

Lecture
8_8.1

Dizionari



Paolo PRINETTO

Politecnico di Torino (Italy)
Univ. of Illinois at Chicago, IL (USA)
CINI Cybersecurity Nat. Lab. (Italy)

Paolo.Prinetto@polito.it

www.consorzio-cini.it

www.comitato-girotondo.org

License Information

This work is licensed under the
Creative Commons BY-NC
License



To view a copy of the license, visit:
<http://creativecommons.org/licenses/by-nc/3.0/legalcode>

Disclaimer

- **We disclaim any warranties or representations as to the accuracy or completeness of this material.**
- **Materials are provided “as is” without warranty of any kind, either express or implied, including without limitation, warranties of merchantability, fitness for a particular purpose, and non-infringement.**
- **Under no circumstances shall we be liable for any loss, damage, liability or expense incurred or suffered which is claimed to have resulted from use of this material.**

Goal

- This lecture aims at presenting how Dictionaries as usually implemented.

Prerequisites

- **Lectures:**
 - **8_1.1 Pointers & Dynamic Memory**
 - **8_4.1 Introduction to ADT**
 - **8_5.1 Lists**
 - **8_7.1 Sorting Algorithms**

Outline

- Introduzione
- I Dizionari come ADT
- Implementazioni dei Dizionari

Outline

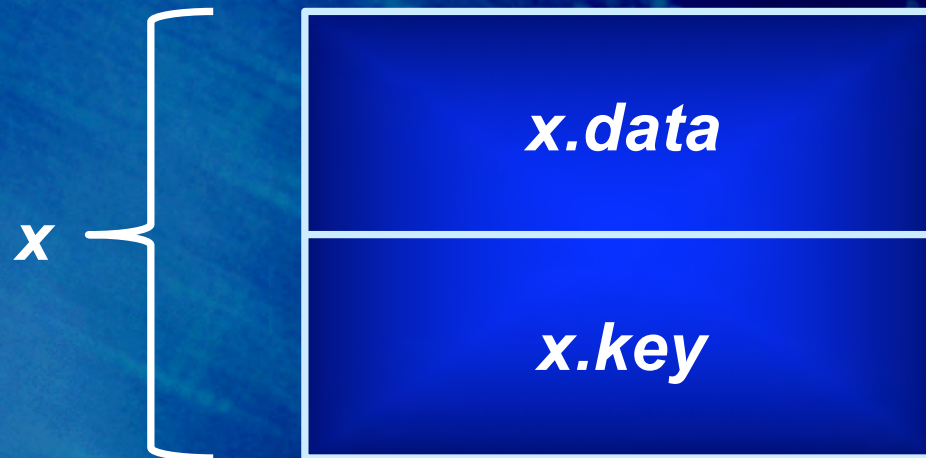
- **Introduzione**
- **I Dizionari come ADT**
- **Implementazioni dei Dizionari**

Introduzione

- **Il problema da affrontare è la verifica dell'appartenenza di un elemento a un determinato insieme, ossia della ricerca di un elemento all'interno di un insieme**

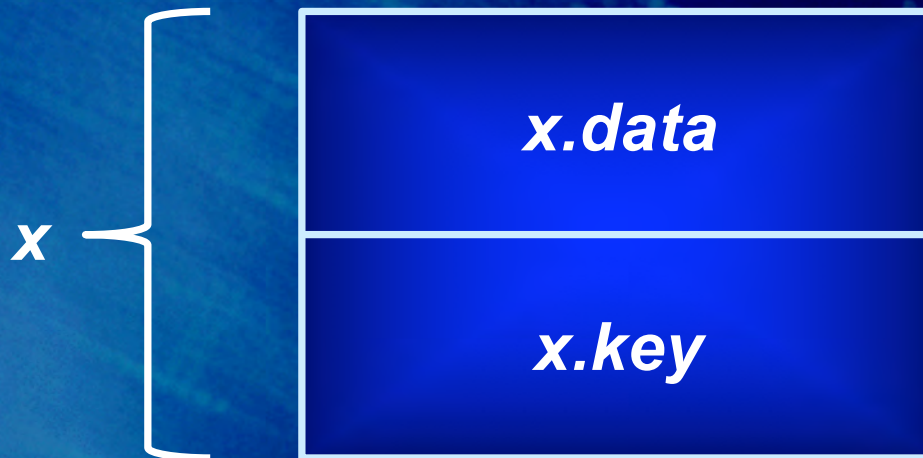
Assunzioni di base

- Tra i campi di ciascun elemento x dell'insieme è individuabile un campo **chiave** (**key**), il cui contenuto caratterizza l'elemento stesso:



