Ollscoil Chathair
Bhaile Átha Cliath
Dublin City University

TERMINAL ASSIGNMENT– SEMESTER II - 2019/2020

MODULE: ***Software Development for Engineers (EM108)***

PROGRAMME(S):

CE B.Eng. in Common Entry into Engineering
BMED B.Eng. in Biomedical Engineering
ME B.Eng. in Mechatronic Engineering
CAM B.Eng. Mechanical & Manufacturing Eng
ECE BEng Electronic & Computer Engineering

YEAR: ***1 (one)***

EXAMINER: ***Associate Professor Gabriel-Miro Muntean (ext. 7648)***

TIME ALLOWED: **48 hours**

INSTRUCTIONS: ***Answer THE QUESTION.***

This booklet contains 5 pages, including the cover sheet.

PLEASE DO NOT TURN OVER THIS PAGE UNTIL YOU ARE INSTRUCTED TO DO SO

Requirements for this paper (Please mark (X) as appropriate)

	Log Tables		*Thermodynamic Tables*
	Graph Paper		*Actuarial Tables*
	Dictionaries		*MCQ Only – Do not publish*
	Statistical Tables		*Attached Answer Sheet*
	Bible		*Exam Paper to be returned with Booklet*

Dublin City University
School of Electronic Engineering
School of Mechanical and Manufacturing Engineering

Directions

The terminal assignment consists of one problem. You need to design a solution, write the code using the C programming language, test your program, write a report and submit your work in loop.

In order to compile and test your program, either a locally-installed compiler such as BorlandC or one of the following online compilers can be used:
https://www.onlinegdb.com/online_c_compiler
https://www.tutorialspoint.com/compile_c_online.php
https://repl.it/languages/c

A report template is available at the end of this document and can also be downloaded from the EM108 loop site.

The input files can be created using the data at the end of this document or can be downloaded from loop.

You should submit the report as a single file (PDF or DOC/DOCX) via loop in the *Terminal Assignment* entry before **April 27th 2020 at 18:00**. The code should be included at the end of the report file and not submitted separately.

Problem [Total Marks: 100]

A university keeps records of students and companies engaged in the INTRA programme using a computer program. This program uses two files (named ***students.txt*** and ***companies.txt***) which store details on the students and companies, respectively. The files use lines with the following format for storing information.

students.txt
Name Surname PPS Nationality Married Children Registration

Each line contains 7 pieces of information about a student, separated by space. The first two pieces of information represent the student **name** and **surname**, the third the associated **PPS** as an alphanumeric code, the fourth the student **nationality** as an alphabetic field, the fifth is an indication whether the individual is **married** or not (e.g. 1 or 0), the sixth is the number of **children** the student has and the final field is an alphanumeric entry which contains the **registration code** of the company the student performs his/her INTRA with.

companies.txt
Company Registration Salary Benefits

Each line contains 4 pieces of information about companies, separated by space. The first is the **company name**, the second is the company unique alphanumeric **registration code**, the third is the **salary**, expressed in euro and finally **benefits** indicates whether the company offers or not benefits (1 or 0).

You are required to develop a program which will perform the following steps:

1. Write a function to read the data from the two input files into two arrays of structures *(5 marks)*. Assume a maximum of 20 characters for each alphabetic name or alphanumeric code. Note there are no spaces introduced between the potential multiple words of any name. Once read, the program should not allow data to be read again from the files and a warning message will be displayed instead *(5 marks)*. It is assumed that there are no more than 1000 entries in each file. **(10 marks)**

2. Ask the user to input a company name. Write a function to search for the company in the company array of structure and return its registration if found or -1 otherwise *(5 marks)*. Ask the user for a student name. Write a function to search for the student and - if found - record the fact that the student will do his/her INTRA with the inputted company by storing company's registration in student's record. If the student is not found, print a message *(10 marks)*. **(15 marks)**

3. Ask for a student PPS. Search for the student in the data structure and - if found - print his/her details: name, nationality, married status and number of children. Print a message if the student is not found *(5 marks)*. Print also the INTRA-related details as follows: company name, registration code, salary *(10 marks)*. You may find helpful if you write a function that receives as a parameter a company registration code, searches for the company in the company array of structure and returns its array index, if found or -1 otherwise.

(15 marks)

4. For each student, compute and store in the student array of structure their income according to their INTRA company details. In the student structure, add a new integer entry named **income** *(5 marks)*. In the student array of structure, for all the students, fill this entry as follows. If the company does not offer benefits, the income is equal to the company advertised salary *(10 marks)*. If the company does offer benefits, an additional allowance of 5% for married students and a child support of 3% for each of student's children needs to be added. The international students get a fixed travel allowance of 200 euro *(10 marks)*.
(25 marks)

5. Ask the user to input a letter. Print on the screen the list of students whose names start with that letter and for each student data in the following format:

	| Name	**| Surname**	**| PPS**	**| Company**	**| Income**	**|**
e.g.	**| John**	**| OBrien**	**| 1234567D**	**| Vodafone**	**| 32500**	**|**

(10 marks)

6. Read a file name from the keyboard, and write in a file with that name *(5 marks)* student information in the following format: *(5 marks)*:

Name Surname Nationality Company Registration Income

e.g. **John OBrien Irish Vodafone X4321674 32500**

(10 marks)

The program should include at least three functions, demonstrate good coding practices with regard to spacing, indentation, commenting, etc. and offer the user a menu and repeated access to the program options. **(5 marks)**

Make sure the program compiles and executes. Test the program rigorously. Record, report and comment all test results. **(10 marks).**

File Listing: companies.txt

Vodafone X432167Y 37500 1
Smurfit F123456T 40000 1
Ryanair A734167H 42000 0
Primark B224267N 45000 0
KerryGroup 8234567B 47000 1
RoadStone 4234567F 50000 0
Musgrave P894527M 55000 1
HP 7734567E 60000 1
IBM S438137P 75000 1
AIB M524993R 75000 0

File Listing: students.txt

Jimmy OBrien **1234567D Irish 1 2 X432167Y**
David Keogh **3234567F Irish 0 0 F123456T**
David Hanahoe **5234567G Irish 1 2 X432167Y**
Tom Byrne **6234567N Irish 1 4 A734167H**
Thomas Collins **8234567B Irish 0 0 S438137P**
Aoife Murphy **4234567F Irish 1 3 P894527M**
Joan OBrien **5234567M Irish 1 1 S438137P**
Maria Ionescu **7734567E International 1 2 A734167H**
Celine Keogh **2254893R Irish 1 0 A734167H**
Eva Swan **5234997P International 1 1 M524993R**

Terminal Assignment Report Template

EM108: SOFTWARE DEVELOPMENT
ACADEMIC YEAR 2019-2020
REPORT

Student ID:
Class:
Date:

Plan

Development

Test

Conclusions

Code attached