

ESERCIZIO 3 - TABELLE HASH -

F. MOGAVERO

Implementare una libreria di funzioni in **Linguaggio C++** per la gestione di una struttura dati dinamica di tipo **tabella hash** contenente **dati generici**, ovvero interi, stringhe, ecc. Tale struttura dovrà essere implementata sfruttando le seguenti due **tecniche di risoluzione dei conflitti**: (1) **indirizzamento chiuso / hashing aperto / concatenamento**; (2) **indirizzamento aperto / hashing chiuso**.

Le funzionalità da realizzare sono di seguito elencate:

- (1) **costruzione e distruzione** di una struttura dati;
- (2) operazioni di **assegnamento e confronto** tra istanze diverse della struttura dati;
- (3) operazioni specifiche della struttura dati: **inserimento/eliminazione** di un elemento; **ridimensionamento** della tabella;
- (4) operazioni generiche sulla struttura dati: **controllo di esistenza** di un dato valore; test di **vuotezza**; lettura della **dimensione**; **svuotamento** della struttura.

Al fine di poter testare adeguatamente il funzionamento delle librerie sopra descritte, si richiede di definire (esternamente alle stesse, in un opportuno file di test richiamato dal “main”) un insieme di procedure di **test unitario**. In aggiunta a questo, è necessario prevedere l’accesso alla funzionalità di test prevista dal docente.

Il codice sorgente prodotto dovrà seguire pedissequamente (sia nei nomi delle funzioni di libreria, sia nella strutturazione, gerarchia di classi e nei nomi delle diverse directory e file “.cpp” e “.hpp”) la forma riportata nel template Exercise3.zip associato al presente esercizio.

Per lo sviluppo delle librerie, in aggiunta alle spiegazioni del docente, si faccia riferimento alla sezione 9.4 di [1].

RIFERIMENTI BIBLIOGRAFICI

- [1] Clifford A. Shaffer. *Data Structures and Algorithm Analysis*, Edition 3.2 (C++ Version), 2013.