

**UNIVERSITY OF WINDSOR  
COMPUTER SCIENCE 60-140-01/02  
MIDTERM TEST SOLUTION**

Examiners: Dr. C.I. Ezeife

Given: Fri., Oct 30, 2015

Student Name: \_\_\_\_\_

Student Number: \_\_\_\_\_

Lecture Section (circle one): 1, 2

Lab. Section(circle one): 51, 52, 53, 54, 55, 56, 57

---

**INSTRUCTIONS (Please Read Carefully)**

Examination Period is 1 hour 30 minutes. **DO NOT WRITE WITH PENCIL.**

Answer all questions. Write your answers in the spaces provided in the question paper.

May use only simple calculators. **WRITE WITH PEN.**

Total Marks = 100. Total number of questions = 3. Total number of pages = 8.

---

**CONFIDENTIALITY AGREEMENT & STATEMENT OF HONESTY**

**I confirm that I will keep the content of this assignment/examination confidential. I confirm that I have not received any unauthorized assistance in preparing for or doing this assignment/examination. I confirm knowing that a mark of 0 may be assigned for copied work.**

\_\_\_\_\_  
Student Signature

\_\_\_\_\_  
Student Name (please print)

\_\_\_\_\_  
Student I.D. Number

\_\_\_\_\_  
Date

---

For Marking Purposes Only (this part not to be filled by students)

Question	Mark
1 (35 marks)	
2 (15 marks)	
3a (20 marks)	
3b (10 marks)	
3c (20 marks)	
Total Mark (100 marks)	

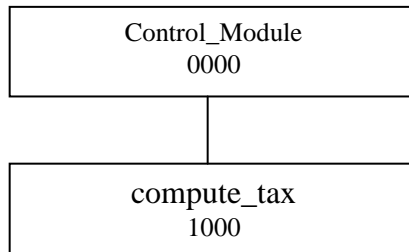
**Problem 1:**

The income tax paid by each person in a country is the sum of the person’s city, provincial and federal taxes. Given the person’s city, provincial and federal tax rates in percentages, the respective taxes are computed as the product of the tax rate and the person’s income. In a country, the city, provincial and federal tax rates are given as 8%, 15% and 25% respectively. You are required to write a C program with functions using only the structure chart given below, to compute and print the (i) the income tax paid by Rose and John given their incomes, and (ii) the city, provincial and federal taxes paid by Rose and John separately. Also, print their names and incomes as shown in the sample input and output below.

Your program should print the results in a nice table format showing the person name, input income for each person with the output of income tax, city, provincial and federal taxes as shown in the sample output. ....(35 marks)

**DO NOT USE GLOBAL VARIABLES.** You should make only parameter calls (it is up to you what type of parameter you should use to write your program), and read all input data as well as print all desired output data in the main function. All tax computation operations should be done in the function compute\_tax. Do not use repetition instruction or a different structure chart. You should call this function once for each person. You may use constant values like tax rate directly in the function.

Use the following structure chart for solving the problem.



**Sample Input:**

Please type the income of Rose:  
60000  
Please type the income of John:  
85000

**Sample Output:**

The taxes paid by these people are:

Person	income	total tax	city	provincial	federal
Rose	60000.00	28800.00	4800.00	9000.00	15000.00
John	85000.00	40800.00	6800.00	12750.00	21250.00

\*\*\*\*\* (note that this line is not part of output of programs) \*\*\*\*\*

**SOLUTION**

```

/*
The income tax paid by each person in a country is the sum of the
person's
city, provincial and federal taxes. Given the person's city, provincial
and
federal tax rates in percentages, the respective taxes are computed as
the
product of the tax rate and the person's income. In a country, the
city,
provincial and federal tax rates are given as 8%, 15% and 25%
respectively.
You are required to write a C program with functions using only the
structure
chart given below, to compute and print the (i) the income tax paid by
Rose
and John given their incomes, and (ii) the city, provincial and federal
taxes
paid by Rose and John separately. Also, print their names and incomes
as
shown in the sample input and output below.

