**National University of Computer and Emerging Sciences, Lahore Campus**

| | | | |
|---|---|---|---|
| **Course:** | **Advance Database Systems** | **Course Code:** | **CS451** |
| **Program:** | **BS(Computer Science)** | **Semester:** | **Spring 2018** |
| **Out Date:** | **13-Mar-2018** | **Total Marks:** | **40** |
| **Due Date:** | **Tue 20-Mar-2018** ***(Start of Class)*** | **Weight:** | |
| **Section** | **CS** | **Page(s):** | **1** |
| **Assignment:** | **2** ***(Disk Storage, File Structures and Hashing)*** | | |

***Instructions:*** Use proper assignment papers for solving your assignment questions. Assignment done on diary pages, register pages, rough pages will not be credited. Do not copy the work of your peers. In case cheating is detected, then your case will be referred to DC.

---

**Q1:** ***(20 point)***

Consider a file system on a disk with block size *B=1000 bytes*. A block pointer is *P=6* bytes long, and a record pointer is $P_R$ *=7* bytes long. A file has *r=10,000,000* STUDENT records of fixed-length (un-spanned). Each record has the following fields: *ROLLNO (10 bytes), NAME (25 bytes), DEPTNO (10 bytes), ADDRESS (30 bytes), PHONE (9 bytes), BIRTHDATE (8 bytes), SEX (1 byte), PROGRAMCODE (2 bytes), CGPA (4 bytes, real number)*. An additional byte is used as a deletion marker. No of departments = 200 and students per department = 50,000. RollNo is a primary key column.

Estimate the number of block fetches needed to compute the following queries:

**a)** SELECT * FROM student WHERE rollno=1234; (Assume file is not ordered.)
**b)** SELECT * FROM student WHERE rollno=1234; (Assume file is ordered on RollNo.)
**c)** SELECT * FROM student WHERE deptno=10; (Assume file is not ordered.)
**d)** SELECT * FROM student WHERE deptno=10; (Assume file is ordered on DeptNo.)

**Solution**

R = 10+25+10+30+9+8+1+2+4+1=100bytes
bfr = floor(B/R) = floor(1000/100) = 10
number of blocks required = ceil(r/bfr) = ceil(10,000,000/10) = 1,000,000 blocks

a) = b = 1,000,000
b) =$\log_2(b)$ = $\log_2(1{,}000{,}000)$ = 20
c) = b = 1,000,000
d) b1 = r1/bfr = 50,000/10 = 5000 , $\log_2(b)$=20 , total block fetches = 20 + 5000 - 1 = 5019 blocks

**Q2: Hashing** ***(20 points)***

An employee file has following employee id values:

2831 , 8963 , 1313 , 4981 , 2245 , 3539 , 5182 , 4484 , 5802 , 8186 , 9234 , 3126 , 2128 , 6831 , 7433 , 6129 , 7984 , 1234 , 9999 , 2111 , 3538 , 4199 , 2223

a) Consider these employee ids as hash key values. The file uses 6 buckets named 0 to 5. One bucket cannot hold more than 3 records, means at max a bucket can hold 3 records. Load these records in file using hash function h(k) = k mod 6, in the given order.
b) Calculate the average number of block accesses for random retrieval on employee id
c) Load the given values in expandable hash files based on extendible hashing, show structure on each step, use hash function h(k) = K mod 16, max 3 records can be kept in one bucket.
d) Load the given values in expandable hash files based on dynamic hashing, show structure on each step, use hash function h(k) = K mod 16, max 3 records can be kept in one bucket.

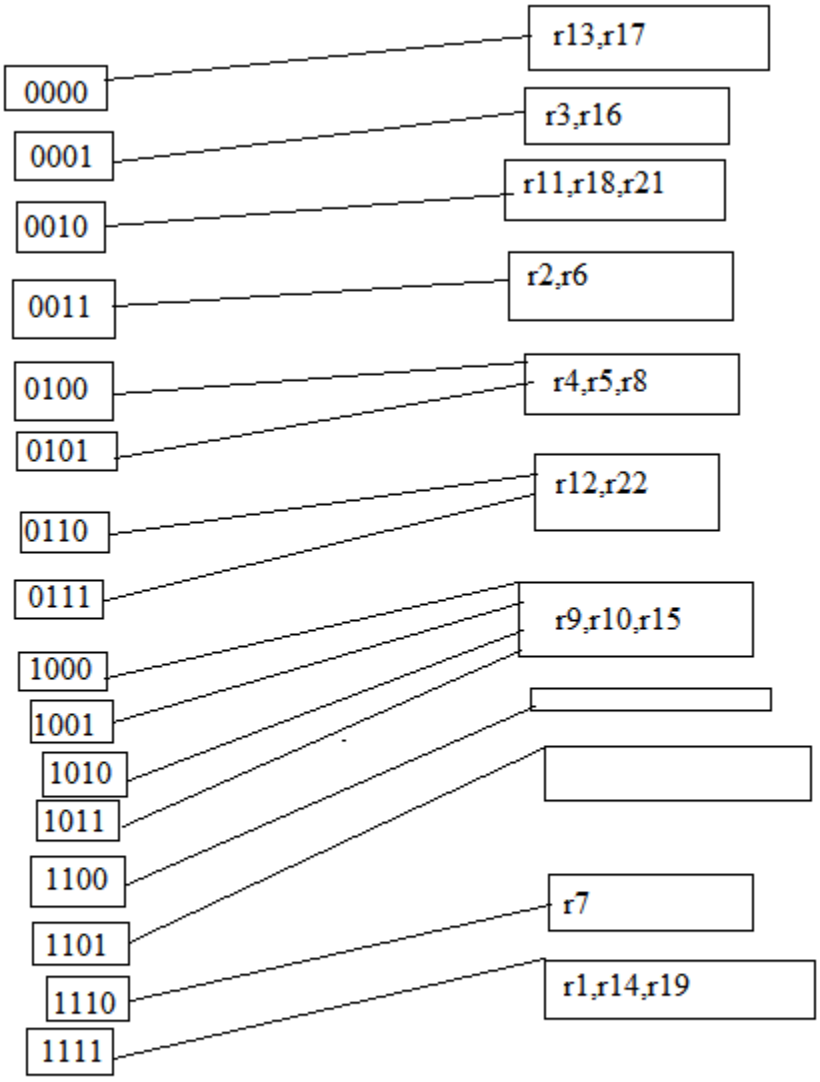
**Solution**

a) mod is calculated like, 5802 mod 6 =0 so bucket zero, and so on,

| | |
|---|---|
| bucket 0 | 5802<br>9234<br>3126 |
| bucket 1 | 4981<br>2245 |
| bucket 2 | 4484<br>8186 |
| bucket 3 | 6831<br>6129<br>9999 |
| bucket 4 | 5182<br>2128<br>7984 |
| bucket 5 | 2831<br>8963<br>1313 |

| | |
|---|---|
| _ | 2223 |
| _ | 1234<br>3538 |
| _ | 3539<br>7433<br>2111 |
| _ | 7433<br>2111<br>4199 |

b) [(1*(16/23))+(2*(6/23))+(3*(1/23))] = 1.34
c) final after inserting all records:




for the insertion of last 2 records in 1111, we would have to modify the hash function to k mod 32 and split accordingly.

d) Do yourself