

## Home Work 9

---

- 10.2

---

**a.** It is stored as an array containing physical page numbers, indexed by logical page numbers. This representation gives an overhead equal to the size of the page address for each page.

**b.** It takes 32 bits for every page or every 4096 bytes of storage. Hence, it takes 64 megabytes for the 64 gigabyte of flash storage.

**c.** If the mapping is such that, every  $p$  consecutive logical page numbers are mapped to  $p$  consecutive physical pages, we can store the mapping of the first page for every  $p$  pages. This reduces the in memory structure by a factor of  $p$ . Further, if  $p$  is an exponent of 2, we can avoid some of the least significant digits of the addresses stored.

- 10.8

---

Hash table is the common option for large database buffers. The hash function helps in locating the appropriate bucket, on which linear search is performed.

- 10.14

---

**a.** It does not matter on what we store in the offset and length fields since we are using a null bitmap to identify null entries. But it would make sense to set the offset and length to zero to avoid having arbitrary values.

**b.** We should be able to locate the null bitmap and the offset and length values of non-null attributes using the null bitmap. This can be done by storing the null bitmap at the beginning and then for non-null attributes, store the value (for fixed size attributes), or offset and length values (for variable sized attributes) in the same order as in the bitmap, followed by the values for non-null variable sized attributes. This representation is space efficient but needs extra work to retrieve the attributes.

- 10.17

---

**a.** Advantages of storing a relation as a file include using the file system provided by the OS, thus simplifying the DBMS, but incurs the disadvantage of restricting the ability of the DBMS to increase performance by using more sophisticated storage structures.

**b.** By using one file for the entire database, these complex structures can be implemented through the DBMS, but this increases the size and complexity of the DBMS.