```

```

Your program should print the results in a nice table format showing
the
person name, input income for each person with the output of income
tax,
city, provincial and federal taxes as shown in the sample output.
*/

```

```

#include <stdio.h>

/* function prototype */
void compute_tax(float, float *, float *, float *, float *);

/* This is the main Module 0000 which calls module compute_tax and */

int main(void)
{
char namer[10]="Rose", namej[10]="John";
float rose_income, john_income;
float rose_cityt, rose_provt, rose_fedt, rose_inct;
float john_cityt, john_provt, john_fedt, john_inct;

/* Now the sequence of instructions */
printf("Please type the income of Rose:\n");
scanf("%f", &rose_income);
printf("Please type the income of John:\n");
scanf("%f", &john_income);

compute_tax(rose_income, &rose_cityt, &rose_provt, &rose_fedt,
&rose_inct);
compute_tax(john_income, &john_cityt, &john_provt, &john_fedt,
&john_inct);

printf("The taxes paid by these people are:\n");
printf("Person | income | total tax | city |provincial| federal
\n");

```

```

printf("*****\n"
);
printf("%s | %-6.2f | %-6.2f | %-6.2f | %-6.2f | %-6.2f\n",namer,
rose_income, rose_inct, rose_cityt, rose_provt,rose_fedt) ;
printf("%s | %-6.2f | %-6.2f | %-6.2f | %-6.2f | %-6.2f\n",namej,
john_income, john_inct, john_cityt, john_provt, john_fedt) ;

return 0;
}

/***** The function definition for function compute_tax which accepts
5 grade points and computes both the term and major gpas of a student*/

/* This is called module 1000*/

void compute_tax(float income, float *cityt, float *provt, float *fedt,
float *inct)
{
    *cityt = income * 0.08;
    *provt = income * 0.15;
    *fedt = income * .25;
    *inct = *cityt + *provt + *fedt;
}

```

### Marking Scheme:

- Assign - 5 marks for proper definition of input and output variables/constants
- 5 marks for proper declaration of function prototypes
- 15 marks for correct logic in the control module body including correct use of program instructions and function calls.  
(broken down as 5 marks for correct reading of input data, 5 marks for correct function calls, 5 marks for correct printing of output data and messages)
- 10 marks for function definition (for each function, 6 marks for correct logic and 2 marks for correct function header return and input pars and 2 marks for output parameter passing)
- \*\*\*\*\* the above will make up 35 marks and marks are lost for the following:
  - 10 marks off for improper parameter passing mechanisms.
  - Take 15 marks off if parameters are not used in the solution. Then, mark the solution based on 2 marks for variable declaration, 3 for function prototypes, 5 marks for control module body, 5 marks for function definition and 5 marks for correct printing of output.
  - Take 20 marks off for not using functions at all. Then, mark the solution based on 5 marks for variable declaration, 5 marks for control module body, and 5 marks for correct printing of output.
  - 10 marks off for using a different structure chart.
  - 10 marks off for printing in functions and not in main.
  - 10 marks off for disallowed computation operations in main

2. This question wants you to trace (that is, execute with hand the way the computer would) through the following simple program (with no functions) and fill in the output of the program in the chart provided below for this question. Show how you arrived at your result for full marks. To show your work, you can show by the side the values of all variables in memory as each instruction is executed. (15 marks)

```
#include <stdio.h>
void main(void)
{
    /* Declaration of variables */
    int quarters=2, dimes=1, nickels = 5, pennies = 28;
    int change;
    float notes;

    change = quarters * 25 + pennies % nickels;
    notes = change/100.0;
    printf("change is %d\n", change);
    printf("notes is %0.2f \n", notes);
    printf("other amount is %d \n", change % 100);
}
```

The output printed by the above program's 3 printf are:

Which printf	Output by Program's printf is to be written on this column
First printf	change is 53
2 <sup>nd</sup> printf	notes is 0.53
3 <sup>rd</sup> printf	Other amount is 53

**Marking Scheme:**

Assign 5 marks for each correct tracing of changes and printing of correct values by the printf for a total of 15 marks for the three printf. We allowed a mark of only 2 for each correct tracing with wrong calculation.

3. This question wants you to trace (execute with hand) through a program that has functions and uses call-by-reference, call-by-value as well as some global and local variables. It also wants you to identify important parts of this program as described later.

