

Esercitazione 27.10.2022

Programmazione 1

Esercizio 1

- Scrivere un programma C che dato un numero intero N stampi l'elenco di tutti i numeri primi compresi fra 2 e N.
- *Suggerimento.* Utilizzare un array e applicare il classico algoritmo noto come *crivello di Eratostene*.
- Sviluppare il programma secondo l'approccio di decomposizione top-down e raffinamenti progressivi, con analisi, specifica e progettazione delle varie parti

Analisi dei dati

Dati input: N intero

Dati output: p_1, p_2, \dots, p_k dove p_i è l' i -esimo numero primo compreso

Dati intermedi: un array A di $N-1$ interi, un intero i che serve da indic

Specifica

Precondizione: $N \geq 2$

Postcondizione: p_1, p_2, \dots, p_k son tutti e soli i numeri primi fra 2 e N

PROGETTAZIONE

1. Leggi il valore di N
2. Se $N < 2$ termina
3. Inserisci in un vettore A di taglia $N-1$ i valori $2, 3, 4, \dots, N$
4. Applica ad A il crivello di Eratostene
5. Stampa i valori di A non cancellati

Raffinamento passo 3

3.1 Dato un array di interi di riempimento n
per i da 0 a $n-1$
 inserisci $i+2$ in $A[i]$

Raffinamento passo 4

4.1 Dato un array di interi di riempimento n
per i da 0 a $\text{rad-quad}(n+2)$

4.1.1 assegna $A[i]$ a j se $A[i]$ non zero

4.1.2 finché $j * A[i] \leq n+1$

4.2.1 azzerà $A[j * A[i] - 2]$

4.2.2 incrementa j di 1

Raffinamento passo 5

5.1 Dato un array di interi di riempimento
per i da 0 a $n - 1$

5.1.1 se $A[i]$ diverso da zero

5.1.2 stampa $A[i]$

```
void crivello (int A[ ], int n)
{
    int i, j;

    for (i = 0; i*i <= n+1; i++)
    {
        if (A[i] != 0)
            j = A[i];
        else
            continue;
        while (j * A[i] <= n+1)
        {
            A[j * A[i] - 2] = 0;
            j++;
        }
    }
}
```


Esercizio per casa: Realizzare una funzione che dato un array A di n elementi interi ne cancelli gli elementi pari a zero, compattando A con degli shift a sinistra. La funzione deve restituire in numero di elementi rimasti dopo il compattamento.