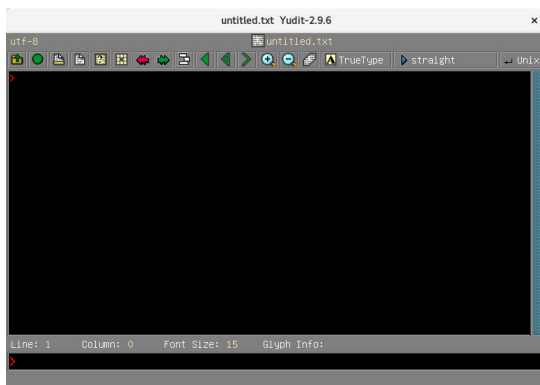


FIGURA 1: L'interfaccia di Yudit.

superiore, una barra dei pulsanti, lo spazio di inserimento del testo (che può avvenire da sinistra a destra — SD da ora in poi — o da destra a sinistra — DS da ora in poi —), una barra di informazioni inferiore e una riga di comando. La figura 1 mostra l'editor appena avviato.

Nella barra superiore delle informazioni troviamo l'indicazione sulla codifica del file (utf-8) e il nome del file aperto. Per default viene tentata l'apertura di `untitled.txt`. La sottostante barra dei pulsanti mette a disposizione dell'utente le seguenti funzionalità: “apri file”, “salva”, “stampa”, “stampa anteprima”, “cerca”, “vai a”, “annulla”, “riesegui”, “allinea testo”, “ignora direzione”, “ignora inclusione”, “forza inclusione”, “ingrandisci” e “rimpicciolisci” (il font del testo), “modalità di colorazione contestuale del testo” o “evidenziazione” (*highlighting*), “font” (il pulsante mostra quello in uso correntemente), “modalità di inserimento” (il pulsante mostra quella correntemente in uso) e “fine riga” (Dos, Mac, Unix e, dalle versioni più recenti, Ps/Ls, cioè *Unicode Paragraph/Line Separator*).

Sullo spazio di inserimento del testo non c'è nulla da dire se non che il testo può essere inserito nelle varie modalità previste dallo standard Unicode. La sottostante barra delle informazioni contiene il numero di riga e colonna in cui si trova il cursore, la dimensione in punti del font usato nell'editor e le informazioni sul glifo, segnatamente il suo codice ASCII/Unicode. A seconda della modalità di inserimento e della direzione troviamo informazioni aggiuntive sul tasto o sulla sequenza di tasti usati per digitare il glifo e la direzione.

Da bravo editor spartano, Yudit non ci fornisce alcun menù tramite cui configurare il suo aspetto o comportamento. Possiamo però agire sul file delle proprietà per configurare molto dettagliatamente il programma. Nella cartella (che per retaggio culturale mi ostinerò a chiamare directory) `~/.yudit` troviamo le sottodirectory `data`, `font`, `syntax` e il file `yudit.properties`. Quest'ultimo è un semplice file di testo che possiamo modificare in base alle nostre esigenze. Vediamo come.

Il file inizia con una serie di righe di spiegazione iniziate dal carattere `#`. Questo indica che tali righe sono dei commenti utili all'utente ma inutilizzate da Yudit. La prima proprietà, che però non possiamo personalizzare, è il numero di versione di Yudit. Pur provandoci, Yudit sovrascriverà comunque l'informazione con quella propria.

Le successive proprietà configurabili sono: colore⁴ dello sfondo della finestra, colore di sfondo della riga di comando e del suo cursore SD e DS, font usato per i comandi e sua dimensione, colore dei caratteri per l'immissione SD e viceversa. Seguono due informazioni su dove si trova il file locale di configurazione (i commenti iniziali ci ricordano che, in mancanza di esso, Yudit si rifarà alla configurazione generale sita in `/usr/share/yudit/config/yudit.properties`) e dei file `.my` (cioè i file delle mappe della tastiera contenuti nella directory `data`).

Le due seguenti proprietà indicano quale codifica avrà la *clipboard* (cioè lo spazio di memoria in cui viene salvato ciò che viene tagliato o copiato) e il file di default (`untitled` in formato UTF-8).

Immediatamente di seguito troviamo la configurazione del formato del file di testo. Il formato predefinito è, di nuovo, UTF-8 ma abbiamo a disposizione un nutrito numero di formati alternativi, peraltro selezionabili nella finestra di salvataggio (come mostrato nella figura 2).

