ព្រះរាជាណាចក្រកម្ពុជា

ជាតិ សាសនា ព្រះមហាក្សត្រ

វិទ្យាស្ថានបច្ចេកវិទ្យាកម្ពុជា

ដេប៉ាតឺម៉ង់ថ្នាក់ឆ្នាំសិក្សាមូលដ្ឋាន

TP2-Variable

Subject: Data strucure & Programming

Student name	ID	score
1.SOUNG VICHKA	e20240672	…..

Teacher: Mr.ROEUN Pachroth

TP3-Condition1

Problem1- Write a C program to get a number from a user. The program tells us whether this input number is a positive or a negative number.

INPUT:

Enter a number: -100

OUTPUT:

-100 is a negative number.

Code:

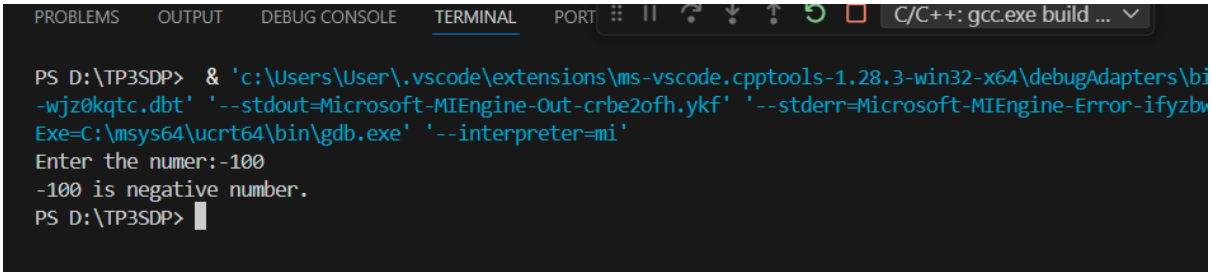
problem1.c problem2.c problem6.c problem5.c

problem1.c > main()

```c
#include <stdio.h>
int main(){
    int num;
    printf("Enter the numer:");
    scanf("%d",&num);
    if (num>0){
        printf("%d is positive number.",num);
    }else {
        printf("%d is negative number.",num);
    }
}
```

- **#include <stdio.h>**
  - This line tells the compiler to include the **Standard Input/Output library**, which allows you to use functions like printf() (for output) and scanf() (for input).
- **int main()**
  - The **main function** is where your program starts running.
  - Every C program must have a main() function.
- **int num;**
  - This declares an **integer variable** named num.
  - It will store the number that the user enters.
- **printf("Enter the number:");**
  - This prints a message asking the user to type a number on the screen.
- **scanf("%d", &num);**
  - This reads an **integer** (%d means integer) from the keyboard.
  - The &num means "store the input value in the variable num".
- **if (num > 0)**
  - This is a **condition**: it checks whether the number is **greater than zero**.
  - If the condition is true, it runs the code inside the { } block just below.
- **printf("%d is positive number.", num);**
  - If the number is greater than 0, this line prints something like:
    👉 "5 is positive number."
- **else { printf("%d is negative number.", num); }**
  - If the number is **not greater than 0**, the program goes to this else block.
  - It prints something like:
    👉 "-3 is negative number."

Result:



Problem2- Write a C program to get a number from a user. The program tells us whether this input number is either one of the following categories below: A) negative and odd numbers, B) negative and even numbers, C) positive and odd numbers, and D) positive and even numbers.

INPUT:

Enter a number: 50

OUTPUT:

It is in the category D because it is a positive number and it is also an even number.

