Class Notes CSE462

Dr. Rajasekaran

19 Feb 2007
## Contents

1 Static Search of Data Items in $O(1)$ time 5  
1.1 Problem Statement: ................................. 5  
1.2 Level-1 Hashing ................................. 6  
1.3 Level-2 Hashing ................................. 6  
1.4 Space Complexity ................................. 6

2 Probabilistic Approach 9  
2.1 Introduction: ........................................ 9  
2.1.1 Idea-1 ........................................ 9  
2.1.2 Idea-2 ........................................ 9  
2.2 Examples: ........................................ 9  
2.2.1 Example-1 ....................................... 9  
2.2.2 Example-2 ....................................... 10  
2.2.3 Example-3 ....................................... 10  
2.2.4 Example-4 ....................................... 11
Chapter 1

Static Search of Data Items in $O(1)$ time

1.1 Problem Statement:

INPUT: A sequence of $k_1, k_2, \cdots, k_n \in M, M = [0, 1, 2, \cdots, m-1]$
OUTPUT: Develop a static datastructure with Constant($O(1)$) search time for any key $k_i$.

SOLUTION: The Idea is based on two levels of hashing. Let $S \subseteq M$ be any input set with $|S| = s$. Let $p > m$ be a prime number (we can always find a prime number between $[m, 2m]$ according to number theory). For any $1 \leq k \leq (p-1)$ let $h_k(x) = (kx \mod p) \mod n$. $B_i(k, n, S)$ be set of elements of $S$ that have a hash value $i$ under $h_k$, $B_i(k, n, s) = \{x \in M : h_k(x) = i\}, i = 0, 1, 2, \cdots, n-1$

**Lemma:** The number of pairs $(x, y), x \in M, y \in M$ from $S$ that collide under $h_k$

\[
\sum_{k=1}^{m-1} \sum_{i=0}^{n-1} C_{2}^{B_i(k,n,S)} \leq \frac{ms^2}{n}
\]

**Corollary:** For any $S \subseteq M$ there exists a $k$ such that

\[
\sum_{i=0}^{n-1} C_{2}^{B_i(k,n,S)} \leq \frac{s^2}{n}
\]
1.2 Level-1 Hashing

In the first level of hashing we use \( n = s \). Then according to above corollary, \( \exists k \) such that the number of pairs \((x, y)\) from \( S \) that collide under \( h_k \) is

\[
\sum_{i=0}^{n-1} C_2^{B_i(k,n,S)} \leq \frac{s^2}{s} = s.
\]

From this we can conclude that the total space used for level-1 hashing is of the \( O(s) \).

1.3 Level-2 Hashing

Now we have \( B_i(k, n, S) \) elements (lets call them as \( S_i \) and \(|S_i| = s_i\)) in every bucket which need to be hashed. At this stage we proceed on the same lines as in the previous step (level-1 hashing). The only difference is that at level-1 we chose the number of buckets to be \( n = s \), now with in the bucket \( i \) of level-1 we create a hash with size \( n_i \), we choose \( n_i = |B_i(k, n, S)|^2 \), i.e square of the number of elements within the bucket. Lets call the buckets in level-2 as \( B'_i \). Now applying the same corollary to this level implies \( \exists k \) \( (h'_k \text{ at this level call it } h'_k) \) such that the number of collisions is given by the following

\[
\sum_{i=0}^{n-1} C_2^{B'_i(k,n_i,S_i)} \leq \frac{s_i^2}{n_i}
\]

Since \( s_i = |B_i(k,n,S)| \) and \( n_i = |B_i(k,n,S)|^2 \) the above inequality (which is nothing but the total number of collisions) turns as

\[
\sum_{i=0}^{n-1} C_2^{B'_i(k,n_i,S_i)} \leq 1
\]

Thus we will not have any collisions within the bucket, Since we dont have collisions at second level the total time to find a key is \( O(1) \). In summary we need to preprocess the \( S \) to determine \( h_k \) at level-1 and \( h'_i, i = 0, 1, 2 \cdots n \) which satisfy the inequality in the corollary to make sure that it takes only \( O(1) \) time to find a key.

1.4 Space Complexity

At level-1 we know that we use space \( n = s \) or \( O(s) \) space. At level-2 we use a space of \( \sum_{i=0}^{n-1} n_i - 1 |B_i(k,n,S)|^2 \). But we know the following from level-1

\[
\sum_{i=0}^{n-1} C_2^{B_i(k,n,S)} \leq s \Rightarrow \sum_{i=0}^{n-1} (B_i(k,n,S))(B_i(k,n,S) - 1) \leq 2s
\]
1.4. SPACE COMPLEXITY

\[ \Rightarrow \sum_{i=0}^{n-1} |B_i(k, n, S)|^2 \leq 2s + \sum_{i=0}^{n-1} |B_i(k, n, S)| \leq 3s \Rightarrow \sum_{i=0}^{n-1} |B_i(k, n, S)|^2 \leq 3s \]

From the above the space at level-2 is \( O(s) \), so the total space complexity is \( O(s) \)
CHAPTER 1. STATIC SEARCH OF DATA ITEMS IN O(1) TIME
Chapter 2

Probabilistic Approach

2.1 Introduction:

Probabilistic approach is used to prove the existence of objects with certain properties, it is based on the following two ideas.

