

60-140-01 ASSIGNMENT #5 SOLUTION
Handed Out: Thurs. Oct 29, 2015 for (60-140-01 and 60-140-02)
Due: Thurs Nov 12, 2015 for (60-140-01 and 60-140-02)
Total: 50 marks

Objective: To write a C program to solve a problem using **functions** and **function calls**, **decision** and **repetition instructions**, but with no arrays. Also, practice on use of flowchart, internal documentation.

Scope: Assignment covers materials up until end of chapter 7.

Other Things to learn from Assignment: How to use decision (*if* and *switch case* instructions) as well as repetition in problem solving.

Important: Do not forget to type in your *full name, student number, lecture section number, lab [section number]* and *date* in BOTH the algorithm and source C program files.

Electronic Assignment Submission:

03-60-140-1 students: email script file to cs140_01@cs.uwindsor.ca with subject including: Name, student id, lecture section, lab [section], assignment #5 (in the subject of the mail submission of script file).

03-60-140-2 students: email script file to cs140_02@cs.uwindsor.ca with subject including: Name, student id, lecture section, lab [section], assignment #5 (in the subject of the mail submission of script file).

*Only the assignments currently due that are submitted to this site within two days before and by the due date, are retrieved for marking. Others are deleted soon after.

If one day you run your own company, you will find that getting your taxes correct is an important part of doing business. Write a program capable of computing the amount of tax you owe given the following tax table, showing the marginal tax rate for six ranges of income (also called six income brackets).

Tax Bracket	Income	Marginal tax rate
1	0 - 10,000	5%
2	\$10,001 - 20,000	10%
3	\$20,001 - 30,000	15%
4	\$30,001 - 50,000	20%
5	\$50,000 - 100,000	25%
6	> \$100,000	30%

Let T_k be the tax for tax bracket k and income i , the tax for each income and tax bracket is computed using the formula on the rightmost end of the line.

$T_1 = 5\% * i$	$\rightarrow 0.05 * i$
$T_2 = T_1 + 10\% * (i - 10,000)$	$\rightarrow 500 + 0.10*(i - 10,000)$
$T_3 = T_1 + T_2 + 15\% * (i - 20,000)$	$\rightarrow 1500 + 0.15*(i - 20,000)$
$T_4 = T_1 + T_2 + T_3 + 20\% * (i - 30,000)$	$\rightarrow 3000 + 0.20*(i - 30,000)$
$T_5 = T_1 + T_2 + T_3 + T_4 + 25\% * (i - 50,000)$	$\rightarrow 7000 + 0.25*(i - 50,000)$
$T_6 = T_1 + T_2 + T_3 + T_4 + T_5 + 30\% * (i - 100,000)$	$\rightarrow 19500 + 0.30*(i - 100,000)$

Your program should be able to compute any number of n taxes given n incomes and output for each income, its tax bracket and income tax as shown in the sample input and output below.

- The input should be from the keyboard and consist of:
 - The number of incomes you want to calculate the tax for, n
 - The n incomes
- The output should be to the screen and consist of:
 - The tax Bracket
 - The income tax.
- Your program must use the following three functions as described:
 1. **Tax_Bracket_Identification.** This function will identify the tax Bracket.
This function is:
 - Of *type int*
 - Has one *call-by-value* argument (income)
 - Uses nested if-else statements
 2. **Tax_Calculation.** Will calculate the tax for each income scanned
 - Of *type void*
 - Has three arguments:
 - The income (*call-by-value*)
 - The Tax Bracket (*call-by-value*)
 - The Tax (*call-by-reference*)
 - Uses a Switch case statement to calculate the Income tax.
 3. **PrintAll.** This function will Print the tax bracket and income tax. It is:
 - Of *type void*
 - Has two arguments:
 - The Tax bracket (*call-by-value*)
 - The income tax *call-by-value*)

Sample Input and Output

Please type in the number of incomes to be processed: 3

Please scan in income number 1: 85471

Your Tax bracket is: 5

Your Tax is: 15867.75

Please scan in income number 2: 10000000

Your Tax bracket is: 6
Your Tax is: 2989500.00

Please scan in income number 3: 28000

Your Tax bracket is: 3
Your Tax is: 2700.00

Thank you for using our Computing tax program.
Please visit us again!!!

