

|  |
| --- |
| **COMP 2130**  **Assignment #1 – Household Survey Results, no functions** |

## Due Date: Beginning of your seminar on January 22, 2020

**NOTE: Add these comments to the top of each file you create:**

/\*\*\*\*

\* Name: XXXX XXXXXXXX

\* Student Number: XXXXXXX

\* Assignment Number: X

\* Due Date: eg January 22, 2020

\* Program Description: include a **detailed description** of what this program does

\*\*\*\*/

**Student Academic Integrity**

All assessments given for the computing science courses are governed by the academic honesty policy of both TRU and the Computing Science Department. Academic dishonesty is a serious offense. All work submitted must be of your own. Preliminary discussions may be held with other students but the work produced finally must be your own. Jointly produced work will result in a zero for that assignment for all students involved in the joint work and may mean loss of a letter grade for the course. The student may also be expelled from the University.

**DO NOT SHARE CODE OR ANYTHING IN WRITING WITH YOUR CLASSMATES. DO NOT USE CODE FROM THE INTERNET. YOUR ENTIRE PROGRAM MUST BE WRITTEN BY YOU.**

**PROBLEM**: **Write a C program** that will produce the **EXACT** output shown below.

1. Using initialization lists, **create 3 one-dimensional arrays, one for each of these:**

* hold the names of the households (see data and below, and note at the end)
* hold the incomes of the households (see data below)
* holds the number of members of each household (see data below)

Use the data from the example below.

1. Your C program should take the data in the arrays and produce the output below, **neatly formatted as shown**:

Household Income Members Comments

Armstrong 25295 7

Burns 35178 3 \*\*

Cousins 66711 5 \*

Duggan 10455 1 \*\*\*\*

Evans 28943 3

Field 43226 3

Garnett 30869 4 etc.

Hadfield 21143 5

Johnston 26444 3

Lovett 33777 4

McDonald 17551 1

Petersen 41323 4

Singh 39117 2

\* - Very High Income (income is more than $10,000 above the average)

\*\* - Above Average Income

\*\*\* - Income is below the Poverty Level

\*\*\*\* - Extreme Poverty (more than $5,000 below the poverty level)

The average income of the households surveyed is $XXXXX

The percentage of households below the poverty level is YY%.

The poverty level is calculated using the formula, poverty level = $18,000 + $1,500(members – 2). Note that the poverty level depends on the number of household members (i.e. the more members in the household, the more income needed to house, feed and clothe everyone, and therefore the higher the income needed so that the household is NOT living in poverty)

**Assignment Submission:**

Submit a print-out of your program’s source code, and the output window (change the colors of the window – see below).

Use File > Print to print the file.

Use Alt + Print Screen to capture the output as a picture and paste it into a Word document. (OR you may need to use Fn + Alt + Print Screen)

**Change the text/background colors before you capture the output. You may crop the picture if you wish. (See Using Dev C++.docx for an explanation of how to do this.)**

The assignment will be marked out of 2 using this marking scheme:

* 2 – program is generally written well, is efficient and produces correct output
* 1 – program needs some improvements (see below)
* 0 – program needs a lot of improvements

Well written programs have the following characteristics

* header is included
* a copy of the output is included
* proper indentation is used
* blank spaces and lines included where needed
* good variable names
* the code is efficient
* all elements of the problem are solved as described in the problem statement
* the output is correct
* program works for any data set

**Notes:**

1. **you will be rewriting this program several times so please save it.**
2. **String arrays don’t exist in C, so you must use this notation (we’ll explain why later)**

char \*households[] = { “Armstrong”, “Burns”, etc};

* you can use this arrays like any other array, eg households[1] would be Burns