Given the following program solution to a problem, answer questions 3a to 3c using this solution. Each of the 10 answers in question 3a to 3c is worth 5 marks. See input data below. (50 marks total)

```
#include <stdio.h>

// Global Variables

int      num4 = 5, num5=8, num6=4, sum2=0, product2=0;
float    average2=0.0;

/* fn prototypes */
int FindSum (int, int, int);
void FindProd (int, int, int, int *);
void FindAve (int, int, float);

int main (void)
{
    int      num1, num2, num3, sum1=0, product1=0;
    float    averagel=0.0;

    /* Now write the instructions of the control module */

    scanf("%d %d %d", &num1, &num2, &num3);
    // 1st printf in main follows
    printf("%d %d %d %d %d %d\n", num1, num2, num3, num4, num5,num6);

    sum1=FindSum(num1, num2, num3);
    sum2=FindSum(num4, num5, num6);
    // 2nd printf in main is next
    printf("%d %d\n", sum1, sum2);

    FindProd(num1, num2, num3, &product1);
    FindProd(num4, num5, num6, &product2);
    //3rd printf in main is next
    printf("%d %d\n", product1, product2);

    FindAve(sum1,3, averagel);
    FindAve(sum2,3, average2);
    //4th printf in main is next
    printf("%0.2f %0.2f\n", averagel, average2);
    return 0;

} // end of main

/* Now begin the definitions of the functions */
int FindSum(int first, int second, int third)
{
    int sum = 0;
    third = (first + second + third);
```

```

        return sum;
    }

int divisor = 3;           // more global variables

void FindProd (int num1, int num2, int num3, int *prod)
{
    *prod = (num1 * num2 * num3) % divisor;
}

void FindAve (int sum, int knt, float ave)
{
    ave = (float)sum/knt;
}

```

3a.) With a sample test data 10 5 4 typed at the scanf prompt for the values of the variables Num1, Num2 and Num3 respectively, write the values printed by the four printf instructions in main in the table below. Show your work to get full marks here. To show your work, you can show by the side the values of all variables in memory as each instruction is executed (that is, show your tracing work).

After printf instruction at Position..	Values Printed by CPU in correct order are
First printf in main	10 5 4 5 8 4
Second printf in main	0 0
Third printf in main	2 1
Fourth printf in main	0.00 0.00

3b.) List all global and local variables of the functions (that each function can use) in the table below:

Function	Global variables (that can be used by each of the functions)	Local variables (not including its formal parameters)
main	num4, num5, num6, sum2, product2, average2	num1, num2, num3, sum1, product1, average1
FindProd	num4, num5, num6, sum2, product2, average2=0, divisor	none

3c.) Fill in the correct description of listed parts of the program solution of question 3 in the table below: (Place your answer to each question in the column on its right).

A function call in the main is ..(provide one complete instruction)	Any one of the following is a fn call sum1=FindSum(num1, num2, num3); sum2=FindSum(num4, num5, num6); FindProd(num1, num2, num3,&product1); FindProd(num4, num5, num6,&product2);
---	---

	<pre>FindAve(sum1,3, average1); FindAve(sum2,3, average2);</pre>
A list of formal parameters in the program is. Provide one complete list with the brackets..	<pre>(int first, int second, int third) used in FindSum or (int num1, int num2, int num3, int *prod) used in FindProd, or (int sum, int knt, float ave) used in FindAve.</pre>
A list of actual parameters in the program is. Provide one complete list with the brackets..	<pre>(num1, num2, num3) or (num4, num5, num6) used in calls to FindSum, or (num1, num2, num3, &amp;product1) or (num4, num5, num6, &amp;product2) used in calls to FindProd, or (sum1,3, average1) or (sum2,3, average2) used in calls to finaAve.</pre>
One function prototype in the program of Problem 3 is ..	<pre>int FindSum (int, int, int); or void FindProd (int, int, int, int *); or void FindAve (int, int, float);</pre>

**Marking Scheme:**

Take 5 marks off for any of the 10 questions in 3a to 3c incorrectly answered. For question 3a, we allow only a maximum mark of 1 (that is – 1.5) for each correct tracing or correct calculation with written answer wrong. For question 3b, we deduct 0.5 mark for each variable in the list missed or for each extra wrong variable included.