You are required to provide the structure chart, program, flowchart with internal documentations (comments and remarks) using correct logic structures and instructions. Use switch_case to compute the tax.

You are required to:

1. Type the C program solution into a source file called your **userid_asn5.c**.
2. Hand in for marking the following:
 - a. **Your script file called userid_script5.txt, created with script userid_script5.txt should show:**
 - Your source program with **cat userid_asn5.c**
 - The compilation of the program with **cc userid_asn5.c**
 - The running of the program with the input and output data shown using **./a.out**
 - Remember to exit for
 - b. The **Structure chart** and **Flowchart** for your source program. You can draw these two charts with MS word and Print and attach as a second file. You can also draw them with your hand neatly, scan and also attach as a second file.

Observe the following conditions

1. All function calls should be parameter calls.
2. Give the programs and flowcharts including internal documentations.

The following marking scheme will be used:

Use of function and Parameters	→ 10 marks
Understanding and solving correctly, structure and flowcharts	→ 10 marks
Algorithms (efficiency), e.g., use of correct repetition structure	→ 10 marks
Internal documentation and readability of solution (indentation)	→ 10 marks
Correct input and output data in script file	→ 10 marks

Make sure you have your name, student id and lab on all documents handed in.

60-140 ASSIGNMENT # 5 Solution

1. Script File

```
Script started on Fri 23 Oct 2015 01:13:38 PM EDT
cezeife@bravo:~/fall15/assignmt$ cat cezeife_asn5.c
/* This program internal documentation includes what problem being solved
as follows:
```

```
Write a program capable of computing the amount of tax you owe given the
following tax table, showing the
marginal tax rate for six ranges of income (also called six income
brackets).
```

Tax Bracket	Income	Marginal tax rate
1	0 - 10,000	5%
2	\$10,001 - 20,000	10%
3	\$20,001 - 30,000	15%
4	\$30,001 - 50,000	20%
5	\$50,000 - 100,000	25%
6	> \$100,000	30%

```
Let Tk be the tax for tax bracket k and income i , the tax for each income
and tax bracket is computed using
the formula on the rightmost end of the line.
```

$$\begin{aligned} T1 &= 5\% * i && = 0.05 * i \\ T2 &= T1 + 10\% * (i - 10,000) && = 500 + 0.10*(i - 10,000) \\ T3 &= T1 + T2 + 15\% * (i - 20,000) && = 1500 + 0.15*(i - 20,000) \\ T4 &= T1 + T2 + T3 + 20\% * (i - 30,000) && = 3000 + 0.20*(i - 30,000) \\ T5 &= T1 + T2 + T3 + T4 + 25\% * (i - 50,000) && = 7000 + 0.25*(i - 50,000) \\ T6 &= T1 + T2 + T3 + T4 + T5 + 30\% * (i - 100,000) && = 19500 + 0.30*(i - 100,000) \end{aligned}$$

```
Your program should be able to compute any number of n taxes given n
incomes and output for each income,
its tax bracket and income tax as shown in the sample input and output
below.
```

```
The input should be from the keyboard and consist of:
```

```
The number of incomes you want to calculate the tax for, n
```

```
The n incomes
```

```
The output should be to the screen and consist of:
```

```
The tax Bracket
```

```
The income tax.
```

```
Your program must use the following three functions as described:
```

```
1. Tax_Bracket_Identification. This function will identify the tax
Bracket.
```

```
This function is:
```

```
Of type int
```

```
Has one call-by-value argument (income)
```

```
Uses nested if-else statements
```

```
2. Tax_Calculation. Will calculate the tax for each income scanned
```

```
Of type void
```

```
Has three arguments:
```

