

**Lab 12 (30 Points)****Objectives:**

- Learn how to implement generic binary search.

**Description:**

The follow is the struct you need for this lab:

```
typedef struct {
    int id;
    int age;
    int department;
    float salary;
    char name[20];
} Employee;
```

Implement the following functions:

`Employee **readFile(void)`

- This function reads data from the employeeInfo.csv file, creates an array of Employee pointers, and point to Employee structs that containing their information. This returns the array of Employee pointers.

`void *binarySearch(void *empPtrs[], int f, int l, void *query)`

- This function search for an Employee ID where the return value is a pointer to the object or NULL if the object isn't in the array.

`int compare(void * obj1, void * obj2)`

- This function compares obj1's id with obj2's id where the return value is negative if obj1's id is less than obj2's id, zero if they're equal, and positive if obj1's id is greater than obj2's id.

`void freeArray(void * employeePtrs[])`

- This function frees all dynamically-allocated memories.

Implement the main function as following:

- Call your readFile function.
- Display a prompt to ask a user to enter in an employee id.
- Search for the employee info based on the entered employee id, using binarySearch function.
- Display employee info if he/she is an employee of the company.
- Free your dynamically-allocated memories, using freeArray function.

**Every user-defined function must have a comment describing:**

- What function does;
- What parameter values are;
- What value it returns.

**Example from the terminal window:**

```
$ ./a.out
Please enter an employee ID to search the personal info: 12
Employee ID : 12
Name       : MARJORIE
Age        : 40
Department : 806
Salary     : $88801.60

$ ./a.out
Please enter an employee ID to search the personal info: 1000
Employee ID : 1000
Name       : MARGARET
Age        : 33
Department : 808
Salary     : $79536.31

$ ./a.out
Please enter an employee ID to search the personal info: 1100
Employee ID 1100 doesn't exist!!
```

**Grading Criteria:**

- Main program: 6 points
- readFile function: 6 points
- binarySearch function: 6 points
- compare function: 6 points
- freeArray function: 6 points

**Note:**

- If your code does not compile with **-Wall** and **-Werror**, you will receive a **zero** for this assignment.
- You need to finish at least **three** peer reviews within three days of this lab. Otherwise, you will get a 20% penalty.
- You will lose points if you don't have enough comments.