4.12 C++ example: Salary calculation with loops

zyDE 4.12.1: Calculate adjusted salary and tax with deductions: Using loops.

A program may execute the same computations repeatedly.

The program below repeatedly asks the user to enter an annual salary, stopping when the user enters 0 or less. For each annual salary, the program determines the tax rate and computes the tax to pay.

1. Run the program below with annual salaries of 40000, 90000, and then 0.
2. Modify the program to use a while loop inside the given while loop. The new inner loop should repeatedly ask the user to enter a salary deduction, stopping when the user enters a 0 or less. The deductions are summed and then subtracted from the annual income, giving an adjusted gross income. The tax rate is then calculated from the adjusted gross income.
3. Run the program with the following input: 40000, 7000, 2000, 0, and 0. Note that the 7000 and 2000 are deductions.

Note: The calculation is inaccurate to how taxes are formally assessed and is a simplification for educational purposes only.

Load default template...

```
#include <iostream>
#include <string>
using namespace std;

int main() {
   const string SALARY_PROMPT = "\nEnter annual salary (0 to exit): ";
   int annualSalary;
   int deduction;
   int totalDeductions;
   double taxRate;
   int taxToPay;

   cout << SALARY_PROMPT;
   cin >> annualSalary;

   while (annualSalary > 0) {
      // FIXME: Add a while loop to gather deductions. Use the variables
      // deduction and totalDeduction for deduction handling.
```

```
40000
90000
0
```

Run

Feedback?

A solution to the above problem follows. The input consists of three sets of annual salaries and deductions.

zyDE 4.12.2: Calculate adjusted salary and tax with deductions: Using loops (solution).

Load default template...

```
#include <iostream>
#include <string>
using namespace std;

int main() {
   const string PROMPT_SALARY = "\nEnter annual salary (0 to exit): ";
   const string PROMPT_DEDUCTION = "Enter a deduction (0 to end deductions): ";
   int annualSalary;
   int oneDeduction;
   int totalDeductions;
   int adjustedSalary;
   double taxRate;
   int taxToPay;

   cout << PROMPT_SALARY << endl;
   cin >> annualSalary;

   while (annualSalary > 0) {
```

```
40000 3000 6000 0
90000 5000 0
60000 2000 1000 1450 0
```

Run

Feedback?

zyDE 4.12.3: Create an annual income and tax table.

A tax table shows three columns: an annual salary, the tax rate, and the tax amount to pay. The program below shows most of the code needed to calculate a tax table.

1. Run the program below and note the results.
2. Alter the program to use a for loop to print a tax table of annual income, tax rate, and tax to pay. Use starting and ending annual salaries of 40000 and 60000, respectively, and a salary increment of 5000.
3. Run the program again and note the results. You should have five rows in the tax table.
4. Alter the program to add user prompts and read the starting and ending annual incomes from user input.
5. Run the program again using 40000 and 60000, respectively, and the same salary increment of 5000. You should have the same results as before.
6. Alter the program to ask the user for the increment to use in addition to the starting and ending annual salaries.
7. Run the program again using an increment of 2500. Are the entries for 40000, 45000, 50000, 55000 and 60000 the same as before?

Note: The calculation is inaccurate to how taxes are formally assessed and is a simplification for educational purposes only.

Load default template...

```
#include <iostream>
using namespace std;

int main() {
   int annualSalary;
   double taxRate;
   int taxToPay;
   int startingAnnualSalary;
   int endingAnnualSalary;

   annualSalary = 0;
   startingAnnualSalary = 0;   // FIXME: Change the starting salary to 40000
   endingAnnualSalary = 0;     // FIXME: Change the ending salary to 60000

   // FIXME: Use a for loop to calculate the tax for each entry in the table.
   // Hint: the initialization clause is annualSalary = startingAnnualSalary

      // Determine the tax rate from the annual salary
```

```
40000 60000 5000
```

Run

Feedback?

A solution to the above problem follows.

zyDE 4.12.4: Create an annual income and tax table (solution).

Load default template...

```
#include <iostream>
using namespace std;

int main() {
   int annualSalary;
   double taxRate;
   int taxToPay;
   int startingAnnualSalary;
   int endingAnnualSalary;
   int incomeIncrement;

   cout << "Enter first annual salary for the table: " << endl;
   cin >> startingAnnualSalary;

   cout << "Enter last annual salary for the table: " << endl;
   cin >> endingAnnualSalary;

   cout << "Enter the increment for the table: " << endl;
```

```
40000 60000 2500
```

Run

Feedback?

How was this section? Provide section feedback