

Analisi e progettazione

Sia dato un numero intero $0 < n < 10.000.000.000$, proporre un programma che lo fattorizzi nei suoi fattori primi. Deve essere implementato il tempo di esecuzione del programma mediante la libreria `time`.

L'obiettivo è quello di verificare se il numero inserito in input è un numero primo o meno.

Esempio:

input $\rightarrow 12$
output $\rightarrow 2 * 2 * 3 \rightarrow$ composto

input $\rightarrow 31$
output $\rightarrow 1 * 31 \rightarrow$ primo

Per risolvere questa problematica abbiamo applicato il metodo della "divisione di prova". L'idea essenziale alla base della divisione di prova verifica se un intero n , l'intero da fattorizzare, può essere diviso per ogni numero a sua volta inferiore a n . Ad esempio, per l'intero $n = 12$, gli unici numeri che lo dividono sono 1, 2, 3, 4, 6, 12. Selezionando solo le potenze prime più grandi in questo elenco si ottiene che $12 = 3 \times 4 = 3 \times 2^2$.

In questa situazione abbiamo scritto un algoritmo che va ad applicare il metodo della divisione di prova, innanzitutto creiamo un array $A[]$ dove verranno salvati i fattorizzatori del numero n .

- Per prima cosa abbiamo creato un ciclo *while* che cicla finché il $n \bmod 2 == 0$ se il resto è 0 allora entra nel ciclo *while* e aggiunge nell'array $A[]$ il 2 perché è un fattorizzazione del numero n e poi divido per 2 il numero n per continuare la ricerca dei divisori di n . Il ciclo termina quando il numero non è più divisibile per 2 e quindi aumento il valore del divisore a $f = 3$.
- Inizia il secondo ciclo per trovare i fattori di n , per prima cosa controllo se il fattore $f * f$ sia $\leq n$ perché se il numero è più piccolo del quadrato del fattore allora significa quel numero non potrà essere diviso per quel fattore. Se il numero non entra nel secondo ciclo significa che questo è l'ultimo fattore del numero n e termina la funzione.
- Entrato nel ciclo faccio un controllo se il numero $n \bmod f == 0$ se il resto è uguale a 0 entra nell'if si salva il fatto nell'array $A[]$ e divide n per il fattore f e ricomincia il ciclo, se invece il resto del modulo del controllo è diverso da 0 aumento il fattore di 2.
- Successivamente ricomincia il ciclo con un fattore diverso.
- Nella funzione `main` ho deciso di creare una variabile `start_time` e una variabile `end_time` che mi serviranno per calcolare il tempo di esecuzione della funzione di fattorizzazione.