Code:

problem1.c | problem2.c | problem6.c | C/C++: gcc.exe build ... | prob | problemm4.exe

problem2.c > ...

```c
#include<stdio.h>
int main(){

    int num;
    printf("Enter a number: ");
    scanf("%d", &num);

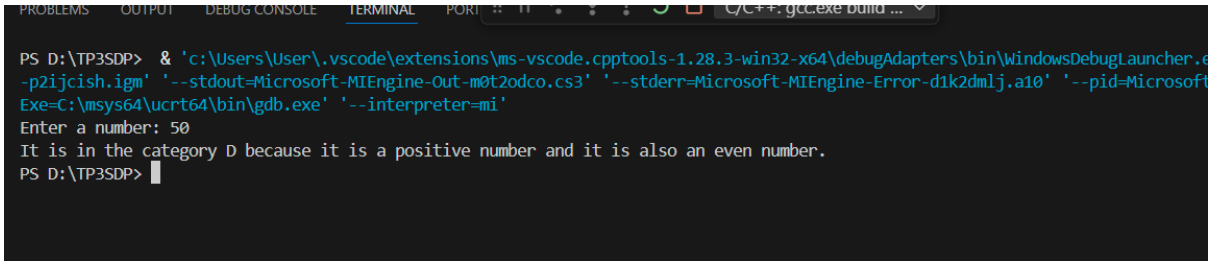
    if(num < 0 && num % 2 != 0){
        printf("It is in the category A because it is a negative number and it is also an odd number.");
    }else if(num < 0 && num % 2 == 0){
        printf("It is in the category B because it is a negative number and it is also an even number.");
    }else if(num > 0 && num % 2 != 0){
        printf("It is in the category C because it is a positive number and it is also an odd number.");
    }else{
        printf("It is in the category D because it is a positive number and it is also an even number.");
    }

}

```

- **#include <stdio.h>**
  - This line includes the **Standard Input Output** library.
    It allows the program to use printf() (to display text) and scanf() (to read input).
- **int main() { ... }**
  - This is the **main function**, where your program starts running.
- **int num;**
  - Here, you declare a variable num of type **integer**.
    It will store the number entered by the user.
- **printf("Enter a number: ");**
  - This prints a message asking the user to input a number.
- **scanf("%d", &num);**
  - This reads an **integer** value from the keyboard and stores it into the variable num.
    The & means “address of” — it tells the program where to store the value.

Result:



Problem3- Write a C program to ask a user for 7 numbers (7 variables). Find the minimum number and display on screen.

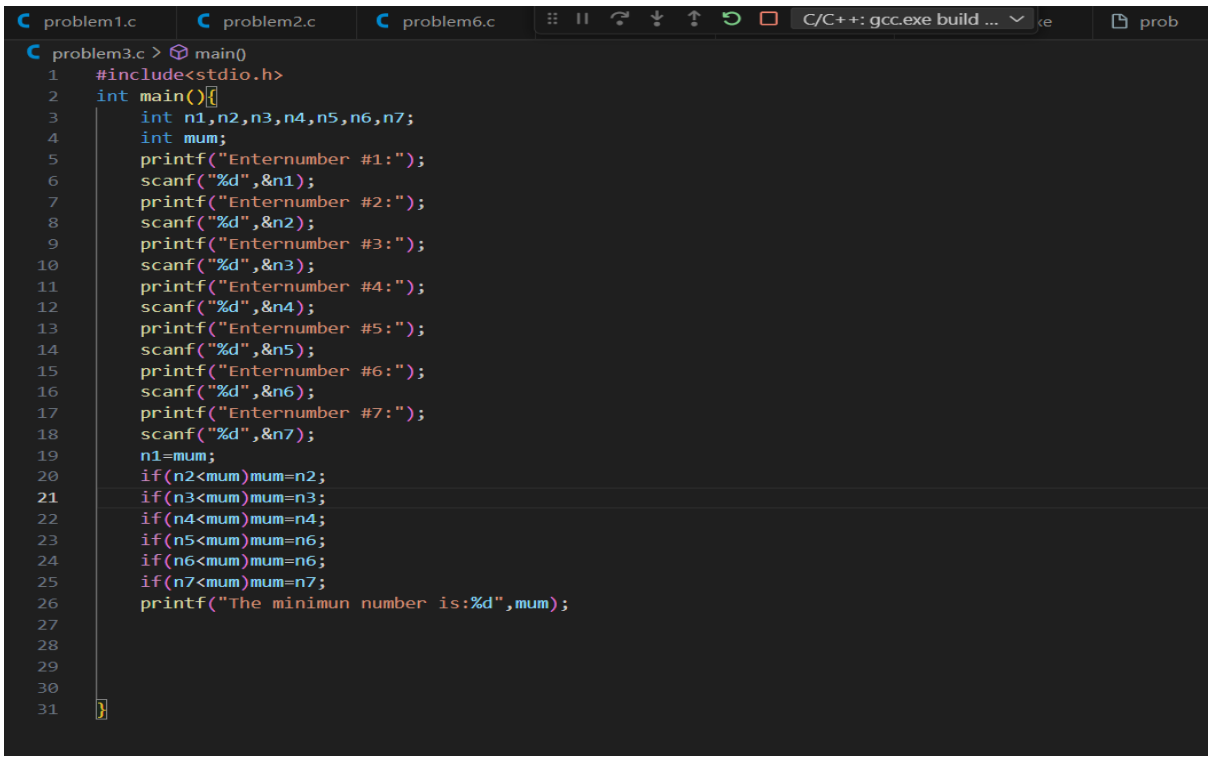
INPUT:

