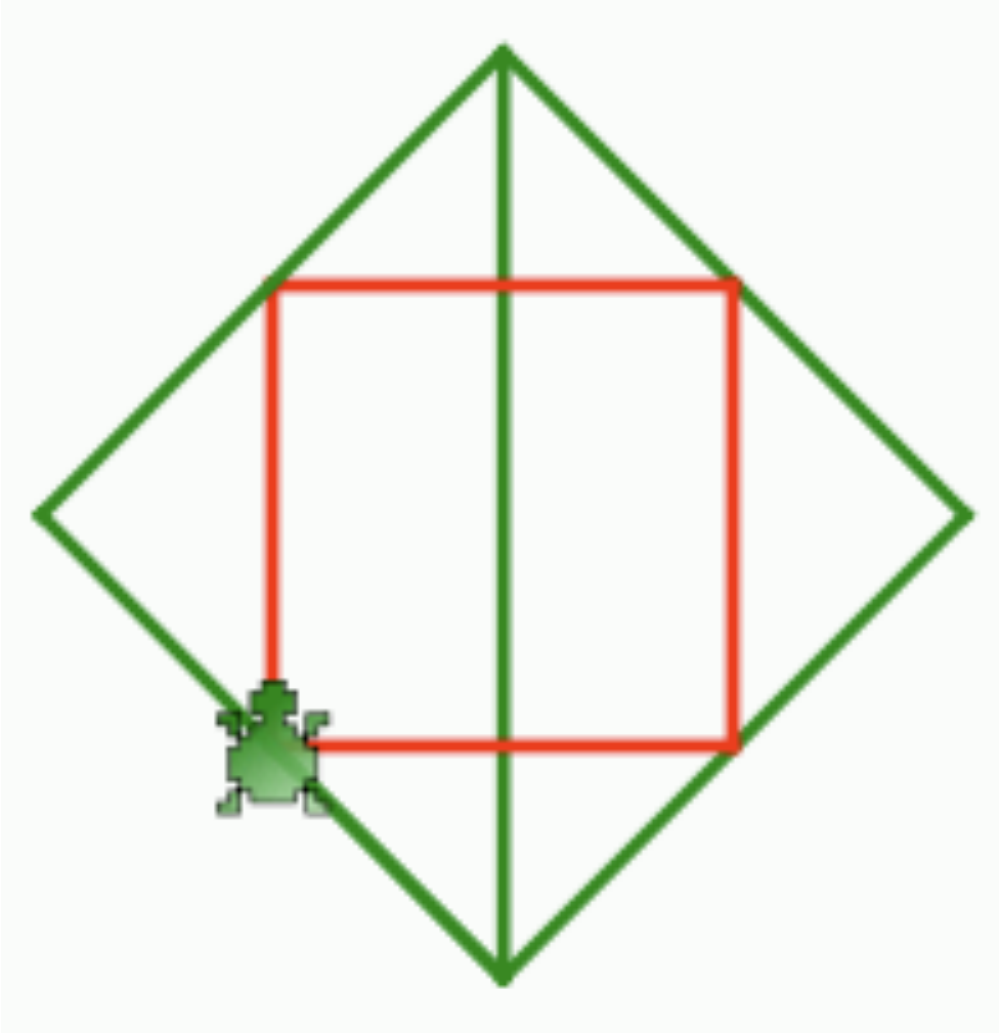
C O R S O D I L O G O

MANUALE DI RIFERIMENTO
ED ESERCIZIARIO, LEZIONE N.1



LEZIONE N.1

versione 1.0.1 18/02/2009

Lezione n.1

Introduzione

Definire l'informatica in poche parole non è semplice, ma ci proveremo lo stesso :) La definizione ufficiale è che l'informatica è la scienza che studia come manipolare le informazioni, ma questo non ci dice molto. Accendere il computer a casa, navigare su internet, scrivere una lettera e poi guardare un video su youtube: si sta facendo dell'informatica?

