

# TPA 2023/24 – Order Matrix

Questo è il testo dell'esame del corso "Tecniche di Programmazione Avanzata" del Dipartimento di Ingegneria Industriale, anno accademico 2023/2024.

## Istruzioni

L'esercizio consiste nella progettazione e nell'implementazione di un software C++ e OpenMP per la creazione concorrente e riordino di matrici di interi. Il software deve essere scritto mediante il paradigma di programmazione ad oggetti, quindi utilizzando ad esempio (ove possibile e sensato):

- classi
- incapsulamento
- ereditarietà
- polimorfismo
- passaggio per riferimento
- passaggio per riferimento costante
- puntatori
- metodi const
- operatori
- overload
- valori di default
- template

Nessun codice di partenza è fornito. Lo studente deve occuparsi sia della progettazione che dell'implementazione. Si consiglia di consegnare anche il progetto cartaceo del software, soprattutto se l'implementazione dovesse essere incompleta (ad esempio per mancanza di tempo). Il progetto sarà tenuto in conto nella valutazione finale.

L'implementazione deve includere anche una serie di test per verificare il corretto funzionamento del codice scritto.

# Descrizione

Dovete scrivere un software parallelo che crei e ordini delle matrici di intere. Un processo (producer) deve generare dei numeri interi positivi randomici e due processi (consumer) devono inserire questi numeri nella matrice A se pari e nella matrice B se dispari. I numeri vanno inseriti in ordine row-major (prima di inserire un numero nella seconda riga bisogna aver inserito tutti i numeri della prima riga). Se un numero prodotto è già presente nella matrice non lo si deve inserire.

Quando una matrice è stata riempita, ulteriori numeri prodotti non vengono considerati.

Quando entrambe le matrici sono state riempite si deve procedere all'ordinamento crescente degli elementi di entrambe ( $A[0][0]$  conterrà il valore minore e  $A[N-1][N-1]$  conterrà l'elemento maggiore) e alla stampa su due file separati.

# Implementazione

Per questo esercizio considerate i seguenti valori:

Dimensione matrici (A e B): 10 x 10

Numeri randomici: da 0 a 1000 (interi positivi)

Un processo deve generare il numero randomico e aspettare finché il numero non sia letto da un altro processo.

Due processi, in parallelo, leggono il numero randomico generato (quando disponibile) e lo inseriscono nella matrice A (se pari) o nella matrice B (se dispari). Nell'inserimento si deve controllare se il numero sia già presente. In caso ci sia già un numero uguale all'interno della matrice A o B il numero non viene inserito.

Quando una matrice è completa non occorre inserire altri numeri.

Quando entrambe le matrici sono completate si devono ordinarne gli elementi in maniera crescente e poi stampare le due matrici in due file (A.txt e B.txt), separando gli elementi con uno spazio e aggiungendo un endLine alla fine di ogni riga.

Per scrivere il codice creare una classe che abbia i seguenti metodi:

costruttore: inizializza le matrici e il generatore randomico

fill: riempie le matrici

order: ordina gli elementi di una matrice

Creare inoltre un programma di test (main) che esegua le operazioni richieste e stampi le matrici finali in due file A.txt e B.txt al termine della simulazione.

Ovviamente si devono parallelizzare in OpenMP tutte le operazioni parallelizzabili, garantendo la correttezza del risultato.

## Suggerimenti:

-Scrivere il codice sequenziale e poi passare al parallelo, parallelizzando un'operazione alla volta.

-State attenti alla selezione della matrice dove inserire i valori e al controllo (parallelo) che il valore sia già nella matrice.

## Semplificazioni (con riduzione del voto):

-Chi avesse problemi con la creazione della classe e dei metodi può scrivere tutte le funzioni nel main.

-Chi avesse problemi con l'ordinamento può lasciare le matrici non ordinate.

-Chi avesse problemi con la parallelizzazione può lasciare il codice sequenziale.

## Complicazioni (con aumento del voto):

-Evitare di far aspettare il producer mentre il numero pari o dispari viene inserito nella matrice.

-Ordinare la matrice on-the-fly (finché si inseriscono i valori).