Enter number #1: 2
Enter number #2: 99
Enter number #3: 1
Enter number #4: 2
Enter number #5: 9
Enter number #6: -7
Enter number #7: 5

OUTPUT:

The minimum number is -7.

Code:



- **#include<stdio.h>**

  - This line includes the **Standard Input Output (stdio)** library.
  - It allows you to use functions like printf() (to display text) and scanf() (to read input from the user).

- **int main(){**

  - This marks the **start** of the main function — where the program begins execution.

- **int n1,n2,n3,n4,n5,n6,n7;**
- int mum;

- These lines declare **integer variables**.
  - n1 to n7 are used to store the 7 numbers entered by the user.
  - mum will be used to store the **minimum number** among them.

- printf("Enternumber #1:");
- **scanf("%d",&n1);**

  - printf() displays a message asking the user to enter the **first number**.
  - scanf("%d", &n1) reads that number and stores it in the variable n1.
  - The symbol & means **"address of"**, telling the program where to store the input value.

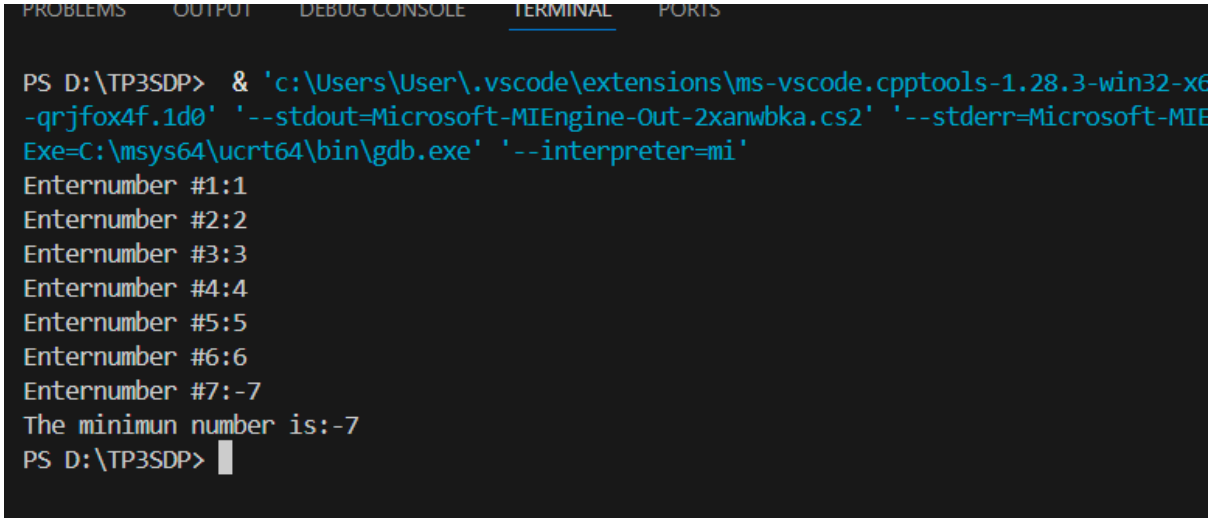
  - (Same explanation applies to n2, n3, …, n7 — each line asks for and stores another number.)
- **if(n2 < mum) mum = n2;**
- **if(n3 < mum) mum = n3;**
- **if(n4 < mum) mum = n4;**
- **if(n5 < mum) mum = n5;**
- **if(n6 < mum) mum = n6;**
- **if(n7 < mum) mum = n7;**