Una definizione semplice (ma non rigorosa) è che fare informatica vuol dire "inventare" dei procedimenti per insegnare a *qualcuno* ad eseguire dei compiti: questo *qualcuno* è tipicamente un computer, ma non è detto che sia così! Facciamo un'analogia: l'astronomia è la scienza che studia gli astri. Il telescopio è utile per osservare le stelle, ma è possibile fare astronomia anche senza telescopio (e gli uomini lo hanno fatto per millenni!). Dunque, è possibile fare informatica anche senza il computer.

Ogni qualvolta che si inventa un procedimento "meccanico" per svolgere un compito, si sta facendo un passetto verso l'informatica. Ad esempio, fare una divisione con carta e penna è un procedimento "meccanico": è garantito che, seguendo i passi previsti dal procedimento, *e se non si fanno errori di calcolo*, si otterrà il corretto risultato dell'operazione. Questo procedimento è chiamato algoritmo (dal nome del matematico arabo Al Khwarizmi). Insegnare un *algoritmo* ad un computer, ad un robot, od in generale ad un automa non intelligente, vuol dire fare informatica.

Insegnare un procedimento al computer ha due vantaggi incredibili rispetto all'insegnare lo stesso procedimento all'uomo: quando il computer esegue i passi del procedimento che gli abbiamo insegnato (le *istruzioni* dell'*algoritmo, o programma*), non commette mai errori di calcolo, e lo fa in maniera velocissima! Lo svantaggio del computer è che è completamente stupido, ossia non sa fare niente se qualcuno non glielo ha prima insegnato, mentre l'uomo è molto intelligente e può imparare da solo a fare cose nuove. Uomo e computer possono dunque andare molto d'accordo, poichè uno è intelligente ma lento, e l'altro è stupido ma veloce: insieme, sono riusciti ad andare sulla luna!

Usare un programma scritto da altri (per navigare su internet, o per giocare) vuol dire essere un utente (colui che usa): l'informatico è colui che ha scritto il programma, non quello che lo usa. Chi guida l'automobile sa come funziona *esattamente* al suo interno? Saprebbe costruire un'automobile tutta sua? Per analogia, l'automobilista è l'utente, mentre l'ingegnere che l'ha progettata è l'informatico.

Esattamente come per le automobili, un programma complesso è progettato e scritto da diverse persone, ognuna con competenze diverse, e raramente da una sola persona.

Riassumiamo brevemente:

• L'informatico insegna al computer a fare delle cose che prima non sapeva fare, scrivendo delle istruzioni che il computer può capire. L'insieme di queste istruzioni è un programma. Il procedimento per realizzare un qualcosa si chiama *algoritmo*. Un programma è un insieme di istruzioni che realizzano uno o più algoritmi.

• L'utente è colui che usa i programmi.

• L'uomo è intelliugente ma lento, il computer è stupido ma veloce.

• Il computer appena comprato sa già fare delle cose, semplicemente poichè è venduto con dei programmi preinstallati. Il programma principale è il ***Sistema Operativo***, che permette al computer di funzionare e dialogare con l'esterno. Esempi di Sistemi Operativi: Windows, Linux Ubuntu, Mac OSX . Su un computer possono essere installati contemporaneamente più di un sistema operativo, esattamente come per tutti gli altri tipi di programmi.

Kturtle

KTurtle è un'implementazione open source del logo, un linguaggio di programmazione didattico sviluppato dal professor Seymour Papert al MIT (Massachussets Institute of Technology). Il software è gratuito e fa parte del gestore grafico KDE, in particolare del pacchetto Education, che contiene altri utili programmi educativi per i bambini. Kturtle è un'ottimo strumento per insegnare l'informatica e la geometria.

KTurtle può essere installato su qualsiasi sistema linux, anche senza il KDE completo. La modalità di installazione dipende da una serie di variabili: la particolare distribuzione linux, il gestore grafico scelto, i pacchetti selezionati al momento dell'installazione iniziale. E' possibile che kturtle sia già installato: se non lo fosse, si deve utilizzare la procedura specifica per la distribuzione scelta. Con Ubuntu, attualmente la distribuzione linux più diffusa in ambito desktop, il modo più semplice è utilizzare il programma Synaptic.

