Generazione di numeri primi

Si genera un numero casuale dispari seguito dal test di primalità, ripetendo finché non si ha un primo con probabilità di errore sufficientemente bassa.

Il numero di primi minori di N tende a $\frac{N}{\ln N}$ per $N \to +\infty$, quindi per N sufficientemente grande in un suo in torno di ampiezza $\ln N$ cade mediamente un numero primo, quindi il numero di tentativi che dobbiamo effettuare è polinomiale.

Algoritmo

Generazione di un numero primo (con probabilità di errore $1/4^k$) di almeno n bit.

PRIMO(n, k)

1 $S =$ sequenza casuale di $n-2$ bit
2 $N = 1S1$ // N dispari e con n bit significativi
3 **while** TESTMR$(N, k) == 0$
4 $N = N + 2$
5 **return** N // al massimo $n+1$ bit (tra N e $2N$ ci sono primi)

Costo: $O(n^4)$, eseguendo n volte il test di Miller-Rabin ($O(n^3)$)