The income (call-by-value)
 The Tax Bracket (call-by-value)
 The Tax (call-by-reference)
 Uses a Switch case statement to calculate the Income tax.
 3. PrintAll. This function will Print the tax bracket and income tax.
 It is:
 Of type void
 Has two arguments:
 The Tax bracket (call-by-value)
 The income tax call-by-value)

Program written by Dr. Christie Ezeife, October 23, 2015.

```

*/

#include <stdio.h>

// Function Prototypes for the 3 functions used follow.
int Tax_Bracket_Indentification (float inc );
void Tax_Calculation (float inc, int tax_br, float *tx);
void PrintAll (int, float);

int main(void)
{
    /* variable dictionary          */
    int i;                          // i is used as for loop index
    int current_income;             // current_income holds counter to the
    jth income to be processed.
    int n;                          // n is used to input number of incomes
    to process.
    int tax_bracket;                // tax_bracket indicates the tax
    bracket for computing tax.
    float income;                   // the income being processed.
    float tax;                       // the income tax to be computed.

    printf ("\n\nPlease type in the number of incomes to be processed:
");
    scanf ("%d", &n);

    /* Using repetition each income is read and the functions
    Tax_Bracket_identification,
    Tax_Calculation and PrintAll are called with appropriate
    parameters to compute
    the income tax using the given formula and print the results.
    Other slightly
    different structure for these functions are possible.

    */

    for (i = 0; i<n; i++)
    {
        current_income = i+1;
        printf("\nPlease scan in income number %d: ", current_income);
        scanf("%f", &income);
    }
}

```

```

        tax_bracket = Tax_Bracket_Indentification (income);
        Tax_Calculation (income, tax_bracket, &tax);
        PrintAll (tax_bracket, tax);

    }
    printf("\n\nThank you for using our Income Tax Computation
Program.\n");
    printf("Please visit us again!!!\n");
    printf("***\n");

    return(0);
}

/* The function definitions follow */

int Tax_Bracket_Indentification (float inc)
{
    int tax_br;

    // First if instruction is to validate that income is within ranges >0

    if (inc < 0 )
        {printf("This income CANNOT be processed\n");
          tax_br = 9;
        }
    else
    if ( inc <= 10000)
        tax_br = 1;
    else
    if (inc <= 20000)
        tax_br = 2;
    else
    if (inc <= 30000)
        tax_br = 3;
    else
        if (inc <= 50000)
            tax_br = 4;
        else
            if (inc <=100000)
                tax_br = 5;
            else
                tax_br = 6;

    return(tax_br);
}

// This function uses switch_case instruction to pick one choice that
matches to the
// income tax bread passed in as parameter.

void Tax_Calculation (float inc, int tax_br, float *tx)
{
    switch (tax_br)

```

```

    {
        case 1: *tx = 0.05 * inc;
                break;
        case 2: *tx = 500 + 0.10*(inc-10000);
                break;
        case 3: *tx = 1500 + 0.15*(inc-20000);
                break;
        case 4: *tx = 3000 + 0.20*(inc-30000);
                break;
        case 5: *tx = 7000 + 0.25*(inc-50000);
                break;
        case 6: *tx = 19500 + 0.30*(inc-100000);
                break;
        default:
                break;
    }

}

void PrintAll (int tax_br, float tx)
{
    if (tax_br !=9)

    {
        printf("\n");
        printf("Your Tax bracket is:\t%d\n", tax_br);
        printf("Your Tax is:\t\t%6.2f\n", tx);
    }
}

```

```

cezeife@bravo:~/fall15/assignmt$ cat cezeife_asn5.c
/* This program internal documentation includes what problem being solved
as follows:

```

Write a program capable of computing the amount of tax you owe given the following tax table, showing the marginal tax rate for six ranges of income (also called six income brackets).