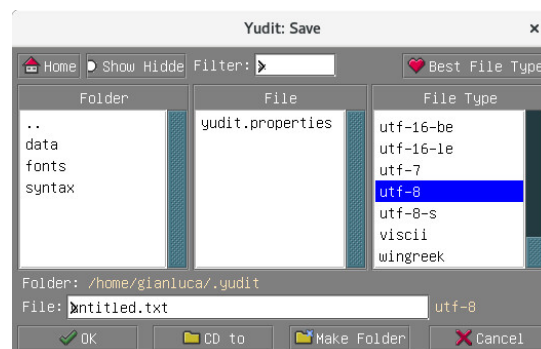


FIGURA 2: Nella colonna di destra della finestra di salvataggio possiamo selezionare il formato del file di testo. Il numero quasi rivalessa con quello messo a disposizione da `iconv`.

Quindi sono indicate le configurazioni relative al font usato da Yudit per tutte le indicazioni del programma all'utente. Sconsiglio di modificare tali impostazioni perché a ogni modifica del font (che è indicato come “default” ed è di sicuro uno di quelli indicati a pagina 4) ho sempre ottenuto delle scritte illeggibili perché composte da quadrati col codice Unicode del glifo anziché dal glifo stesso.

Seguono l'indicazione della geometria della finestra, della lingua, del comando da usare per l'anteprima (`gv`), il colore di sfondo dell'area di

4. I (nomi dei) colori sono quelli definiti per X11. Si possono trovare anche nel manuale del pacchetto `xcolor`.

scrittura e del cursore (nei casi SD e DS), i tipi di file (in aggiunta a quelli di `uniconv`), il font corrente nell'editor, quindi segue una lunga lista di associazioni di etichette e font possibili da usare nell'area di editing e di colorazioni dell'evidenziazione.

Per avere le informazioni relative alla configurazione possiamo digitare, nella riga di comando di Yudit (si passa dallo spazio di editing alla riga di comando e viceversa premendo Esc, quasi in stile vi, o col mouse),

`howto configure`

mentre per avere informazioni generali digitiamo

`help`

nello stesso spazio adibito ai comandi all'editor.

Iniziamo la digitazione in modalità `straight`, cioè quella identicamente uguale alla tastiera di sistema (o quasi: io ho difficoltà a inserire le accentate in questa modalità da quando ho Linux Fedora 26). Per cambiare modalità di inserimento (cioè cambiare mappa della tastiera e quella dei caratteri attivi⁵), che da ora in poi chiameremo pur impropriamente *mappa tastiera*, abbiamo due modi: normale e rapido. Per cambiare in modo normale premiamo il pulsante inserimento e si aprirà una finestra (figura 3). Nella colonna di sinistra troviamo tutte le mappe disponibili, nella colonna centrale ne troviamo 12 associate ai tasti F1...F12 (in mezzo a queste due colonne c'è un pulsante che useremo per associare una delle mappe a uno dei tasti funzione dopo aver selezionato entrambi) e nella colonna di destra che mostra quale glifo è associato a quale tasto o sequenza di tasti.

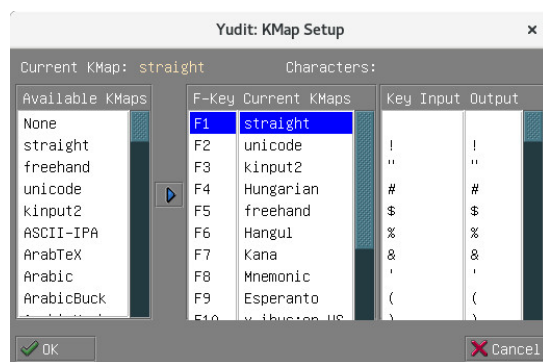


FIGURA 3: Associazione mappa tastiera-tasti funzione al primo avvio di Yudit.

Proviamo a cambiare associazioni per un caso reale: il mio. Voglio associate al tasto F3 il russo in configurazione `яверты` (più comoda perché mappa i caratteri traslitterandoli quando possibile: `a` → `a`, `b` → `б` e così via) e non `йцукен` e al tasto F4 `TeX`. Basta selezionare la mappa corretta e il tasto a

5. Definiamo *caratteri attivi* quei caratteri che in combinazione con determinati altri caratteri successivi inseriscono nel testo un glifo specifico diverso da quelli inseriti.