La versione di kturtle utilizzata in questo manuale è la 0.8 beta, inclusa nel kde 4.1. Attenzione: versioni precedenti o successive possono avere dei menu diversi, o anche piccole differenze sintattiche nel linguaggio (come nuovi comandi o utilizzo di caratteri diversi)

KDE, e di conseguenza kturtle, può essere installato anche su sistemi operativi Windows o Mac, ma le indicazioni date in questo manuale sono valide solo per linux. Per maggiori informazioni riguardo kde e windows, consultate il sito http://windows.kde.org/

Lanciare kturtle

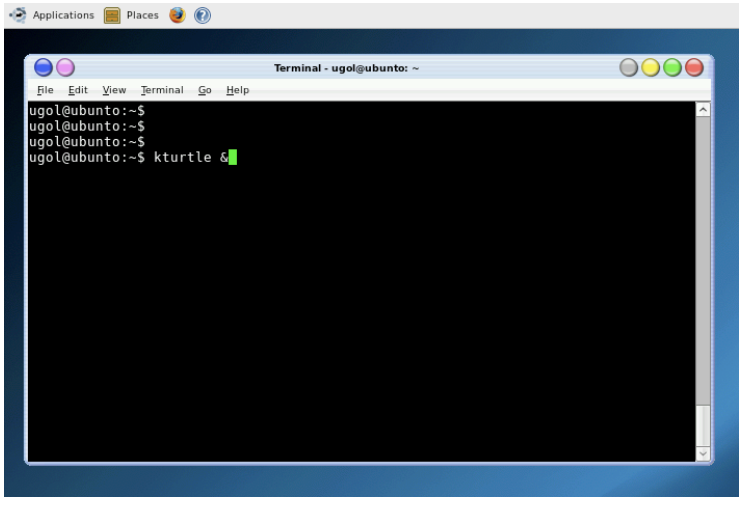
In un'installazione tipica di Linux, Kturtle può essere lanciato dal menu grafico.

Applications -> Education -> Kturtle

o, in un'installazione in lingua Italiana:

Applicazioni -> Educazione -> KTurtle

Il menu Applicazioni si trova di solito in alto a sinistra o in basso a sinistra, ma potrebbe essere in uno qualsiasi dei 4 lati dello schermo.



Alternativamente, è possibile aprire un terminale:

Applications -> Accessories -> Terminal

Applicazioni -> Accessori -> Terminale

ed inserire il comando:

