
HW 3: Functions & Arrays - Income Tax Calculation

Write a program to perform a simple income tax calculation. It should prompt the user for data like income, expense, deductions, etc, and calculate the income tax for a given year.

Do not use STL container variables (vector, deque, etc) or variables of type string. Rather, use integer and character arrays and value returning and/or non-returning functions which are passed value and/or reference parameters. Furthermore, its structure must comply with the following guidelines:

1. Use a header file for assignment and your personal information as usual
2. Use a separate header file, say, `inctax.h`, for declarations of functions (listed below) and include it in the file that contains your `main()` (`hw03.cpp`)
3. Perform **I/O** for tax payer name (`taxPayer[]`) and tax year (`year[]`) in `main()`
4. **Input** and processing (eg, calculate total, keep track of number of entries) for income items should be performed in function `getInc()`; **output** in function `prInc()`
5. Similarly, **input** and processing of expense items should be performed in function `getExp()`; **output** in function `prExp()`
6. Same scheme for dependents: perform **input** (name, relationship) in function `getDep()`; **output** in function `prDep()`
7. Finally, write a function, `prSumm()` to calculate and summarize the tax owed using the following formula:

$$tax = \{[incTot - expTot - (depTot * 3000.00)] * 0.18\} \quad (1)$$

Use the following variables (in `main()`):

```
char taxPayer[40];           // name of tax payer
char year[5];                // year in question (notice type char)
double incItem[10];          // gross income from each source (10 max)
char incDesc[10][40];        // gross income item description (10 max)
int isiz;                    // to hold number of income items entered
double incTot;                // total gross income for the year
double expItem[10];          // expense items (mortgage interest, medical)
```

```
char expDesc[10][40];    // expense item description
int esiz;                // to hold number of expense items entered
double expTot;           // total expense for the year
char depNam[10][40];     // name of each dependent
char depRel[10][40];     // relationship of each dependent
int dsiz;                // number of dependents entered if any
```

Notice that `incItem[]/incDesc[] []`, `expItem[]/expDesc[] []`, and `depNam[] []/depRel[] []` are pairs of parallel arrays. (Use them as such in your functions.) `isiz`, `esiz`, and `dsiz` are the corresponding number of entries.

Declare the following functions, filling in for the formal parameter lists, in the header file `inctax.h` and define them in the function file `inctax.cpp`:

```
_____ getInc (_____, _____, _____);
_____ getExp (_____, _____, _____);
_____ getDep (_____, _____, _____);
_____ prInc (_____, _____, _____);
_____ prExp (_____, _____, _____);
_____ prDep (_____, _____, _____);
_____ prSumm (_____, _____, _____);
```

Run `script` and compile/interact with the program as usual (which implies you now also have `hw03.scr`). Package a `tar` file and submit it to canvas by the due date on top of this page.