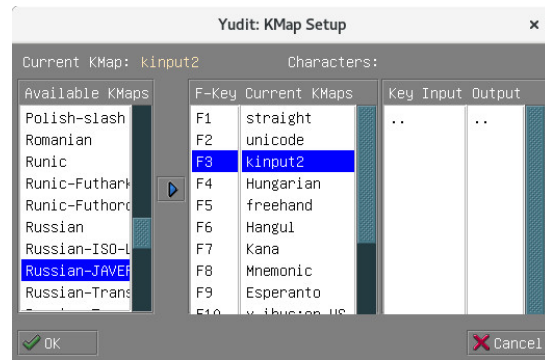


FIGURA 4: Quale mappa tastiera vogliamo associare a quale tasto funzione?

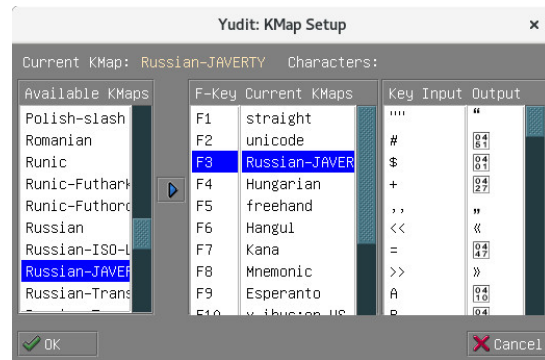


FIGURA 5: La nuova associazione mappa tastiera-tasto funzione è ora attiva.

cui associarla (figura 4), quindi premere il pulsante con la freccia per aggiornare le associazioni e il relativo file di configurazione (figura 5).

Una volta che abbiamo personalizzato o meno le associazioni mappa tastiera-tasto funzione in base ai nostri bisogni possiamo cambiare rapidamente la modalità di inserimento premendo il tasto funzione associato a tale mappa.

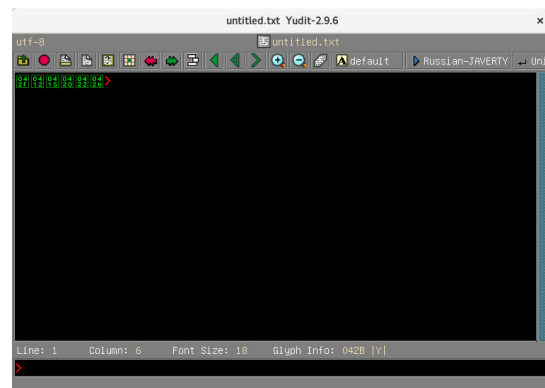


FIGURA 6: Il testo russo risulta illeggibile perché il font in uso non dispone dei glifi adatti.

Nelle figure 5 e 6 notiamo che i glifi cirillici sono invisibili. Al loro posto compaiono i relativi codici Unicode. Questo significa che i font

di default di Yudit (molti dei quali non possono essere distribuiti col programma e vanno quindi cercati e copiati nella directory `~/.yudit/fonts`) non sono presenti nel sistema usato o non contengono quei glifi. Dunque rimettiamo mano al file di configurazione per fare in modo di usare un font Unicode diverso da quelli impostati di default. Ci affideremo a GNU Free Serif, Free SansSerif e Free Mono, gli equivalenti liberi di Times, Helvetica e Courier. Per prima cosa troviamo i relativi file nel nostro sistema (in Linux Fedora sono sotto `/usr/share/fonts/gnu-free`) e copiamoli in `~/.yudit/fonts`.⁶ Cerchiamo le stringhe `yudit.font.SansSerif` e `yudit.font.Serif` e dopo il segno `=` scriviamo, rispettivamente, `FreeSans.ttf` e `FreeSerif.ttf` per istruire Yudit a usare gli onnipresenti GNU Free SansSerif e Free Serif. Per usare Free Mono creiamo una nuova categoria di font in `yudit.editor.fonts: Mono`. Dopodiché andiamo, sempre nel file di configurazione, dopo la riga in cui abbiamo configurato Free Serif e scriviamo:

```
yudit.font.Mono=FreeMono.ttf
```

Ora basta salvare le modifiche e riavviare Yudit per aver modo di scrivere anche in Courier (figura 7).⁷

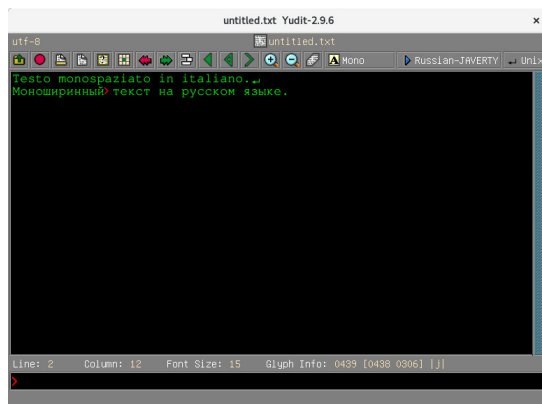


FIGURA 7: Yudit con un font monospaziato.

