

# Laboratorio 8

## Esercizio 3

### Strutture dati utilizzate:

- **TitoloList**: adt di prima classe, implementato come una semplice lista concatenata non ordinata. Contiene item di tipo Titolo.
- **Titolo**: quasi-adt, contiene il nome del titolo rappresentato e il puntatore al BST contenente le quotazioni relative al titolo in questione.
- **QuotazioniBST**: adt di prima classe, ogni nodo del bst contiene una determinata quotazione giornaliera. I nodi sono ordinati in base alla data alla quale si riferisce la quotazione giornaliera in essi contenuta.
- **Quotazione**: quasi-adt, rappresenta una quotazione giornaliera e contiene la data, il valore e il numero di transazioni ad essa associati.
- **Data**: quasi-adt rappresentante una data.

### Strategia risolutiva:

1. Il main inizializza una **TitoloList** e chiama la funzione **readFile**.  
La lettura del file e il conseguente popolamento delle strutture dati viene effettuato nel seguente modo: quando si incontra un nuovo titolo non presente nella lista esso viene inizializzato ed aggiunto nella lista. Per ogni transazione letta si richiama la funzione **TITOL0addTransaction** che ricevendo come parametro il titolo alla quale è associata la transazione, il valore della transazione e il numero di titoli venduti aggiorna la quotazione giornaliera relativa alla data della transazione.  
Nello specifico **TITOL0addTransaction** se non sono presenti nodi relativi alla data della transazione passata come parametro, aggiorna il bst contenuto nel quasi adt **Titolo** creando un nuovo nodo e associandogli un elemento **quotazione** appena inizializzato, altrimenti legge dal bst la quotazione già presente e l'aggiorna chiamando la funzione **QUOTAZIONEaddTransaction**.
2. La ricerca di un titolo azionario è gestita come una semplice ricerca in una lista concatenata. (complessità  $O(N_{titoli})$ )
3. La ricerca di una quotazione in una certa data per un determinato titolo è una comune ricerca in un BST. (complessità  $O(\text{Altezza albero})$ )
4. La ricerca della quotazione minima e massima in un intervallo di date viene effettuata con una visita in-order del BST nella quale vengono prese solo in considerazione le quotazioni con data compresa nell'intervallo (complessità  $O(N_{quotazioni})$ ). La visita

per una questione di ottimizzazione si interrompe una volta superata la data dell'estremo superiore dell'intervallo.

5. La ricerca della quotazione minima e massima lungo tutto il periodo registrato viene ricondotta al punto precedente richiamando la funzione di ricerca lungo il più ampio intervallo consentito.
6. Le dimensioni del cammino più corto e di quello più lungo dell'albero binario ricavate ricorsivamente assegnando 0 come altezza del nodo sentinella e il massimo (o il minimo) tra l'altezza dei figli come altezza di ogni altro nodo. L'eventuale bilanciamento del BST viene effettuato applicando ricorsivamente il partizionamento attorno alla chiave mediana.