Name – Durwankur Naik

Roll No – I4214

Div – 2

Sub – Information Storage & Retrieval

Practical no - 3

```
#include<iostream>
#include<vector>
#include<map>
#include<string>
#include<fstream>
#include<sstream>
using namespace std;
struct word_position
{
 string file_name;
 int line;
 int index;
};
class InvertedIndex
{
 map<string,vector<word_position> > Dictionary;
 vector<string> filelist;
 public:
 void addfile(string filename);
 void show_files();
 void search(string word);
};
void InvertedIndex::addfile(string filename)
{
 ifstream fp;
 fp.open(filename + ".txt",ios::in);
 if(!fp)
 {
 cout<<"File Not Found\n";
 return ;
 }
 filelist.push_back(filename);
 string line,word;
 int line_number=0,word_number=0;
 while(getline(fp,line))
 {
 line_number++;
```

```
 word_number = 0;
 stringstream s(line);
 while(s>>word)
 {
 word_number++;
 word_position obj;
 obj.file_name = filename;
 obj.line = line_number;
 obj.index = word_number;
 Dictionary[word].push_back(obj);
 }
 }
 fp.close();
}
void InvertedIndex::show_files()
{
 int size = (int)filelist.size();
 for(int i=0;i<size;i++) cout<<i+1<<": "<<filelist[i]<<endl;
 if(!size) cout<<"No files added\n";
}
void InvertedIndex::search(string word)
{
 if(Dictionary.find(word)==Dictionary.end())
 {
 cout<<"No instance exist\n";
 return ;
 }
 int size = (int)Dictionary[word].size();
 for(int counter = 0;counter < size ;counter++)
 {
 cout<<counter+1<<":\n";
 cout<<" Filename: "<<Dictionary[word][counter].file_name<<endl;
 cout<<" Line Number: "<<Dictionary[word][counter].line<<endl;
 cout<<" Index: "<<Dictionary[word][counter].index<<endl;
 }
}
int main(int argc, char*argv[])
{
 InvertedIndex Data;
 for(int i = 1 ; i< argc ; i++)
 {
 Data.addfile(argv[i]);
 }
 int choice = 0;
 do
 {
 cout<<"1: See files\n2: Add File\n3: Query Word\n4: Exit\n";
 cin>>choice;
```

```
switch(choice)
{
case 1: Data.show_files(); break;
case 2:
{
cout<<"Enter File Name: ";
string name;
cin>>name;
Data.addfile(name);
break;
}
case 3:
{
cout<<"Enter Word: ";
string word;
cin>>word;
Data.search(word);
break;
}
case 4: break;
default : continue;
}
}while(choice!=4);
return 0;
```

Output –

1: See files 2: Add File 3: Query Word 4: Exit

1

No files added 1: See files 2: Add File 3: Query Word 4: Exit

2

Enter File Name:

ABC.txt

File Not Found 1: See files 2: Add File 3: Query Word 4: Exit

3

Enter Word:

ABC

No instance exist 1: See files 2: Add File 3: Query Word 4: Exit

4

** Process exited - Return Code: 0 **