Tax Bracket	Income	Marginal tax rate
1	0 - 10,000	5%
2	\$10,001 - 20,000	10%
3	\$20,001 - 30,000	15%
4	\$30,001 - 50,000	20%
5	\$50,000 - 100,000	25%
6	> \$100,000	30%

Let T_k be the tax for tax bracket k and income i , the tax for each income and tax bracket is computed using the formula on the rightmost end of the line.

$$\begin{aligned}
 T_1 &= 5\% * i && = 0.05 * i \\
 T_2 &= T_1 + 10\% * (i - 10,000) && = 500 + 0.10*(i - 10,000) \\
 T_3 &= T_1 + T_2 + 15\% * (i - 20,000) && = 1500 + 0.15*(i - 20,000) \\
 T_4 &= T_1 + T_2 + T_3 + 20\% * (i - 30,000) && = 3000 + 0.20*(i - 30,000)
 \end{aligned}$$

$$T5 = T1 + T2 + T3 + T4 + 25\% * (i - 50,000) = 7000 + 0.25*(i - 50,000)$$

$$T6 = T1 + T2 + T3 + T4 + T5 + 30\% * (i - 100,000) = 19500 + 0.30*(i - 100,000)$$

Your program should be able to compute any number of n taxes given n incomes and output for each income, its tax bracket and income tax as shown in the sample input and output below.

The input should be from the keyboard and consist of:

The number of incomes you want to calculate the tax for, n

The n incomes

The output should be to the screen and consist of:

The tax Bracket

The income tax.

Your program must use the following three functions as described:

1. Tax_Bracket_Identification. This function will identify the tax Bracket.

This function is:

Of type int

Has one call-by-value argument (income)

Uses nested if-else statements

2. Tax_Calculation. Will calculate the tax for each income scanned

Of type void

Has three arguments:

The income (call-by-value)

The Tax Bracket (call-by-value)

The Tax (call-by-reference)

Uses a Switch case statement to calculate the Income tax.

3. PrintAll. This function will Print the tax bracket and income tax.

It is:

Of type void

Has two arguments:

The Tax bracket (call-by-value)

The income tax call-by-value)

Program written by Dr. Christie Ezeife, October 23, 2015.

```
*/
```

```
#include <stdio.h>
```

```
// Function Prototypes for the 3 functions used follow.
```

```
int Tax_Bracket_Identification (float inc );
```

```
void Tax_Calculation (float inc, int tax_br, float *tx);
```

```
void PrintAll (int, float);
```

```
int main(void)
```

```
{
```

```
    /* variable dictionary          */
```

```
    int i;                          // i is used as for loop index
```

```
    int current_income;             // current_income holds counter to the
```

```
jth income to be processed.
```

```
    int n;                          // n is used to input number of incomes
```

```
to process.
```



```

    int tax_bracket;           // tax_bracket indicates the tax
    bracket for computing tax.
    float income;             // the income being processed.
    float tax;                // the income tax to be computed.

    printf ("\n\nPlease type in the number of incomes to be processed:
");
    scanf ("%d", &n);

    /* Using repetition each income is read and the functions
    Tax_Bracket_identification,
    Tax_Calculation and PrintAll are called with appropriate
    parameters to compute
    the income tax using the given formula and print the results.
    Other slightly
    different structure for these functions are possible.

    */

    for (i = 0; i<n; i++)
    {
        current_income = i+1;
        printf("\nPlease scan in income number %d: ", current_income);
        scanf("%f", &income);
        tax_bracket = Tax_Bracket_Indentification (income);
        Tax_Calculation (income, tax_bracket, &tax);
        PrintAll (tax_bracket, tax);

    }
    printf("\n\nThank you for using our Income Tax Computation
    Program.\n");
    printf("Please visit us again!!!\n");
    printf("***\n");

    return(0);
}

/* The function definitions follow */

int Tax_Bracket_Indentification (float inc)
{
    int tax_br;

    /* First if instruction is to validate that income is within ranges >0
    But accept if this first step is not included */

    if (inc < 0 )
        {printf("This income CANNOT be processed\n");
        tax_br = 9;
        }
    else
        if ( inc <= 10000)
            tax_br = 1;
}