2.1.1 Idea-1

If $X$ is a random variable with mean $\mu$ then there is a value for $X$ that is $\leq \mu$ and there is a value for $X$ that is $\geq \mu$.

2.1.2 Idea-2

If $U$ is a universe of objects and if the Prob[ a random object from $U$ has a property $P$] $> 0$, then $\exists$ an object in $U$ that has property $P$.

2.2 Examples:

The following examples use the above stated ideas to prove the existence of a certain property.

2.2.1 Example-1

Let $G(V, E)$ be a graph, $|V| = n$, $|E| = m$, then there is a partitioning of $V$ into $V_1, V_2$ such that the number of edges from $V_1$ to $V_2 \geq \frac{m}{2}$

Proof: For every $u \in V$ flip a coin depending on the outcome place it in $V_1$ if you get a head, or place it in $V_2$ if you get a tail. For this partitioning
if $e$ is any edge in $E$ then
$\Rightarrow$ Prob[$e$ goes from $V_1$ to $V_2$] = $\frac{1}{2}$
$\Rightarrow$ Expected number of edges ($\mu$) = $\frac{m}{2}$, so clearly from Idea-1 $\exists$ a partitioning such that the number of edges from $V_1$ to $V_2$ is $\geq \mu = \frac{m}{2}$

\textbf{2.2.2 Example-2}

Let $f$ be a boolean formula in CNF (Conjunctive Normal Form).

$$F = F_1 \land F_2 \land F_3 \land \cdots \land F_m$$

Claim is there is an assignment of $m$ classes to $n$ variables that will satisfy at least $\frac{m}{2}$ classes.

Proof: Give a Random assignment to the variables. Let $F_i$ be a class with $k$ literals in it.
$\Rightarrow$ Prob[$F_i$ is not satisfied] = $\frac{1}{2^k} \leq \frac{1}{2}$
$\Rightarrow$ Prob[$F_i$ is satisfied] $\geq \frac{1}{2}$
$\Rightarrow$ Expected Number of classes ($\mu$) is $\geq \frac{m}{2}$
From Idea-1 $\exists$ an assignment that satisfies in $\geq \frac{m}{2}$ classes

\textbf{2.2.3 Example-3}

Let $C_n$ be a complete graph on $n$ vertices, then the smallest $n'$ such that any coloring of the edges will result in $C_k$ that is red or $C_t$ that is black is called the Ramsey Number $R(k, t)$.

Our problem is to derive the lower bound of the Ramsey Number.

Proof: Color the edges of $C_n$ Randomly with red or black with equal probability.

$\Rightarrow$ Prob[that a subset is unicolored] = $2 \times 2^{1-C_k^k}$
$\Rightarrow$ Prob[$\exists$ a subset of $k$ nodes that have unicolor] $\leq C_k^n \times 2^{1-C_k^k}$
$\Rightarrow$ if $C_k^n \times 2^{1-C_k^k} \leq 1$ then
$\Rightarrow$ Prob[that there is no unicolor subset of size $k$] $> 0$
Now we use Idea-2 and it follows that $\exists$ a coloring for which $C_n$ does not have a unicolored $C_k$ this proves the lemma $R(k, k) > n$.

$$\frac{n \times (n - 1)}{2} < 2^{C_k^k - 1}$$

Using stirlings approximation the following fact follows

$$C_k^a \leq \left(\frac{a \times e}{b}\right)^b \Rightarrow \left(\frac{ne}{k}\right)^k \times 2^{-\frac{k(k-1)}{2}} < 1$$
2.2. EXAMPLES:

\[
\Rightarrow \left( \frac{ne}{k} \right) < 2^{k-1} \Rightarrow R(k, k) = n \geq \frac{k \cdot 2^{k-1}}{e \cdot \sqrt{(2)}}
\]

2.2.4 Example-4

An \((n, \alpha, d, c)\) OR-CONCENTRATOR is a bipartite graph, \(G(L, R, E)\) where \(|L| = |R| = n\) and the degree of each node is \(d\). For every subset \(S\) of \(L\), with \(|S| \leq \alpha n\) the number of neighbours of \(S\) in \(R\) is \(\geq C|S|\), \(C > 1\), \(\alpha, c, d\) are constants.

Lemma: 1 \(\exists\ an\ (n, \frac{1}{3}, 18, 2)\ OR-CONCENTRATOR\ \forall\ n \geq n_0\ where\ n_0\ is\ some\ constant.\)

PROOF: To be continued in next class