

Consider the following header file **emp.h**

```
#if ! defined EMP
#define EMP

/* struct Employee has the following members:
   - name as string of maximum length of 24 letters
   - id as integer
   - salary as floating point number */
typedef struct Employee Employee;

/* function read_employee reads employee's information from
   the Keyboard and fills it in the parameter e */
void read_employee(Employee* e);

/* function print_employee prints the information of
   the employee e to the screen */
void print_employee(Employee e);

/* function copy_employee takes an Employee parameter e and
   returns a pointer to new copy of e */
Employee* copy_employee(Employee e);

/* function emp_gte_5000 takes an array of employees earr of size
   s and returns a new array contains copies of employees whose
   salaries are greater than or equal to 5000, of size res_s */
Employee* emp_gte_5000(Employee* earr, int s, int* res_s);

/* function save_to_file stores the size of array earr in the file
   fn, then stores all employees of earr in the file and returns
   1 if succeeded, and 0 otherwise */
int save_to_file(char* fn, Employee* earr, int size);

#endif
```

1. Write a C program "**emp.c**" that:
 - a. Defines the type `struct Employee`. 1 point
 - b. Implements the function `read_employee` 1.5 point
 - c. Implements the function `print_employee` 1 point
 - d. Implements the function `copy_employee` 1.5 point
 - e. Implements the function `emp_gte_5000` 1.5 point
 - f. Implements the function `save_to_file` 1.5 point
2. Bonus: write a main function that:
 - a. Asks the user to enter number of employees `num` and allocates enough memory for array `emp_arr` to store `num` employees
 - b. Reads the information of the `num` employees from the keyboard
 - c. Stores employees who receive a salary ≥ 5000 in array `emp_5000` of size `num_5000`
 - d. Prints the information of all employees in `emp_5000` to the screen
 - e. Stores the number of employees in `emp_5000` and their information in the file "`emp-5000.dat`"

Sample run:

```
Enter number of employees: 4
Enter the name: Ahmad
Enter the id: 22
Enter the salary: 5700
Enter the name: Ali
Enter the id: 17
Enter the salary: 4200
Enter the name: Omar
Enter the id: 25
Enter the salary: 6200
Enter the name: Fahad
Enter the id: 29
Enter the salary: 3900

Employees receiving more than 5000 as salary:
=====
Name:Ahmad
Id:22
Salary:5700.00
-----
Name:Omar
Id:25
Salary:6200.00
-----
```