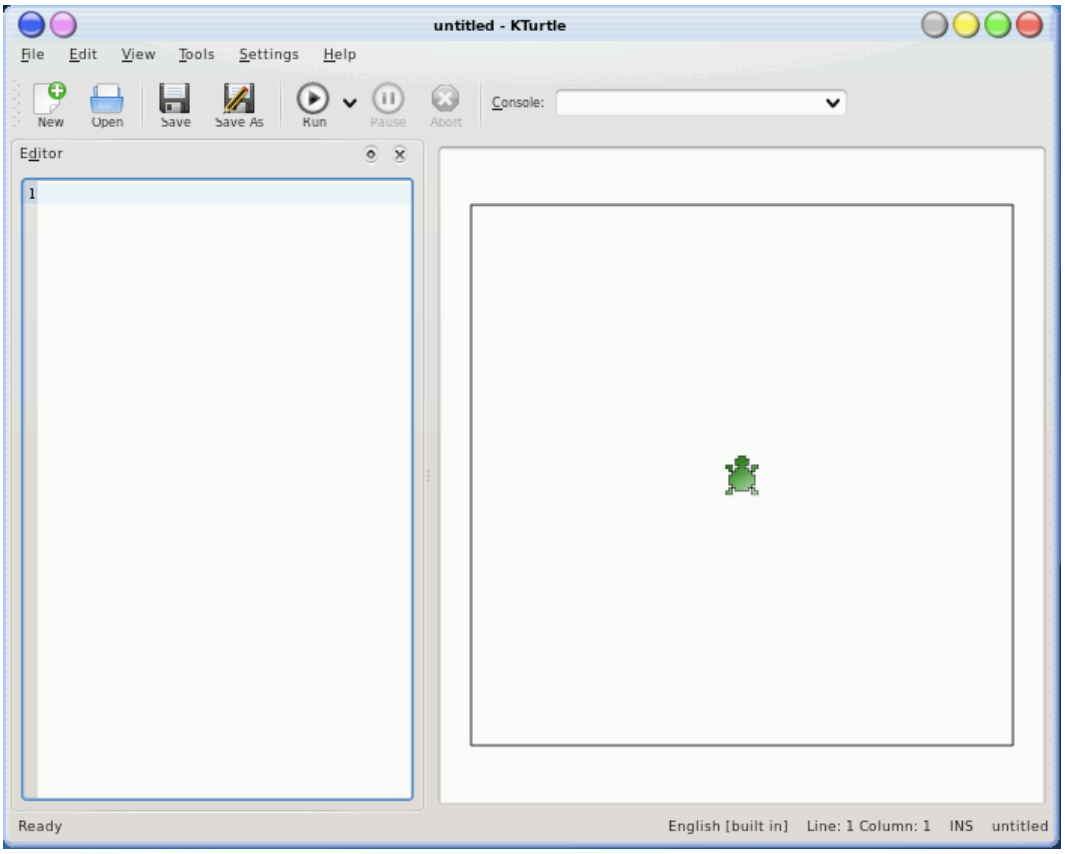
kturtle& ↲

come mostrato in figura.

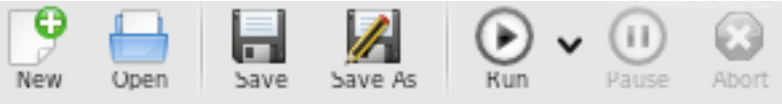
(↲ indica la pressione del tasto "Invio" o "Return")

L'ambiente di lavoro

L'ambiente di kturtle consiste in una finestra principale divisa in tre sezioni.



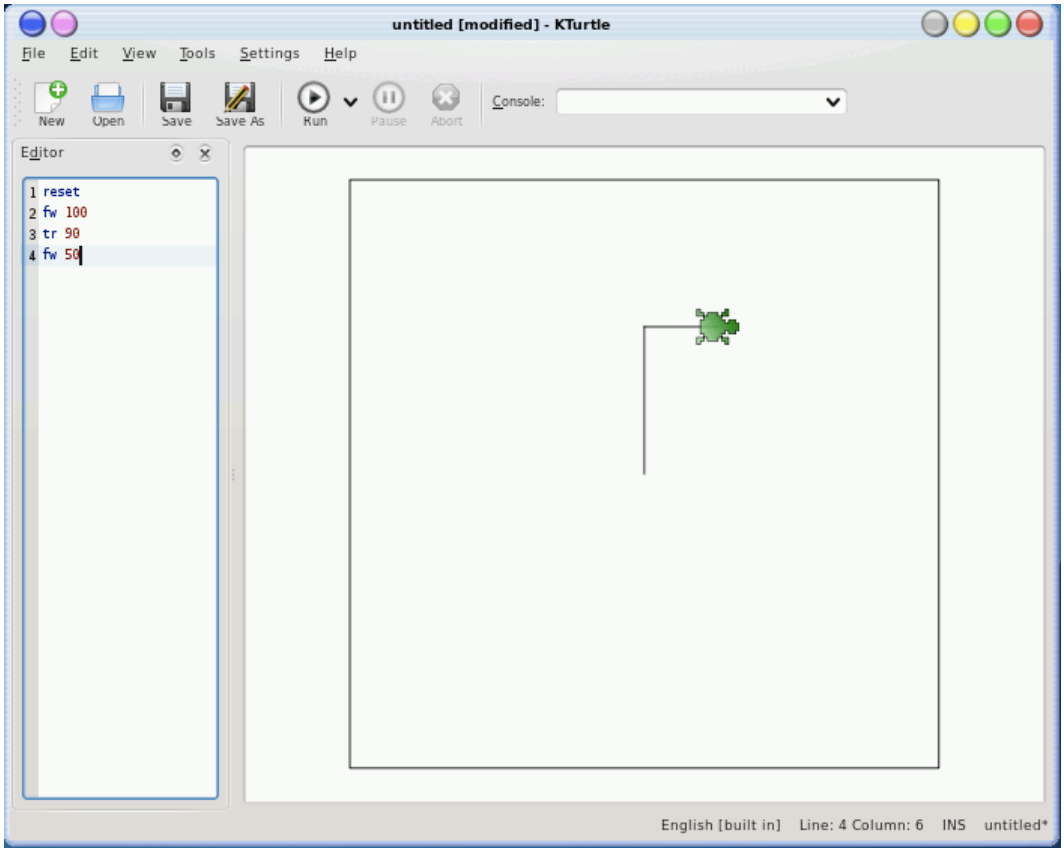
In alto c'è il menu principale: le diverse icone permettono rispettivamente di creare un nuovo progetto, aprirne uno esistente, salvare, salvare con nome, eseguire il programma, metterlo in pausa o terminarlo.