```

```

else
if (inc <= 20000)
    tax_br = 2;
else
if (inc <= 30000)
    tax_br = 3;
else
    if (inc <= 50000)
        tax_br = 4;
    else
        if (inc <=100000)
            tax_br = 5;
        else
            tax_br = 6;

return(tax_br);

}

// This function uses switch_case instruction to pick one choice that
matches to the
// income tax bread passed in as parameter.

void Tax_Calculation (float inc, int tax_br, float *tx)
{
    switch (tax_br)
    {
        case 1: *tx = 0.05 * inc;
                break;
        case 2: *tx = 500 + 0.10*(inc-10000);
                break;
        case 3: *tx = 1500 + 0.15*(inc-20000);
                break;
        case 4: *tx = 3000 + 0.20*(inc-30000);
                break;
        case 5: *tx = 7000 + 0.25*(inc-50000);
                break;
        case 6: *tx = 19500 + 0.30*(inc-100000);
                break;
        default:
                break;
    }

}

}

void PrintAll (int tax_br, float tx)
{
    if (tax_br !=9)

    {
        printf("\n");
        printf("Your Tax bracket is:\t%d\n", tax_br);
        printf("Your Tax is:\t\t%6.2f\n", tx);
    }
}

```

```
}  
}  
cezeife@bravo:~/fall15/assignmt$ ./a.out
```

Please type in the number of incomes to be processed: 3

Please scan in income number 1: 85471

Your Tax bracket is: 5
Your Tax is: 15867.75

Please scan in income number 2: 10000000

Your Tax bracket is: 6
Your Tax is: 2989500.00

Please scan in income number 3: 28000

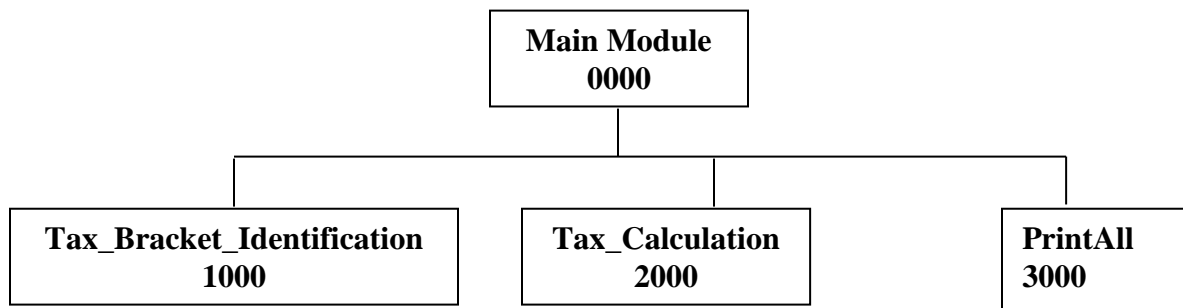
Your Tax bracket is: 3
Your Tax is: 2700.00

Thank you for using our Income Tax Computation Program.
Please visit us again!!!

```
cezeife@bravo:~/fall15/assignmt$ exit  
exit
```

