|  |
| --- |
| **CHAPTER 5 HOMEWORK** |

**PROBLEM 1:**

The dynamic set S is represented by a direct address table T of length m.

To find the maximum elements of S, each time get an element T[k] of the direct address table T and compare it with the variable ‘max’.

If T[k] is greater than ‘max’, store T[k] in ‘max’. Repeat this until all the elements of the direct address table are compared with ‘max’. The last value that is left in ‘max’ is required maximum value.

A procedure to find the maximum element of S:

MAXOFS (T, m)

1. max = -
2. for k = 0 to m – 1
3. if T[k] NULL and max < T[k]
4. max = T[k]
5. return max

If the m – 1 slot isn’t empty, then the procedure checks all the m slots of direct address table T for maximum value. So, the worst case time complexity of the procedure is **O (m)** time.

**PROBLEM 2:**

For each pair of keys k, l, where k ≠ l, define the indicator random variable Xlk = l {h (k) = h (l)}.

Since we assume simple uniform hashing, Pr{Xlk =1} = Pr{h(k) = h(l)} = 1/m, and so E[Xlk] = 1/m.

Now define the random variable Y to be the total number of collisions, so that 𝑌 =

The expected number of collisions is

𝐸[𝑌] = 𝐸 [ ] = = () =

**PROBLEM 3:**

**5; 28; 19; 15; 20; 33; 12; 17; 10**

ℎ(𝑘) = 𝑘 mod 9

|  |  |  |  |
| --- | --- | --- | --- |
| **0** |  |  |  |
| **1** | 28 | 19 | 10 |
| **2** | 20 |  |  |
| **3** | 12 |  |  |
| **4** |  |  |  |
| **5** | 5 |  |  |
| **6** | 15 | 33 |  |
| **7** |  |  |  |
| **8** | 17 |  |  |

**PROBLEM 4:**

* Successful searches: no difference,  Θ (1 + *α*).
* Unsuccessful searches: faster but still Θ (1 + *α*).
* Insertions: same as successful searches,  Θ (1 + *α*).
* Deletions: same as before if we use doubly linked lists Θ(1).

**PROBLEM 5:**

𝑚 = 11, 10, 22, 31, 4, 15, 28, 17, 88, 59

ℎ(𝑘) = 𝑘 mod 𝑚. ℎ’(𝑘) = 1 + (𝑘 mod (𝑚 – 1))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Chaining** | **Linear probing** | **Quadratic probing** | **Double hashing** |
| **0** | 22 88 | 22 | 22 | 22 |
| **1** |  | 88 | 88 |  |
| **2** |  |  |  | 59 |
| **3** |  |  |  | 17 |
| **4** | 4 15 59 | 4 | 4 | 4 |
| **5** |  | 15 | 15 | 15 |
| **6** | 28 17 | 28 | 28 | 28 |
| **7** |  | 17 | 17 | 88 |
| **8** |  | 59 | 59 |  |
| **9** | 31 | 31 | 31 | 31 |
| **10** | 10 | 10 | 10 | 10 |