**Assignment 2**

THIS IS AN INDIVIDUAL ASSIGNMENT. Your submitted work must be "ENTIRELY" your work. Make sure you understand the academic policies and procedures as detailed on the college website at: http://www.georgianc.on.ca/admissions/policies-procedures/academic-misconduct

Pay special attention to warnings about cheating, plagiarism, and acceptable use of college computer resources.

If your submitted work is found out to have been re-produced from any human or non-human resource, for example, the web, I will strictly be following the procedure as in the above document.

--------------------------------------------------------------------------------------------------------------------------------------------

Write a program to produce a bonus report. See the problem definition in the figure below.

1. The payroll records are sorted into ascending sequence by office number within territory number in the input file PAYROLL-MASTER. Thus, all employees within office 01 within territory 01 will appear before employee records for office 02 within territory 01, and so on.
2. Only employees who were hired before 1994 are entitled to a 10% bonus.
3. Print the names of all employees and their bonuses. Print a separate page for each office within each territory. Hence a change in either the office number or territory number causes printing to start from a new page. USE THE CONTROL BREAK PROCESSING DEMONSTRATED IN CLASS TO ACHIEVE THIS.
4. The input data file CH0901.dat is provided. Please study the Record Layout provided as part of the assignment thoroughly to understand the data needed from the input file.
5. Please use the name CH0901.dat in the ASSIGN clause for the input file.
6. A text file CH0901.rpt showing the expected output has also been provided.
7. Please write an efficient program and do not write unnecessary and redundant code. The program must be generic in nature. Do not use ‘magic numbers’ in the code.

**Points to note in the output provided:**

1. Every page shows the page number and current date in the format mm/dd/yyyy. Since SPICE does not support the FUNCTION CURRENT DATE, accept the current date from the user.

Display a message, asking today's date from the user in the format yyyymmdd, accept it and display it in the required format mm/dd/yyyy.

This has been demonstrated in class.

I ran the program on July 16, 2013 so that 'current date' shows on each page in the output provided.

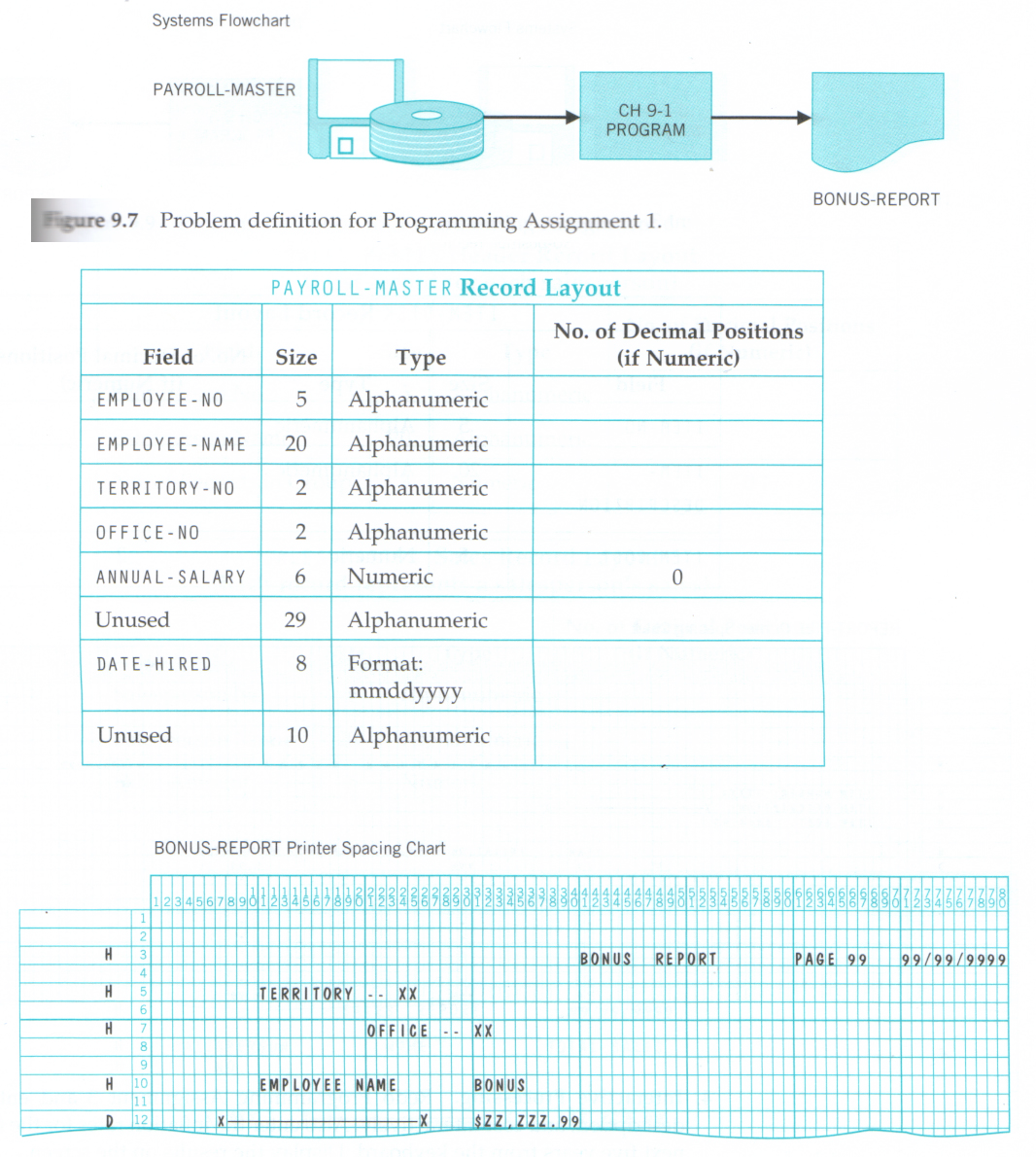
1. Every page shows the employee records in a specific territory and office.

