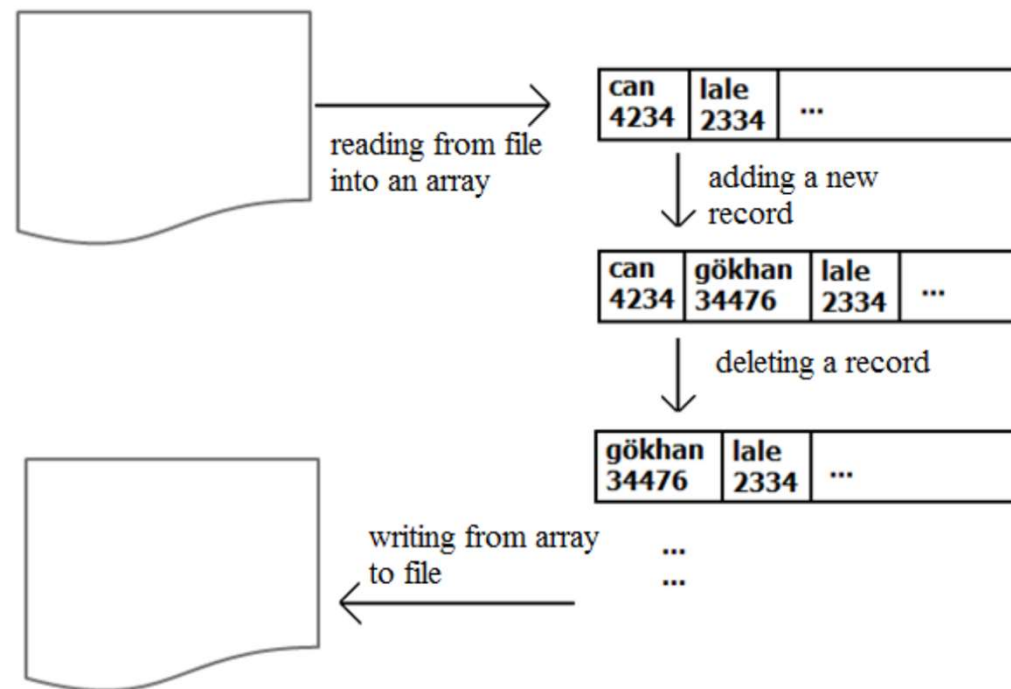


BLG233E Data Structures

Recitation 2

Implementation of Phonebook Application on an Array

- At the start of the program, all the records are read from file into an array.
- All the changes on records are done on the array in the program.
- At the end of the program, all the records are written back to file from array.



Add Array to File structure (fileoperations.h)

```
//Define maximum array size as MAX_RECORDS
#define MAX_RECORDS 100

struct File{
    // BEGIN: No Change
    char filename[20];
    FILE *phonebook;
    void create();
    void close();
    void add(PHONE_RECORD *);
    int search(char []);
    void remove(int recordnum);
    void update(int recordnum, PHONE_RECORD *);
    // END: No Change

    // records: An Array which holds phonebook records
    PHONE_RECORD records[MAX_RECORDS];
    int num_records; // number of records in the array

    // create_array(): writes records from the file to the array
    void create_array();

    // write_to_file(): writes records from the array to the file
    void write_to_file();
};
```

Add create_array() Function (fileoperations.cpp)

```
void File::create(){
    // BEGIN: No Change
    strcpy(filename, "phonebook.txt");
    phonebook = fopen( filename, "r+" );
    if(!phonebook){ // File couldn't be found, create it as an empty file
        if(!(phonebook = fopen( filename, "w+" ))){
            cerr << "Cannot open file" << endl;
            exit(1);
        }
    } // END: No Change

    // read the file and fill in the records array
    create_array();
}

// ADD create_array() function
void File::create_array(){
    num_records = 0;

    fseek(phonebook, 0, SEEK_SET); // Goto start of the file
    while( !feof(phonebook) ){
        fread( &records[num_records], sizeof (Phone_Record), 1, phonebook);
        if( feof(phonebook) ) break;
        if( strcmp(records[num_records].name, "") ) num_records++;
    }
}
```

Rewrite add() and search() Functions (fileoperations.cpp)

```
void File::add(Phone_Record *new_record_ptr){
    strcpy(records[num_records].name, new_record_ptr->name);
    strcpy(records[num_records].phonenum, new_record_ptr->phonenum);
    num_records++;
}

int File::search(char *target){
    int current_index = 0;
    bool all = false;
    int num_found = 0;

    if( strcmp(target, "*") == 0 ) all = true;

    for( ; current_index < num_records; current_index++ ){

        if(!all && strnicmp(records[current_index].name, target, strlen(target))!=0 )
            continue;    // Continue the loop, look at the next record (if any)

        // A match is found, list it and increment number of found records
        cout << current_index << "." << records[current_index].name << " ";
        cout << records[current_index].phonenum << endl;
        num_found++;
    }

    return num_found;
}
```

Rewrite update and remove Functions (fileoperations.cpp)

```
void File::update(int record_num, Phone_Record *new_record_ptr){
    strcpy(records[record_num].name, new_record_ptr->name);
    strcpy(records[record_num].phonenum, new_record_ptr->phonenum);
}

void File::remove(int record_num){
    for(num_records--; record_num < num_records; record_num++)
        records[record_num] = records[record_num+1];
}
```


ADD «Write» to File Menu Option (phoneprog.cpp)

```
void print_menu(){
    system("cls");
    cout << endl << endl;
    cout << "PHONE BOOK APPLICATION" << endl;
    cout << "Choose an operation" << endl;
    cout << "S: Record Search" << endl;
    cout << "A: Record Add" << endl;
    cout << "U: Record Update" << endl;
    cout << "D: Record Delete" << endl;
    cout << "W: Write to File" << endl; // ADD This Menu Option
    cout << "E: Exit" << endl;
    cout << endl;
    cout << "Enter a choice {S,A,U,D,W,E}: "; // ADD W Option to choices
}
```

```
bool perform_operation(char choice){
    bool terminate=false;
    switch (choice) {
        // BEGIN: ADD write_to_file() case
        case 'W': case 'w':
            book.write_to_file();
            break;
        // END: ADD write_to_file() case
        case 'S': case 's':
            search_record();
            break;
        case 'A': case 'a':
            add_record();
            break;
        case 'U': case 'u':
            update_record();
            break;
    }
```

ADD write_to_file() Function (fileoperations.cpp)

```
void File::write_to_file(){
    int current_index = 0;

    if(!(phonebook = fopen( filename, "w+" ))){
        cerr << "Cannot open file" << endl;
        exit(1);
    }

    for(; current_index < num_records; current_index++){
        fwrite( &records[current_index], sizeof (Phone_Record), 1, phonebook);
    }
};
```


THANKS FOR LISTENING