L'obiettivo, però, non è tanto usare un font monospaziato nell'editor quanto usarlo nella finestra di configurazione delle mappe di tastiera per fare in modo che i glifi non ancora visibili diventino. Purtroppo la cosa non è possibile a livello di configurazione. Come confermatomi

6. Naturalmente potremmo sempre modificare la proprietà `yudit.fontpath` del file di configurazione per far sì che Yudit possa riconoscere uno o più font installati nel sistema senza doverli copiare localmente, come invece abbiamo fatto qui. La copia locale dei font è utile solo nel caso in cui tali font non siano installati nel sistema, non possano esserlo o, pur essendolo, debbano essere rimossi.

7. Ricordiamo che Yudit è un editor di testo e non un word processor, pertanto il font usato sarà usato per tutto il testo; ogni cambiamento di font si ripercuoterà su tutto il testo e nessuna informazione sui font verrà salvata.

dall'autore, i font usati in questo caso sono predefiniti nel codice sorgente, precisamente nel file `yudit-2.9.6/swindow/SFont.cpp`:

```
66 SStringVector m(
67     /* yudit.hex has ohungarian
        .hex and part of unifont
        .hex */
68     // Indic range is not
        necessary free. But they
        are the best
69     "yudit.hex, arabforms.hex,
        syriacforms.hex, unifont.
        hex, "
70     "markus9x18.bdf,
        markus18x18ja.bdf, "
71     "--*-medium-r-normal
        --16-*-*--c*-iso8859
        -1, "
72     "--*-medium-r-normal
        --16-*-*--c*-iso8859-1, "
73     "TH00LIUC.TTF:mlym, "
74     "MuktiNarrow.ttf:beng, "
75     "ani.ttf:beng, "
76     "pothana2000.ttf:telu, "
77     "TCRCYoutsoUnicode.ttf:tibt
        , "
78     "raghu.ttf:deva, "
79     "mangal.ttf:deva, tunga.ttf:
        knda, code2000.ttf:taml, "
80     "raavi.ttf:guru, shruti.ttf:
        gujr, "
81     "arialuni.ttf, cyberbit.ttf,
        "
82     "code2000.ttf, code2001.ttf:
        unicode:RL, arial.ttf, "
83     //"rovasSMP.ttf," // Hosszu
        Gabor
84     //"oldhunSMP.ttf," //
        Michael Everson
85     "yudit.ttf"
86 );
```

A meno di non modificare i sorgenti e ricompilare tutto,⁸ la soluzione è installare nel proprio sistema uno dei font elencati. Dopo aver copiato `unifont.hex` nella directory locale dei font il risultato è quello mostrato nella figura 8.

Vediamo ora come l'uso di caratteri attivi ci permetta di inserire una serie di simboli non presenti nella tastiera. Selezioniamo la mappa tastiera Unicode e scriviamo la seguente stringa:

```
u2014_u263a_u2469_u2127_u2055
```

Man mano che scriviamo un codice Unicode ben formato, cioè formato da una 'u' e un numero di 4 cifre esadecimali (la u, mostrata sottolineata, indica l'immissione di un carattere attivo; i successivi

8. Ho suggerito all'autore di includere nel sorgente ufficiale anche i font GNU Free, Linux Libertine/Biolinum e DejaVu.

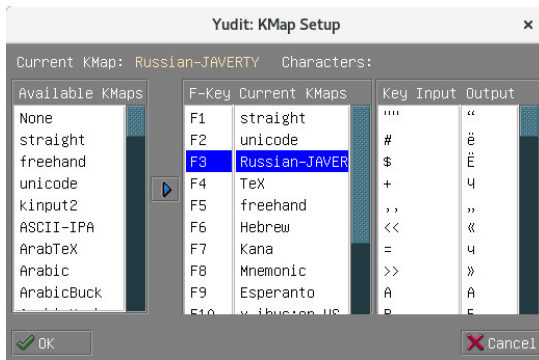


FIGURA 8: Il testo russo risulta leggibile anche nelle finestre dopo aver installato un font che Yudit riconosce perché incluso nel suo codice sorgente.

numeri o lettere a–f sottolineati indicano l'immissione di una sequenza attiva; si veda la figura 9), questo viene sostituito dal corrispondente carattere. La stringa risultante è mostrata nella figura 10.