A sinistra c'è l'editor, dove si possono inserire le istruzioni per la tartaruga, mentre a destra c'è lo spazio di lavoro vero e proprio (canvas, o tela), dove la tartaruga eseguirà i comandi impartiti.

Il primo programma

Il primo programma che scriveremo è molto semplice e consiste nel disegnare delle linee rette.



Le istruzioni si inseriscono nella sezione "Editor". E' consigliabile inserire una sola istruzione per linea per migliorare la leggibilità dei programmi. Kturtle può utilizzare comandi in Inglese (il default) ma può anche essere localizzato in altre lingue, tra cui ovviamente l'Italiano. Per questo mini-corso si è comunque scelto di utilizzare l'Inglese per abituare i bambini all'uso della lingua straniera.

Analizziamo riga per riga il programma scritto:

reset

Questo comando pulisce lo schermo e posiziona la tartaruga al centro. E' consigliabile iniziare ogni programma con questo comando, poichè altrimenti ad ogni esecuzione la tartaruga non "pulirebbe" lo schermo, lasciando i segni dell'esecuzione dei precedenti programmi.

forward 100

Questo comando dice alla tartaruga di camminare in avanti per 100 passi: ogni passo della tartaruga corrisponde ad un punto (pixel) sullo schermo. Può essere abbreviato in **fw 100:** questa forma è più veloce da digitare e permette di commettere meno errori: è consigliabile iniziare con la forma completa e solo successivamente passare a quella abbreviata. La tartaruga, camminando, lascia un segno sulla tela (canvas).

turnright 90

Questo comando dice alla tartaruga di ruotare (turn) a destra (right) di 90°. Può essere abbreviato in **tr 90.** La tartaruga, ruotando sul posto, non lascia ovviamente nessun segno sulla tela.