Use appropriate ADVANCING PAGE clause with the write statement to start from a new page as required. Use CONTROL BREAK PROCESSING to achieve page breaks for each office within each territory.

1. Please pay special attention to the spacing between heading lines and detail lines. Refer to the printer spacing chart and the output file provided.

Use appropriate ADVANCING LINES clause with the write statement to achieve this.

1. Pay attention to column spacing.
2. Notice that when an employee has no bonus, the bonus field is blank. Use appropriate editing clause with the picture given to achieve this. BLANK WHEN ZERO



**How to submit**: Use the Assignment 2 link on blackboard to submit your COBOL code file ONLY and name it **Assignment2 *yourname*.cbl**. Two submissions are allowed just in case you want to make any last minute adjustments.

**Marking Scheme**

|  |  |
| --- | --- |
| Requirement | Marks allotted |
| Program executes with no errors | 4 |
| Program compiles clean with no syntax errors | 4 |
| Input file specifications followed | 4 |
| Output file specifications followed | 4 |
| Output as required with no logic errors | 8 |
|  |  |
|  |  |
| Programmer-defined names are well chosen | 3 |
| Program is modular with well-defined paragraphs | 3 |
| Non-recursive PERFORM’s | 1 |
| Efficient, relevant, non-redundant and generic code in the Procedure Division to achieve the desired output | 9 |
| No ‘magic numbers’ in the Procedure Division code | 1 |
| Page break occurs whenever a territory or office changes | 2 |
| Date is input from the user in yyyymmdd format | 1 |
|  |  |
| Code is commented-name, date and purpose at head; each paragraph has an accompanying comment stating its purpose; the ‘main’ logic is explained/commented fully | 5 |
| Code is easily readable and is indented properly | 2 |
| PICs, VALUEs etc. line up | 2 |
| Periods are present where required in Procedure Division; no extra periods should be present else marks will be deducted | 2 |
| Area A and Area B rules have been followed | 2 |
| Files are closed | 1 |
| Record contains clause is present in all file descriptions and is correct | 2 |
| Total | 60 |