# ASD, Laboratorio 5

## Implementazione

Siano **U** l'insieme delle stringhe che rappresentano parole di senso compiuto nella lingua inglese ed **E** l'insieme delle stringhe che rappresentano definizioni di parole di senso compiuto della lingua inglese.

Si implementi il TDD “Dizionario di coppie **(k, e)** in cui **k** appartiene a **U** ed **e** appartiene ad **E**” mediante la struttura dati “tabella di hash con liste di collisione”.

Il file dictionary-hashtable.cpp contiene molte funzioni già implementate, tra le quali createEmptyDict() (che vi può aiutare a capire come è fatta una variabile di tipo Dictionary) e tutte le funzioni di lettura e di stampa, inclusa la stampa di informazioni statistiche sull'organizzazione del dizionario. Voi dovete implementare:
- h1(), funzione di hash che considera unicamente il valore ascii del primo carattere della chiave (se esiste) e restituisce il resto della divisione di tale valore per tableDim
- h2(), funzione di hash che somma il codice ascii di ogni carattere nella chiave e restituisce il resto della divisione di tale somma per tableDim
- h3(), funzione che dovete inventare voi seguendo qualche criterio ragionevole e che motiverete nel file di comprensione dei dati sperimentali
- insertElem()
- deleteElem()
- search()

Nel file string-utility.cpp sono inoltre implementate funzioni per “nomalizzare” le chiavi, rendendo tutti i caratteri minuscoli ed eliminando gli spazi. Potete trarre ispirazione da queste funzioni per implementare le funzioni di hash.

Notate che **l'ordine in cui sono disposte le funzioni nel file dictionary-hashtable.cpp non corrisponde necessariamente all'ordine in cui le dovete implementare**. In particolare, come avete già riscontrato in altre occasioni, la prima funzione da implementare è l'aggiunta di un elemento al dizionario (**insertElem**) senza la quale non è possibile effettuare la lettura da file dei dati.

## Sperimentazione

I file tempi-di-esecuzione-operazioni-dict.xls e organizzazione-hash-table.xls contengono tabelle da completare con i risultati degli esperimenti. Tra i vari esperimenti proposti ci sono anche quelli che prevedono l'uso di strutture dati diverse dalla tabella di hash per la realizzazione del TDD Dizionario. Tali strutture dati sono già state implementate dai docenti. Voi dovete solo usarle e riportare i dati sperimentali sul file tempi-di-esecuzione-operazioni-dict.xls.

Negli esperimenti che riguardano l'implementazione delle hash table vi viene richiesto di cambiare sia la dimensione della tabella (const int tableDim nel file header), sia la funzione di hash adottata.

I comandi da utilizzare per compilare il programma sono riportati all'inizio del file dictionary-main.cpp . Prestare attenzione ai flag di compilazione indicati, specialmente i

flag -std=c++11 e i flag che iniziano con "-D", i quali servono a selezionare le tre diverse implementazioni del TDD da mettere a confronto, ossia: quella basata su vettore ordinato, quella basata su lista ordinata, e quella basata su tabella hash.

Nel file organizzazione-hash-table.xls si fa riferimento esclusivamente alla implementazione con hash table e si devono riportare sul file excel il numero di elementi memorizzati, il numero di bucket e la deviazione standard che viene calcolata e stampata dalla funzione print.

Lo scarto quadratico medio è uno dei modi per esprimere la dispersione dei dati intorno ad un indice che nel nostro caso è la media aritmetica. In pratica, maggiore è lo scarto quadratico medio, tanto più “diversi dal valor medio” sono i dati.
In statistica lo scarto quadratico medio rilevato su un insieme di N dati indicati con $x_1$, $x_2$, ..., $x_N$ si definisce come:

$$\sigma_X = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N}},$$

dove

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

è la media aritmetica dei dati $x_1$, $x_2$, ..., $x_N$.

Nei nostri esperimenti lo scarto quadratico medio serve a misurare quanto gli elementi (chiave-valore) si sono posizionati in modo uniforme nei vari bucket. Se lo scarto quadratico medio è basso, allora i dati sono “ben distribuiti”. Se è alto, vuole dire che ci sono fenomeni di agglomerazione, con bucket sovraffollati ed altri vuoti. In questo caso, significa che la funzione di hash adottata non è appropriata perché non distribuisce abbastanza uniformemente le chiavi nei bucket.

## Comprensione dei dati sperimentali

Nel file comprendiamoGliEsperimenti.doc sono riportate diverse domande che vi guidano nella comprensione del lavoro che avete svolto e dei risultati che avete ottenuto. **E' molto importante che sappiate rispondere**: solo se sapete rispondere avete davvero capito cosa avete fatto durante questo laboratorio.

I file comprendiamoGliEsperimenti.doc, tempi-di-esecuzione-operazioni-dict.xls e organizzazione-hash-table.xls vanno tutti consegnati insieme al codice sviluppato in un unico file .zip.