



---

## Ricerca multithreading di numeri primi

Data la funzione `is_prime()` realizzare un programma multithreading che distribuisce il carico della ricerca dei numeri primi da 1 a `define MAX_NUMBER 1000000` su `define MAX_THREAD 4`

---

```
1 bool is_prime(int n) {
2     if (n <= 1) return false;
3     if (n == 2) return true;
4     if (n % 2 == 0) return false;
5     for (int i = 3; i * i <= n; i += 2) {
6         if (n % i == 0) return false;
7     }
8     return true;
9 }
```

---

## Affinity: associazione thread/core

Dopo aver studiato il sorgente `thread_affinity.c` modificare l'esercizio precedente associando i thread lanciati ai primi `MAX_THREAD` cores. Verificare, da un altro terminale, con `watch -n 0.5 'ps -eLo pid,tid,psr,cmd | grep thread_affinity'`