forward 50

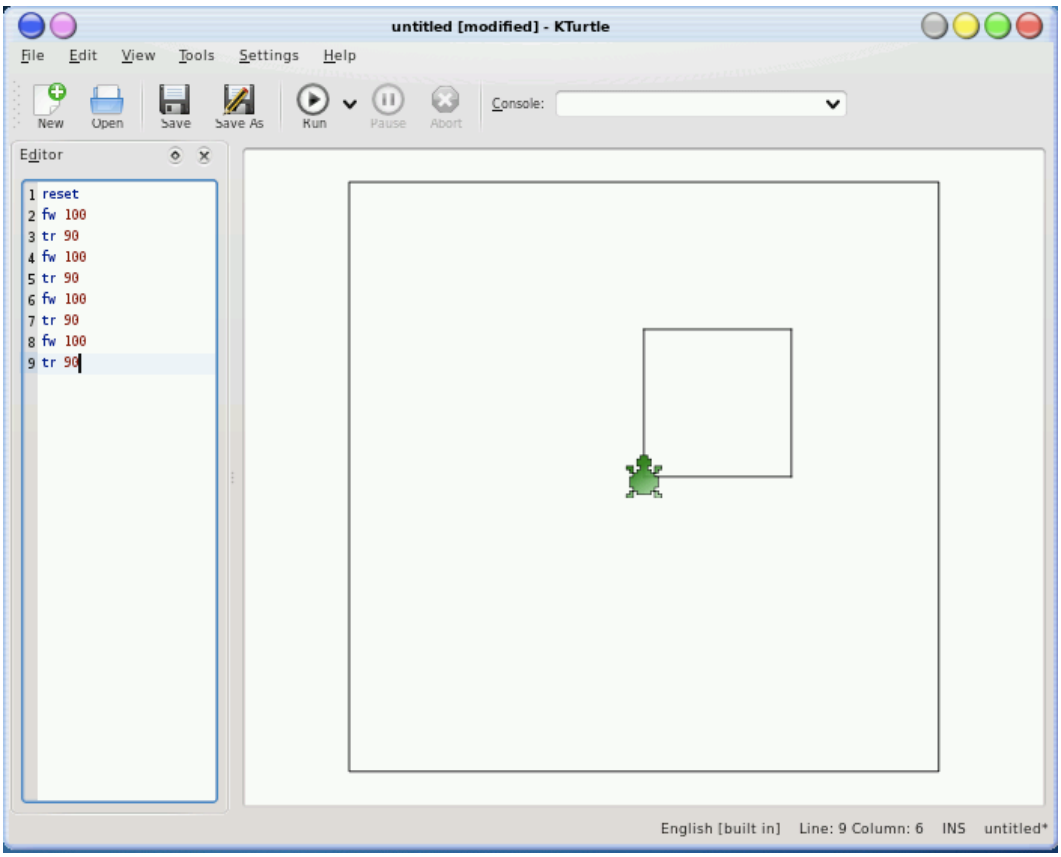
Dopo aver girato a destra, disegnamo una linea più corta in avanti, questa volta la metà della precedente. Notare che la tartaruga è ora rivolta verso destra: i comandi sono relativi alla posizione attuale della tartaruga e non assoluti! Quindi il comando forward va avanti rispetto alla testa della tartaruga, e non rispetto allo schermo.

Per esercizio, sperimentare liberamente per familiarizzare con i comandi. Ricordarsi che la tartaruga non esegue immediatamente i comandi, ma che bisogna clickare con il mouse sull'icona "Run" (o "Esegui").

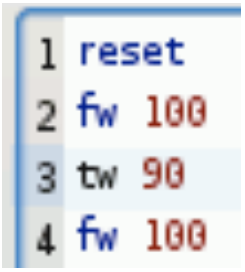
Facciamo un quadrato!

Ora si può passare a qualcosa di più complesso, come un quadrato.

Il modo più immediato di disegnare un quadrato è quello di replicare per quattro volte le istruzioni per il disegno di una riga, ogni volta girando a destra di 90°.



Spunti e riflessioni:



• Si noti come le istruzioni, quando scritte correttamente, cambiano colore e vengon evidenziate. Gli errori più comuni di digitazione sono individuabili semplicemente osservando il colore di cosa abbiamo scritto: in figura l'istruzione **tr** è stata digitata in maniera errata e non è diventata blu.

• I parametri hanno bisogno di uno spazio: "**fw100**" è dunque errato, mentre "**fw 100**" è corretto

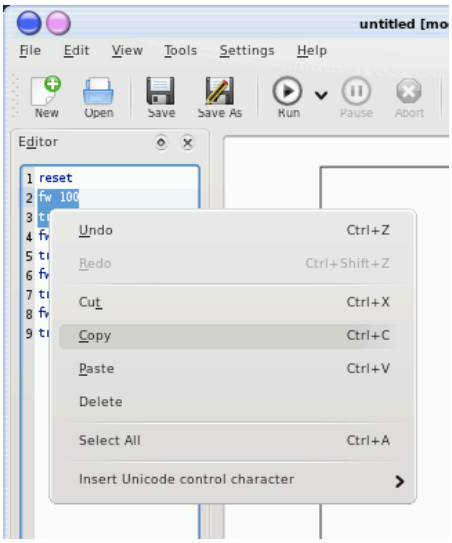
• Si noti come la tartaruga ritorni esattamente al punto iniziale grazie all'ultima rotazione.