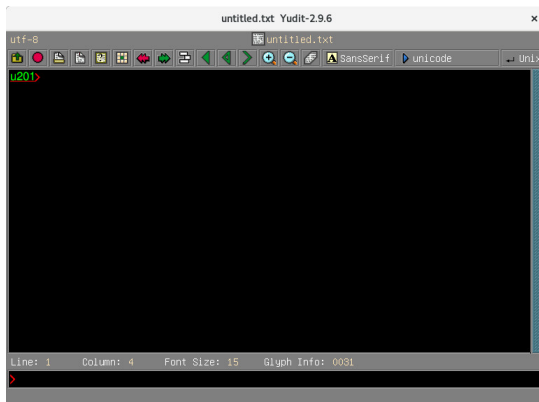


FIGURA 9: Una sequenza attiva inserita con la modalità di inserimento Unicode.

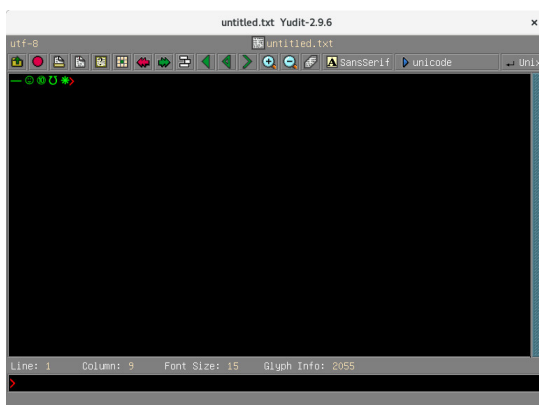


FIGURA 10: Yudit rimpiazza una sequenza attiva col carattere corrispondente se questo è presente nel font usato.

La scrittura DS è un compito molto facile per Yudit. Se scegliamo una lingua scritta in tale verso, ad esempio l'ebraico, Yudit imbandiera a destra tutto il testo scritto in questa lingua (a meno di non

aver agito sul pulsante “allinea testo” per forzare l'allineamento sempre a sinistra o sempre a destra) e tale testo è riconoscibile per il diverso colore (a meno di non aver modificato tale proprietà) rispetto al testo SD.

Può capitare che all'interno di un testo SD occorra inserire delle parole o frasi scritte in una lingua DS. In questi casi bisogna fare attenzione a che la punteggiatura ricada in maniera corretta nel testo. A questo servono i pulsanti “ignora direzione” e “ignora inclusione”. La sezione `bidi` degli *howto* di Yudit fornisce dettagli tecnici e alcuni esempi utili a chiarire la questione. Basta digitare

`howto bidi`

nella riga di comando di Yudit per accedervi. Qui viene spiegato anche l'uso del pulsante “forza inclusione” che serve a portare il testo evidenziato alla direzione principale del testo.

Un'ultima interessante caratteristica di Yudit è quella offerta dalla mappa `freehand`. Questa selezione fa assumere a Yudit l'aspetto mostrato nella figura 11. Nell'area sottostante allo spazio di editing troviamo al centro l'area in cui disegnare un carattere col mouse, a sinistra una serie di “alfabeti” orientali in cui cercare una corrispondenza col carattere disegnato e a destra la lista dei caratteri possibili. La figura 12 mostra la lista dei risultati relativi alla lista “roman” trovati per lo scarabocchio a forma di F. Ovviamente fornisce altri risultati per gli altri “alfabeti” elencati. Basta selezionare il carattere cercato per vederlo comparire nello spazio di editing.