Assunzioni di base

- Tra i campi di ciascun elemento x dell'insieme è individuabile un campo **chiave** (*key*), il cui contenuto caratterizza l'elemento stesso:



- Sulle chiavi è definibile una relazione di ordine (parziale, totale o lessicografico)

Soluzione

- Si ricorre ad ADT particolari, e precisamente a:
 - **Dizionari:**
nel caso in cui sia richiesta la ricerca di un elemento qualsiasi dell'insieme
 - **Code prioritarie:**
nel caso in cui sia richiesta la ricerca di un elemento particolare, vale a dire quello avente chiave minima (o massima)

Caveat

- **Nei casi in cui non sia identificabile alcuna chiave caratterizzante gli elementi dell'insieme occorre ricorrere ad altri ADT, la cui trattazione esula dagli scopi di questo corso.**

Outline

- Introduzione
- **I Dizionari come ADT**
- Implementazioni dei Dizionari

Dizionari

- I dizionari vengono solitamente classificati in base a:
 - *operazioni possibili* sul dizionario
 - *unicità delle chiavi* degli elementi

Dizionari

- I dizionari vengono solitamente classificati in base a:
 - **operazioni possibili** sul dizionario
 - **unicità delle chiavi** degli elementi

- **statico:**
se non sono ammesse operazioni di inserimento o cancellazione di elementi
- **dinamico:**
se sono ammesse sia la cancellazione sia l'inserimento
- **semi-dinamico:**
se è ammesso l'inserimento, ma non la cancellazione

Dizionari

- I dizionari vengono solitamente classificati in base a:
 - **operazioni possibili** sul dizionario
 - **unicità delle chiavi** degli elementi

- **a chiavi uniche:**
ciascun elemento ha una chiave diversa da tutte le altre
- **a chiavi multiple:**
più elementi possono essere caratterizzati da una stessa chiave

Nota

- **Nel seguito si tratteranno esclusivamente dizionari a chiave unica.**
- **Le considerazioni che verranno fatte via via sono comunque estendibili al caso di dizionari con chiavi multiple, ricorrendo a opportune soluzioni implementative.**

Il dizionario come ADT

- **Sull'ADT dizionario sono definite le seguenti operazioni:**

insert (x, D)

**aggiunge al dizionario D
l'elemento x , verificando che la
chiave $x.key$ sia unica**

Ritorna:

- **OK** se l'inserimento è avvenuto
- **ERROR** altrimenti



delete (key, D)

cancella dal dizionario *D*
l'elemento avente chiave *key*

Ritorna:

- OK se la cancellazione è avvenuta
- ERROR altrimenti



search (key, D)

**Ricerca nel dizionario D il record
avente chiave key**

Ritorna:

- **Il record x avente chiave key**
- **ERROR nel caso in cui
l'elemento non sia presente**



next (key, D)

Ricerca in *D* l'elemento con la più piccola chiave $> key$

Ritorna:

- L'elemento avente chiave più piccola tra le chiavi $> key$
- ERROR nel caso in cui l'elemento non sia presente



next (key. D)

*Permette una facile
scansione
sequenziale degli
elementi dell'insieme*



min (D)

**Ritorna la più piccola delle chiavi
memorizzate in *D***



Outline

- Introduzione
- I Dizionari come ADT
- Implementazioni dei Dizionari

Implementazione dei dizionari

- **Due i problemi principali da affrontare nell'implementazione dei dizionari:**
 - **la scelta della struttura dati più idonea**
 - **l'individuazione del metodo di ricerca più idoneo**
- **I due aspetti sono strettamente interdipendenti e funzione:**
 - **della ricerca da effettuare**
 - **della tipologia dell'insieme degli elementi.**

Implementazione dei dizionari

- Si ricorre a soluzioni diverse:
 - *Vettori*
 - *Liste*
 - *Tabelle di Hash*
 - *Alberi*

Implementazione dei dizionari

Dimensioni

Piccole

vettori

**vettori
liste**

**vettori
liste**

Grandi

**vettori
hash table**

**alberi
hash table**

alberi

Statici

Semi-dinamici

Dinamici

Tipo

Малые Автюхи, Калининский район, Республики Беларусь