• Il numero di pixel 100 (il parametro di forward) può ovviamente essere sostituito con qualsiasi altro valore. Si tenga conto però che la tela di default di kturtle è un quadrato di 400x400 pixel: uscire fuori da questo quadrato vuol dire fare scomparire la tartaruga dallo schermo!

• Scrivere valori molto piccoli invece potrebbe portare a quadrati invisibili o quasi, poichè la tartaruga stessa coprirà il disegno fatto.

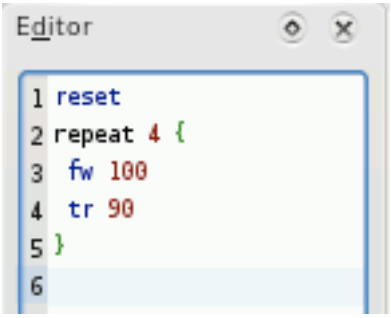
• La somma degli angoli interni del quadrato è 360°, ossia 90° per angolo per 4 angoli.

• Le istruzioni necessarie per fare un quadrato sono 8 (9 con il reset), ossai 2 per ogni lato. E' possibile scriverle velocemente copiando le prime due (seleziona le istruzioni col mouse, poi tasto destro e copia, oppure premi ctrl-c) e poi incollandole per 3 volte (ctrl-v)



Miglioriamo il quadrato

Introduciamo una utilissima istruzione, **repeat**, realizzando un modo più sintetico per realizzare lo stesso quadrato.



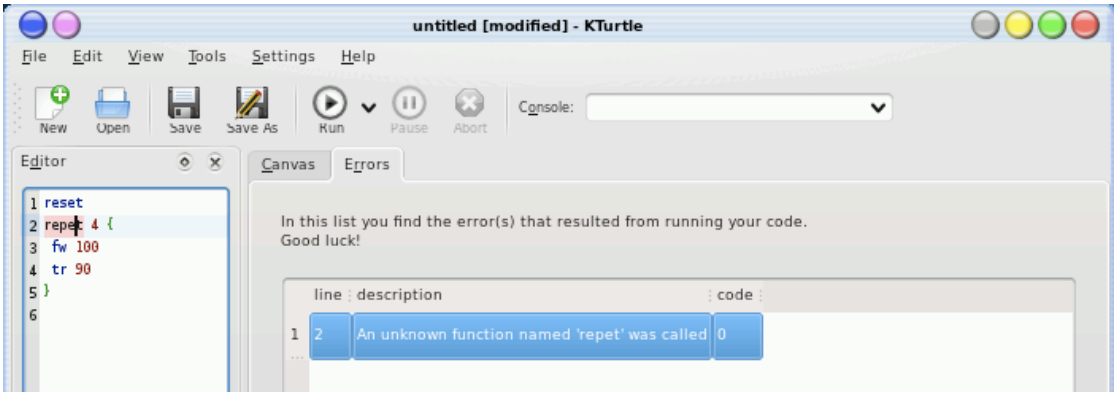
Avendo osservato che le due istruzioni **forward** e **turnright** vanno ripetute 4 volte, invece di riscriverle noi (gli informatici sono pigri) diciamo semplicemente alla tartaruga di... ripeterle per 4 volte!

L'istruzione **repeat** ha come parametro il numero di volte che si deve ripetere un qualcosa (in questo caso 4), e l'inizio e la fine del qualcosa da ripetere. Per indicare l'inizio e la fine del qualcosa da ripetere (una o più istruzioni, o *blocco*), si usano le parentesi graffe. Le parentesi graffe si ottengono normalemente con le seguenti combinazioni di tasti:

PARENTESI	COSA FA	COME SI FA
{	Inizio blocco istruzioni	ALT GR + 7
}	Fine blocco istruzioni	ALT GR + 0

Spunti e riflessioni:

• Ricordarsi di chiudere le parentesi aperte!

