

Linee guida progetto

Emanuele Giona Dipartimento di Informatica, Sapienza Università di Roma

Luca Iezzi Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma

Reti di Calcolatori A.A. 2022/23

Prof.ssa Chiara Petrioli Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma

Emanuele Giona Dipartimento di Informatica, Sapienza Università di Roma

Linee guida per il progetto

2

Dettagli operativi e valutazione:

- Progetto **opzionale**
- Svolto al più in gruppi da **2** persone
- Punteggio bonus rispetto al voto finale: da **0** a **3** punti bonus
- Consegna entro la sessione **giugno-luglio**
- Registrazione tramite Google Form: <https://forms.gle/k1mmdrUaGtMVBquf9>

Linee guida:

- Sviluppato in **C**: solo libreria standard o API POSIX
- **Buone pratiche** di programmazione fondamentali
 - Funzioni standard C sicure, validità puntatori, controlli memory leak, codice chiaro e commentato, ecc.
- Tracce di progetto: sviluppo di una **proposta** oppure **accordata** individualmente
- Durata: orientativamente 2 settimane
- **Nessuna collaborazione** all'infuori dell'eventuale gruppo da 2
- Ogni **condivisione vietata** prima della consegna

Requisiti consegna

3

Il progetto deve essere consegnato includendo:

1. Report in PDF contenente istruzioni di compilazione / esecuzione e motivazioni delle scelte implementative
2. Suite di test per validare il progetto

La consegna è costituita dall'invio di un'email, **allegando il report** e facendo **riferimento alla versione** (tag) del repository GitHub.

Il formato dell'oggetto email deve essere il seguente:

[LabReti] progetto <matricola referente> <eventuale altra matricola gruppo> - **consegna tag:** <tag_id>

Al momento della consegna, il codice sorgente valutato sarà esclusivamente quello risultante dal download del tag specificato nell'email di consegna.

Proposta progetto #1

4

Doubly-linked list thread-safe

- Elementi di tipi uniformi, ma supporto di 2 tipi:
 - unsigned char[8], long double
- Operazioni da supportare
 - `getAt()`
Restituisce il puntatore all'elemento in una data posizione (intera)
 - `insert(), insertAt()`
Inserimento di un elemento (ad una data posizione – intera); restituisce il puntatore all'elemento inserito
 - `remove(), removeAt()`
Rimozione di un elemento (ad una data posizione – intera); restituisce il puntatore all'elemento successivo
 - `map()`
Applica una funzione ad ogni elemento della lista, restituendo il risultato sotto forma dello stesso tipo della lista
 - `reduce()`
Applica una funzione a tutti gli elementi della lista, restituendo un singolo risultato
- Funzionalità di caching: puntatore all'ultimo elemento utilizzato (se ancora presente nella lista)
- Operazioni `map()` e `reduce()` devono sfruttare multi-threading basato su thread-pool
- Thread-safe: tutte le operazioni devono poter essere utilizzate da più thread, in modo trasparente

Proposta progetto #2

5

DBMS per in-memory database

- Elementi fondamentali:
 - struct con 3 membri (chiave, dimensione valore, valore); le chiavi possono solo essere interi positivi
- Operazioni da supportare:
 - Creazione ed accesso a database; in fase di creazione: specifica dimensione da allocare
 - read()
Legge e restituisce il valore associato ad una chiave
 - write()
Scrive (o sovrascrive) un valore associandolo ad una chiave
- Libreria client: permette a programmi terzi di interagire con il DBMS ed i database disponibili
 - Dato uno stesso database, più di un client può operare sullo stesso; va garantita priorità alle operazioni di write()
- Il DBMS viene eseguito prima di ogni altro programma che ne faccia uso
 - I client verificano la raggiungibilità del DBMS, che espone un indirizzo IP per accettare le richieste dei client
 - Un client può *montare* il database in memoria (shared memory) dopo essere stato accettato ed aver ricevuto i dettagli
- Il DBMS permette la configurazione del livello di indicizzazione per le chiavi di un dato database
- Se un database deve aumentare lo spazio allocato, segnala a tutti i client di terminare le operazioni e disconnettersi; dopodiché comunica i nuovi dettagli per ripristinare le operazioni