CSE 350 File Systems Project III

In this part of the project, you are going to create a database system with a single table which uses the file system from Project II. The input file will consist of records associated with Art History. The data file you will use as input consists of records with the following format:

- **Date : 5 bytes**
- **End : 5 bytes**
- **Type : 8 bytes**
- **Place : 15 bytes**
- **Reference : 7 bytes**
- **Description : variable**

In the data file, an asterisk is also used to delimit each field and the last character of each record is an asterisk. The width of any record is never greater than 120 bytes. Therefore you can block the data accordingly. This part of the project will require you to implement the following class:

```
Class Table : Public FileSys
{
Public :
Table(string filename, string flatfile, string indexfile);
int Build_Table(string input_file);
int Search(string value);
Private :
string flatfile;
string indexfile;
FileSys filesystem;
int IndexSearch(string value);
};
```

The member functions are specified as follows :

- ## Table(filename, flatfile, indexfile)

  This constructor creates the table object. It creates the new (empty) files *flatfile* and *indexfile* in the file system on the Sdisk using *filename*.

- ## Build_Table(input_file)

  This module will read records from the input file (the raw data file described above), add the records to the flatfile and create index records consisting of the date and block number, and then add the index records to the index file. (Note that index records will have 10 bytes .. 5 bytes for the date and 5 bytes for the block number.)

- ## Search(value)

  This module accepts a key value, and searches the index file with a call to *IndexSearch* for the record where the date matches the specified value. *IndexSearch* returns the blocknumber of the block in the flat file where the target record is located. This block should then be read and the record displayed.

- **IndexSearch(value)**

  This module accepts a key value, and searches the index file *indexfile* for the record where the date matches the specified value. IndexSearch then returns the block number key of the index record where the match occurs.

The project will also require you to design and implement the machine-user interface. Your system must be capable of answering queries of the form

What record has date=50?

You can implement this function in any way you choose, but it should NOT be a member function of the *Table* class. Instead, it should use the Search function.

IMPLEMENTATION GUIDELINES : It is essential that the procedures satisfy the specifications.

1. NOTE THAT YOU MUST AGAIN WRITE PROCEDURES (DRIVERS) WHICH ALLOW YOU TO TEST YOUR PROGRAM.
2. In Project IV, you will add functions to the Table class that increase the performance of the *Search* function.