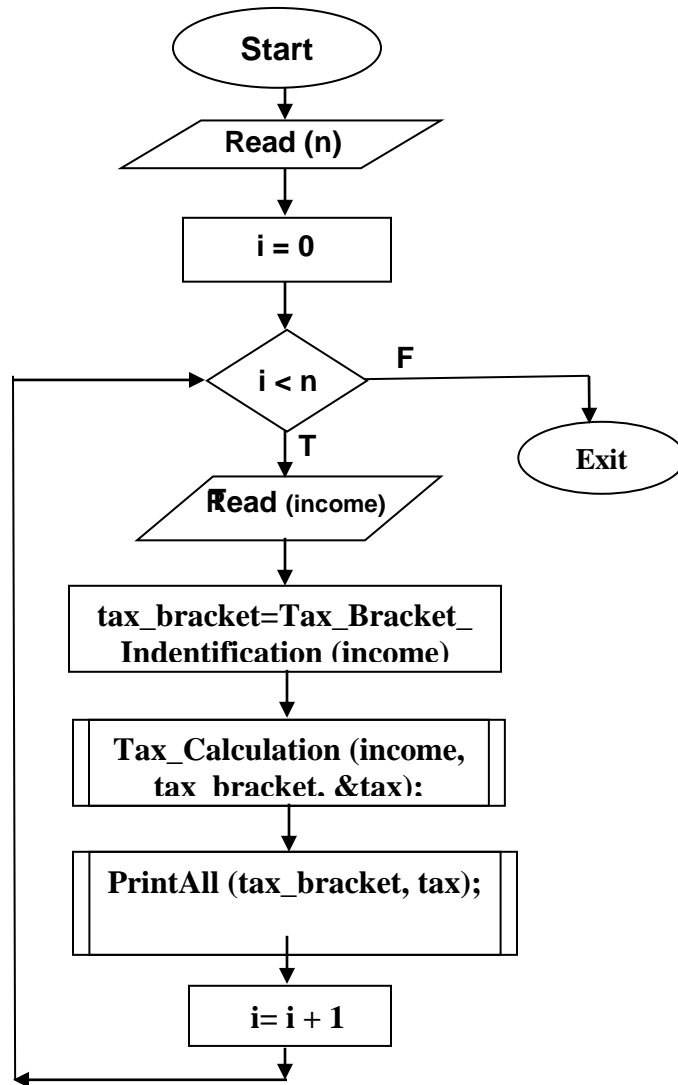
Script done on Fri 23 Oct 2015 01:15:30 PM EDT