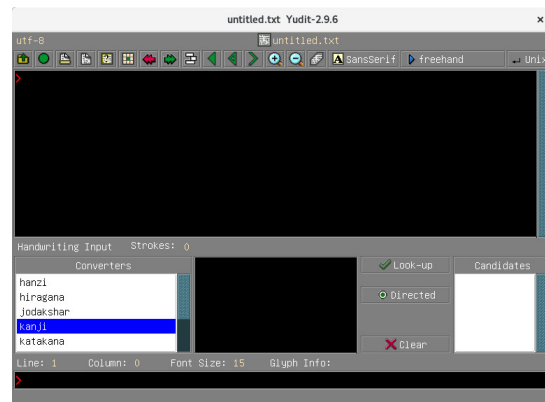


FIGURA 11: Yudit con la mappa `freehand` ci permette di disegnare un carattere per trovarlo in una lista di possibili candidati (cosa utile con lingue come il giapponese).

4 Yudit e L^AT_EX

Ora che abbiamo tutte le basi d'uso di Yudit, sarà facilissimo scrivere qualunque documento L^AT_EX, tenendo presente che Yudit è un editor *general purpose* e quindi non offre scorciatoie per inserire comandi quali `\emph` o simili, però possiamo

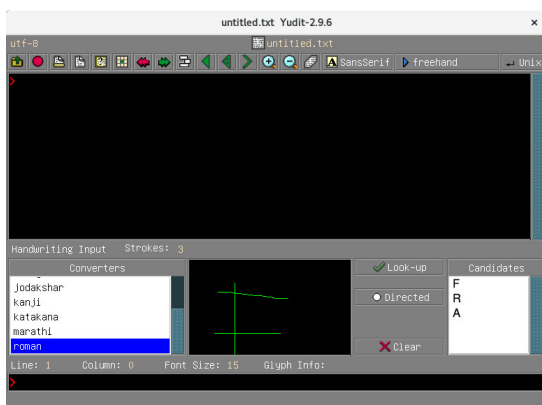


FIGURA 12: Uno scarabocchio a forma di F viene “riconosciuto” come F, R o A secondo l’alfabeto roman.

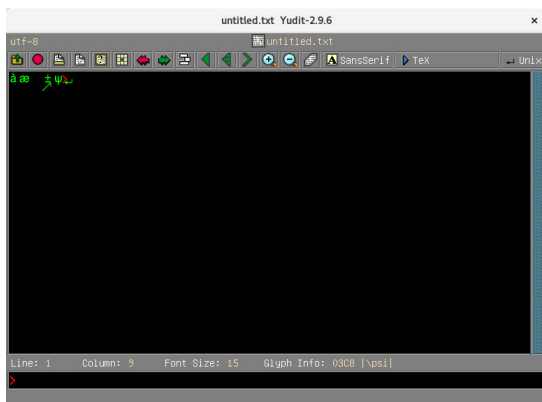


FIGURA 13: Sequenza di caratteri ottenuti tramite la mappatura TEX.

scrivere molti caratteri assenti sulla tastiera con la sintassi uguale o molto simile a quella di LATEX. Per esempio, la sequenza mostrata nella figura 13, è stata ottenuta selezionando la mappa tastiera TEX e digitando:

```
\‘a_ \ae_ \swarrow_ \pm_ \psi
```

Per un TEXnico sarà immediato scrivere qualunque documento sfruttando la propria conoscenza dei comandi di LATEX.

5 Conclusioni

I linguisti potrebbero incontrare alcune difficoltà nella stesura dei propri documenti nonostante la sempre maggior potenza e versatilità dei più comuni *word processor* e il sempre miglior supporto multilingua dei moderni sistemi operativi. In certi casi può essere utile avere a disposizione un editor specializzato nell’inserimento di testi multilingua.

Yudit è senz’altro uno dei prodotti di riferimento del campo: permette di rimappare con molta facilità la propria tastiera per supportarne in diverse lingue; permette di inserire molti caratteri mediante comandi di noti linguaggi di markup (SGML) o composizione tipografica (TROFF, TEX); permette la scrittura da destra a sinistra; consente di salvare il testo col fine riga di diversi sistemi operativi; contiene anche un modulo per la scrittura manuale per trovare agevolmente “caratteri” che si sanno disegnare ma non digitare.

Pur non essendo il mio editor preferito né quello di immediato ripiego (nell’ordine, vi e Emacs), di sicuro Yudit mi ha tolto spesso da grandi impacci in ambito lavorativo.

Riferimenti bibliografici

«The Unicode Consortium». URL <http://www.unicode.org>.

«Yudit». URL <http://www.yudit.org>.

KORPELA, J. K. (2006). *Unicode explained*. O’Reilly, Sebastopol (CA).

▷ Gianluca Pignalberi
g dot pignalberi at gmail dot
com