CS 2123-001 Data Structures

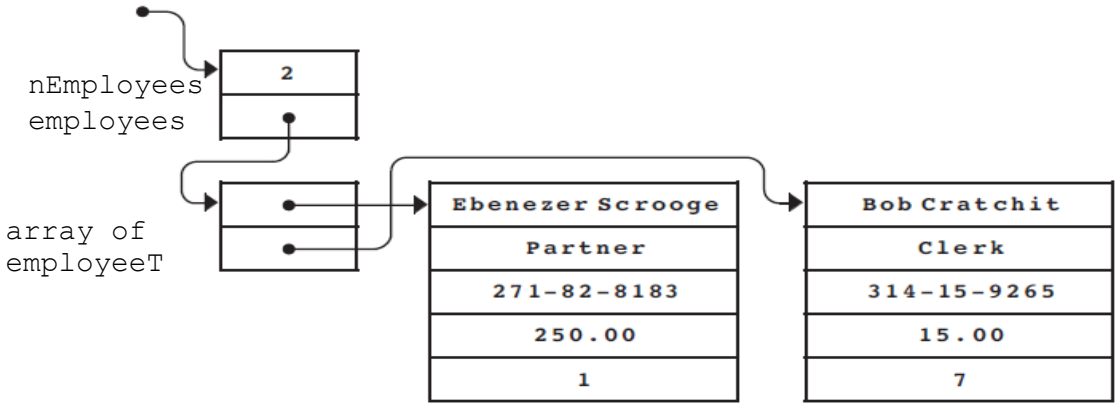
Instructor Dr. Turgay Korkmaz

Homework 2
Due date: check BB
!!!! NO LATE HOMEWORK WILL BE ACCEPTED !!!

(From the textbook ch-2)

Programming Exercise 12 “Design a new type called payroll…” and Exercise 13 “Write a Program that generates the weekly payroll…” In case you don't have the textbook, here are the problems:

12. Design a new type called `payrollT` that is capable of holding the data for a list of employees, each of which is represented using the `employeeT` type introduced in the section on “Dynamic records” at the end of the chapter. The type `payrollT` should be a pointer type whose underlying value is a record containing the number of employees and a dynamic array of the actual `employeeT` values, as illustrated by the following data diagram:



After writing the types that define this data structure, write a function `GetPayroll` that reads in a list of employees, as shown in the following sample run:

```
How many employees: 2
Employee #1:
  Name: Ebenezer Scrooge
  Title: Partner
  SSNum: 271-82-8183
  Salary: 250.00
  Withholding exemptions: 1
Employee #2:
  Name: Bob Cratchit
  Title: Clerk
  SSNum: 314-15-9265
  Salary: 15.00
  Withholding exemptions: 7
```

After the input values have been entered, the **`GetPayroll`** function should return a value of type **`payrollT`** that matches the structure shown in the diagram.

13. Write a program that generates the weekly payroll for a company whose employment records are stored using the type **`payrollT`**, as defined in the preceding exercise. Each employee is paid the salary given in the employee record, after deducting taxes. Your program should compute taxes as follows:

• Deduct $1 from the salary for each withholding exemption. This figure is the *adjusted income.* (If the result of the calculation is less than 0, use 0 as the adjusted income.)

• Multiply the adjusted income by the *tax rate,* which you should assume is a flat 25 percent.

For example, Bob Cratchit has a weekly income of $15. Because he has seven dependents, his adjusted income is $15 – (7 x $1), or $8. Twenty-five percent of $8 is $2, so Mr. Cratchit's net pay is $15 – $2, or $13.

The payroll listing should consist of a series of lines, one per employee, each of which contains the employee's name, gross pay, tax, and net pay. The output should be formatted in columns, as shown in the following sample run:

```
Name                      Gross    Tax      Net
-----------------------------------------------
Ebenezer Scrooge       250.00 - 62.25 = 187.75
Bob Cratchit            15.00 -  2.00 =  13.00
```

***Hint**: First download the program provided by the textbook and Makefile that I put under hw-02 directory or attached here. Compile and execute it. Once you understand what it does, then implement the necessary functions, data structures etc. to do the things in the above questions. In this program, you will find some already implemented code which can help to complete this hw. (Also, I changed all the textbook specific memory allocation things with malloc in that program. So you can simply use gcc to compile it, but you need to check if memory is allocated or not after each malloc. Please make these corrections, too!).*

It is not mentioned in the questions, but from now on, you should make sure you will release (free) the dynamically allocated memories if you allocate any in your programs.… So, before submitting your program, run it with `valgrind` to see if there is any memory leakage… Also if you need to debug your program, compile your programs with –g option and then run it with `gdb` and/or `ddd`.

A NOTE FROM PREVIOUS SEMSTERS:

There is some confusion about Assignment 2 and I wanted to ensure that everyone is on the same page. Many of you were under the impression that writing GetPayroll was unnecessary because the function InitEmployeeTable already initialized an employee table of two

employees. You still need to write GetPayroll for a user to enter an arbitrary number of employees to a payroll. InitEmployeeTable fills out an example table for demonstration purposes.
The output figure at the beginning of page #2 of the assignment shows how this should work. Output is displayed in black and values in blue are entered via the keyboard. You should use the number entered on the first line asking "how many employees" to loop through the input fields of an employee. Remember that there is a limit based on the MaxEmployees macro (100 by default). Use these values to control the loop by which you obtain employee information. This function should return a pointer to a payroll upon termination.

What to return: !!!! NO LATE HOMEWORK WILL BE ACCEPTED !!!

1. Follow the problem solving methodology, and solve the problem(s). Then convert your solution(s) to a C program. Include /* comments */ so that we can understand your solution(s).
2. You are already given some code under hw02.zip. Unzip it, use make to compile employee.c, and then run it as is. Study the structures and code in employee.c. Then you can modify employee.c as needed.
3. Again use make to compile it, then run it. Copy/paste the result in an output file, say out2.txt. Or simply use script thing as in hw1….
4. Finally re-zip hw02 folder and submit it through BB learn.

/* Don't forget to include comments about the problem, yourself and each major step in your program! */

You must submit your work using Blackboard Learn and respect the following rules:

1) All assignments must be submitted as a .tar archive file unless it is a single pdf file.
2) Assignments must include all source code.
3) Assignments must include an output.txt file which demonstrates the final test output run by the student.
4) If your assignment does not run/compile, the output.txt file should include an explanation of what was accomplished, what the error message was that prevented the student from finishing the assignment and what the student BELIEVES to be the underlying cause of the error.