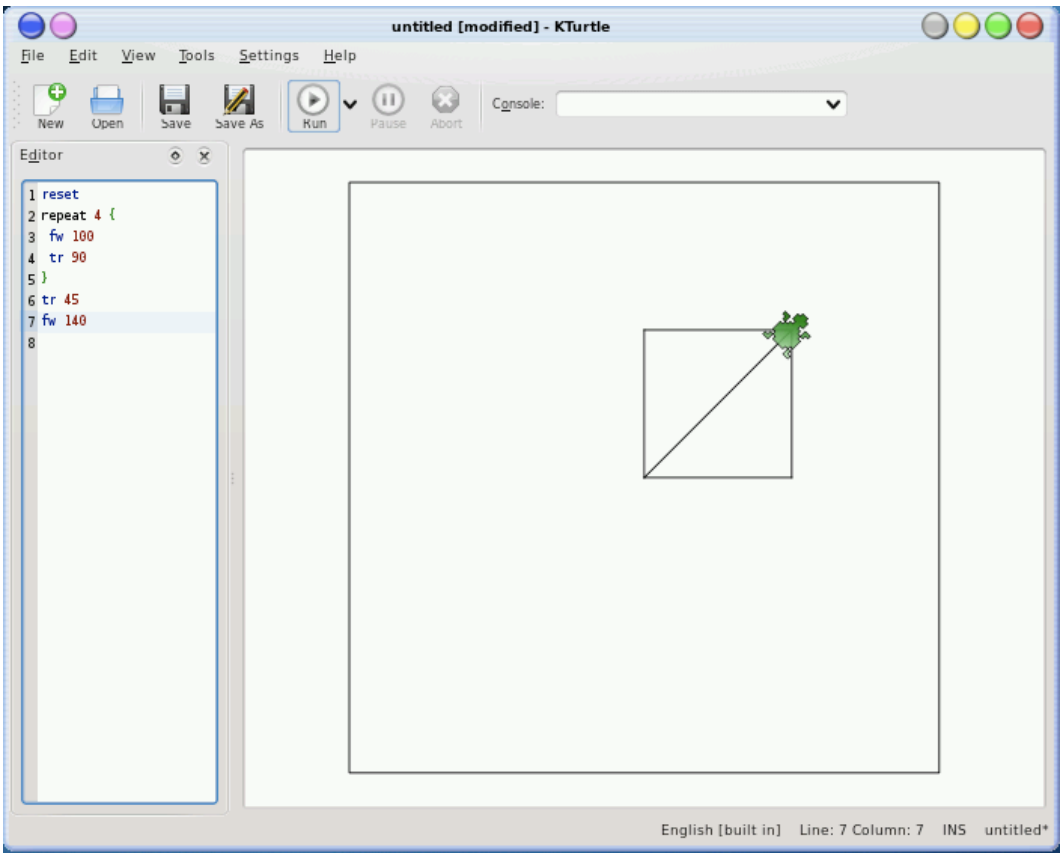


• Verificare cosa succede quando si cerca di eseguire un programma con un errore: kturtle mostrerà l'errore (ed una breve spiegazione dello stesso) in un nuovo tab accanto a quello del canvas. Dopo aver corretto l'errore, ricordarsi di clickare sul tab canvas, poichè kturtle continuerà a mostrare il tab degli errori anche se non ce ne sono più!

• Quando si usa un'istruzione come repeat, che ne racchiude altre, è una buona pratica spostare verso destra il blocco contenuto (come nelle figure mostrate), in modo che il tutto sia più leggibile: questa pratica permette ad esempio di controllare molto semplicemente se tutte le parentesi aperte sono chiuse e si chiama *indentazione*.

• Provare a ripetere più di 4 volte, per esempio 6 o 8. Cosa succede? La tartaruga ridisegna sopra cose già precedentemente disegnate o no?

• Ora proviamo a disegnare la diagonale del quadrato. Se l'angolo retto è 90°, di quanto dobbiamo ruotare per disegnare una diagonale? Osservazione: se il lato del quadrato è 100, anche la diagonale è 100 o è più lunga?



Esercizi

Ispirandosi alle figure mostrate in basso:

- Disegnare un rettangolo
- Disegnare una bandiera
- Disegnare una matita
- Disegnare una sedia