HASH TABLES

1. Introduction

2. Dictionary data type

Definition 2.1. Let $\mathcal{X}$ be a set of *items* and let $\mathcal{U}$ be a set of *keys*. Consider an abstract data type D which dynamically stores a collection of pairs (k, x) where $k \in \mathcal{U}$ and $x \in \mathcal{X}$ in such a way that D does not store two pairs having the same key at the same time. Moreover, we assume that D supports the following operations.

INSERT$\big(D, (k, x)\big)$
Adds pair (k, x) into D if there is no other pair stored in D with k as a first entry.

DELETE$\big(D, k\big)$
Removes a pair with k as a first entry from D if such pair is stored in D.

SEARCH$\big(D, k\big)$
Returns x if a pair (k, x) is stored in D. Otherwise returns *nil*.

An abstract data type with these properties and interface is called *an associative array* or *a dictionary*.

Definition 2.2. Let $\mathcal{X}$ and $\mathcal{U}$ be sets. *Dictionary problem for $\mathcal{X}$ and $\mathcal{U}$* is the task of designing a dictionary with $\mathcal{X}$ as the set of items and $\mathcal{U}$ as the set of keys.

3. Hash functions

In this section we introduce the important notion of a hash function and we discuss some probabilistic properties of it.

Definition 3.1. Let $\mathcal{U}$ be a set. *A hash function* is a mapping $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ where $m \in \mathbb{N}_+$. Given a has function h *a collision* is a pair of keys $k_1, k_2 \in \mathcal{U}$ such that $k(k_1) = h(k_2)$.

Definition 3.2. Let X be a set and let $n \in \mathbb{N}_+$. Then a set

$$X^{\wedge n} = \big\{(x_1, ..., x_n) \in X^n \,\big|\, \forall_{1 \le i < j \le n}\, x_i \neq x_j\big\}$$

is called *antisymmetric cartesian power of X*.

Definition 3.3. Let $\mathcal{U}$ be a measurable space. We consider $\mathcal{U}^{\wedge n}$ as the measurable subspace of a product space $\mathcal{U}^n$. Suppose that P is a probability distribution on $\mathcal{U}^{\wedge n}$. Let $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ be a hash function for some $m \in \mathbb{N}_+$. Suppose that

$$P\big(h(k_i) = h(k_j) \,\big|\, (k_1, ..., k_n) \in \mathcal{U}^{\wedge n}\big) = \frac{1}{m}$$

for every pair of distinct elements $i, j \in \{1, ..., n\}$. Then h is *a simple uniform hashing with respect to P*.

Example 3.4. Let $\mathcal{U} = [0, m]$ for some $m \in \mathbb{N}_+$. Then $\mathcal{U}$ is a measurable space with respect to Borel algebra $\mathcal{B}\big([0, m]\big)$. We define a hash function $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ by formula

$$h(x) = \lfloor x \rfloor$$

Then h is a simple uniform hashing with respect to the normalization of n-dimensional Lebesgue measure on $[0, m]^{\wedge n}$.

Definition 3.5. Let $\mathcal{U}$ be a measurable space. We consider $\mathcal{U}^{\wedge n}$ as the measurable subspace of a product space $\mathcal{U}^n$. Suppose that P is a probability distribution on $\mathcal{U}^{\wedge n}$. Let $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ be a hash function for some $m \in \mathbb{N}_+$. Fix a real number $\epsilon > 0$ and suppose that

$$P\big(h(k_i) = h(k_j) \,\big|\, (k_1, ..., k_n) \in \mathcal{U}^{\wedge n}\big) = \frac{1}{m} + \epsilon$$

for every pair of distinct elements $i, j \in \{1, ..., n\}$. Then h is *a simple ϵ-uniform hashing with respect to P*.

Example 3.6. Let $\mathcal{U} = \{0, 1, ..., m^2 - 1\}$ for some $m \in \mathbb{N}_+$. Then $\mathcal{U}$ is a measurable space with respect to the power algebra $\mathcal{P}\big(\{0, 1, ..., m^2 - 1\}\big)$. Consider $\mathcal{U}^{\wedge n}$ as a probability space with respect to distribution describing random sampling of n-elements without replacement from $\mathcal{U}$. We define a hash function $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ by formula

$$h(x) = x \bmod m$$

where m is a divisor of N. Fix distinct $i, j \in \{1, ..., n\}$ and note that

$$P\big(h(k_i) = h(k_j) \,\big|\, (k_1, ..., k_n) \in \mathcal{U}^{\wedge n}\big) = \frac{m}{m^2} = \frac{1}{m}$$

4. Hash tables with chaining as a solution to dictionary problem

In this section we present the solution to the dictionary problem and discuss its efficiency.

Definition 4.1. Let $\mathcal{U}$ and $\mathcal{X}$ be sets. Let $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ be a hash function for some $m \in \mathbb{N}_+$. We consider an m-element array D_h such that $D_h[i]$ is a linked list storing values from $\mathcal{U} \times \mathcal{X}$ for every $i \in \{0, 1, ..., m-1\}$. We describe dictionary operations.

INSERT$\big(D_h, (k, x)\big)$
Inserts pair (k, x) to the linked list $D_h[h(k)]$ as its new head.

DELETE$\big(D_h, k\big)$
Deletes a pair with first entry k from the linked list $D_h[h(k)]$.

SEARCH$\big(D_h, k\big)$
Searches for the pair with the first entry k in the list $D_h[h(k)]$. If such pair is found, then returns its second entry. Otherwise returns *nil*.

Then D_h together with these operations is a solution of dictionary problem for $\mathcal{U}$ and $\mathcal{X}$. We call it *the hash table with collisions resolved by chaining*.

For the sequel we need the notion of sampling without replacement.

Definition 4.2. Let $\big(\Omega, \mathcal{F}, P\big)$ be a probability space and let $\mathcal{U}$ be a measurable space. We fix a random variable $\mathcal{K} : \Omega \to \mathcal{U}$ and consider independent and identically distributed random variables $\mathcal{K}_1, ..., \mathcal{K}_n : \Omega \to \mathcal{U}$ for $n \in \mathbb{N}_+$ with distributions identical to $\mathcal{K}$. Define

$$\Delta = \big\{(k_1, ..., k_n) \in \mathcal{U}^n \,\big|\, \forall_{i \neq j}\, k_i \neq k_j\big\}$$

Then $\langle \mathcal{K}_1, ..., \mathcal{K}_n \rangle : \Omega \to \mathcal{U}^n$ is a random variable which gives rise to a probability distribution on a measurable space $\mathcal{U}^n$. Hence it also gives rise to a probability distribution $\mu_{\mathcal{K},n}$ on its subspace $\mathcal{U}^n \setminus \Delta$. Then $\mu_{\mathcal{K},n}$ is *the distribution of sampling without replacement with respect to $\mathcal{K}$*.

Definition 4.3. Let $n \in \mathbb{N}_+$ be the number of elements stored in D. Then $\alpha = \frac{n}{m}$ is called *load factor*.

Theorem 4.4. *Let $\mathcal{U}$ be a measurable space and let $\mathcal{X}$ be a set. Let $h : \mathcal{U} \to \{0, 1, ..., m-1\}$ be a hash function for some $m \in \mathbb{N}_+$. Suppose that $\mathcal{K}$ is a random variable with target $\mathcal{U}$ such that h is a simple uniform hashing with respect to $\mathcal{K}$. For a sequence of keys chosen randomly according to distribution $\mu_{\mathcal{K},n}$ and stored in the hash table D_h the expected time of an unsuccessful search is $\Theta(\frac{n}{m})$.*

Proof. ∎