2. The structure chart



3. Flowchart
a. Main Module

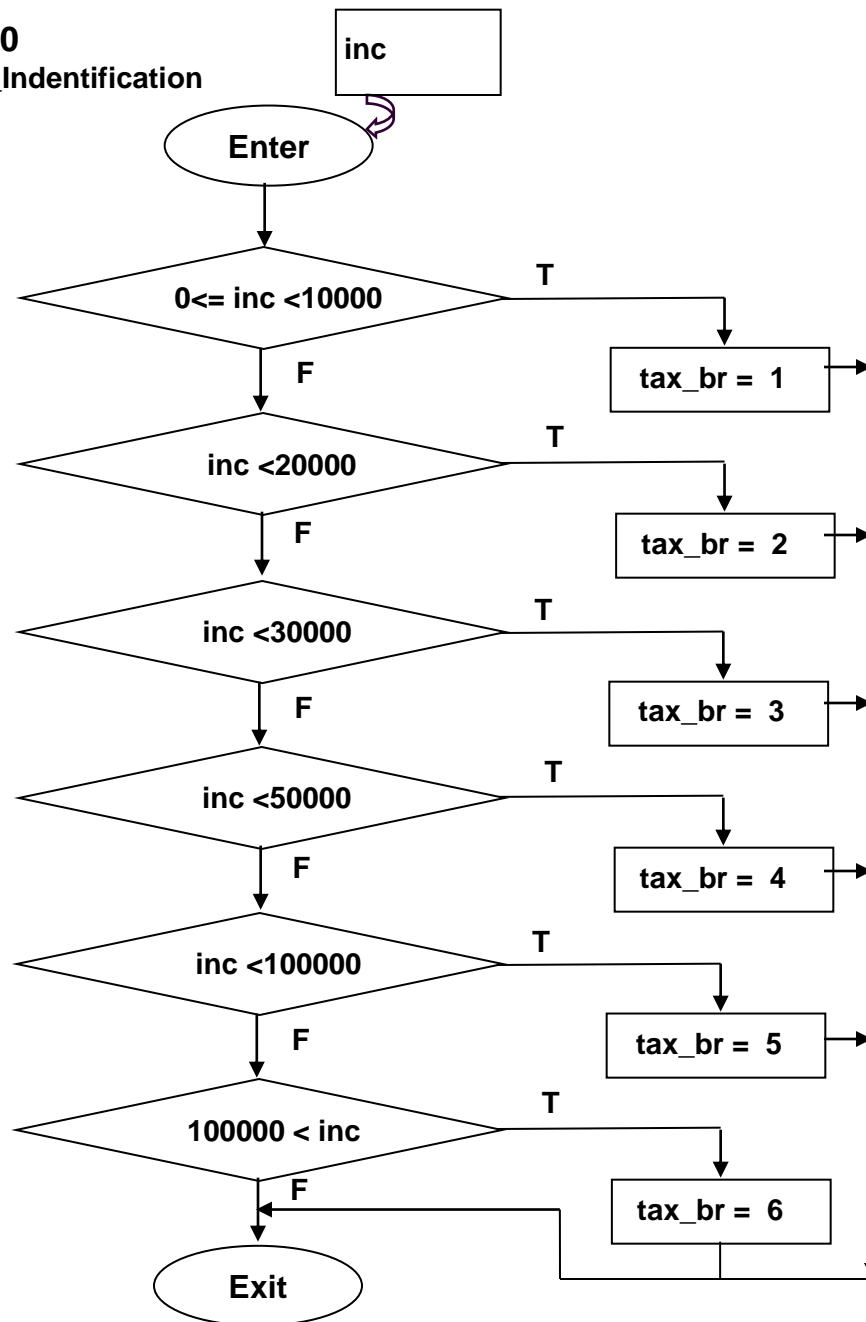
Module 0000



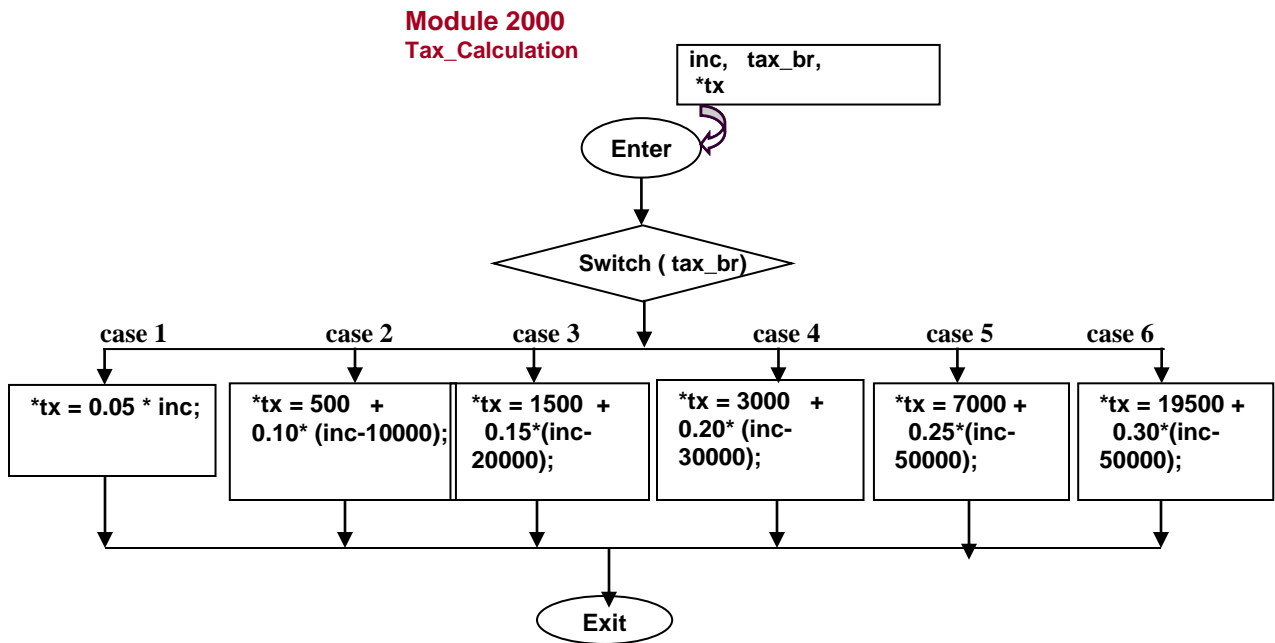
b. Module 1000: Tax_Bracket_Indentification

Module 1000

Tax_Bracket_Indentification



c. Module 1000: Tax_Calculation



d. Module 3000: PrintAll