  - Each if statement checks whether a number is **smaller** than the current mum.
  - If true, it updates mum to that smaller number.
  - After all comparisons, mum will contain the **minimum value** among all the inputs.

- **printf("The minimun number is:%d", mum);**

  - This line prints the smallest number (stored in mum).
  - %d is a placeholder for an integer value.

Result:



Problem4- Write a C program to ask a user for 10 numbers (10 variables). What are the max and min numbers among these 10 numbers?

INPUT:

Enter number #1: 10
Enter number #2: 15
Enter number #3: 33
Enter number #4: 5
Enter number #5: 0
Enter number #6: 45
Enter number #7: 2
Enter number #8: 4
Enter number #9: 6
Enter number #10: 8

OUTPUT:

Among these 10 numbers (10, 15, 33, 5, 0, 45, 2, 4, 6, 8), the max is 45 and the min is 0.

Code:

problem2.c | problem6.c | problem5.c | problem7.c | i lov

problem4.c > main()

```c
#include <stdio.h>

int main() {
    int a1, a2, a3, a4, a5, a6, a7, a8, a9, a10;
    int min, max;
    int printf(const char *__restrict__ _Format, ...)
    printf("Enter number #1: ");
    scanf("%d", &a1);
    printf("Enter number #2: ");
    scanf("%d", &a2);
    printf("Enter number #3: ");
    scanf("%d", &a3);
    printf("Enter number #4: ");
    scanf("%d", &a4);
    printf("Enter number #5: ");
    scanf("%d", &a5);
    printf("Enter number #6: ");
    scanf("%d", &a6);
    printf("Enter number #7: ");
    scanf("%d", &a7);
    printf("Enter number #8: ");
    scanf("%d", &a8);
    printf("Enter number #9: ");
    scanf("%d", &a9);
    printf("Enter number #10: ");
    scanf("%d", &a10);
    min = a1;
    if (min > a2) min = a2;
    if (min > a3) min = a3;
    if (min > a4) min = a4;
    if (min > a5) min = a5;
    if (min > a6) min = a6;
    if (min > a7) min = a7;
    if (min > a8) min = a8;
    if (min > a9) min = a9;
    if (min > a10) min = a10;
    max = a1;
    if (max < a2) max = a2;
    if (max < a3) max = a3;
    if (max < a4) max = a4;
    if (max < a5) max = a5;
    if (max < a6) max = a6;
    if (max < a7) max = a7;
    if (max < a8) max = a8;
    if (max < a9) max = a9;
    if (max < a10) max = a10;

    printf("\nThe minimum is: %d\n", min);
    printf("The maximum is: %d\n", max);

    return 0;
}
```

- **int a1, a2, a3, a4, a5, a6, a7, a8, a9, a10;**
- **int min, max;**
  - These lines declare integer variables.
  - a1 to a10 will store 10 numbers entered by the user.
  - min and max will later store the smallest and largest numbers respectively.
- **printf("Enter number #1: ");**
- **scanf("%d", &a1);**
  - printf() displays a message asking the user to enter the first number.
  - scanf("%d", &a1) reads that number from the keyboard and stores it in the variable a1.
  - The symbol & means "address of", telling the program where to store the value.
- **printf("Enter number #2: "); scanf("%d", &a2);**
- **printf("Enter number #3: "); scanf("%d", &a3);**
- **….**
- **printf("Enter number #10: "); scanf("%d", &a10);**
  - After this, the user has entered 10 numbers which are stored in variables a1 → a10.

- **min = a1;**

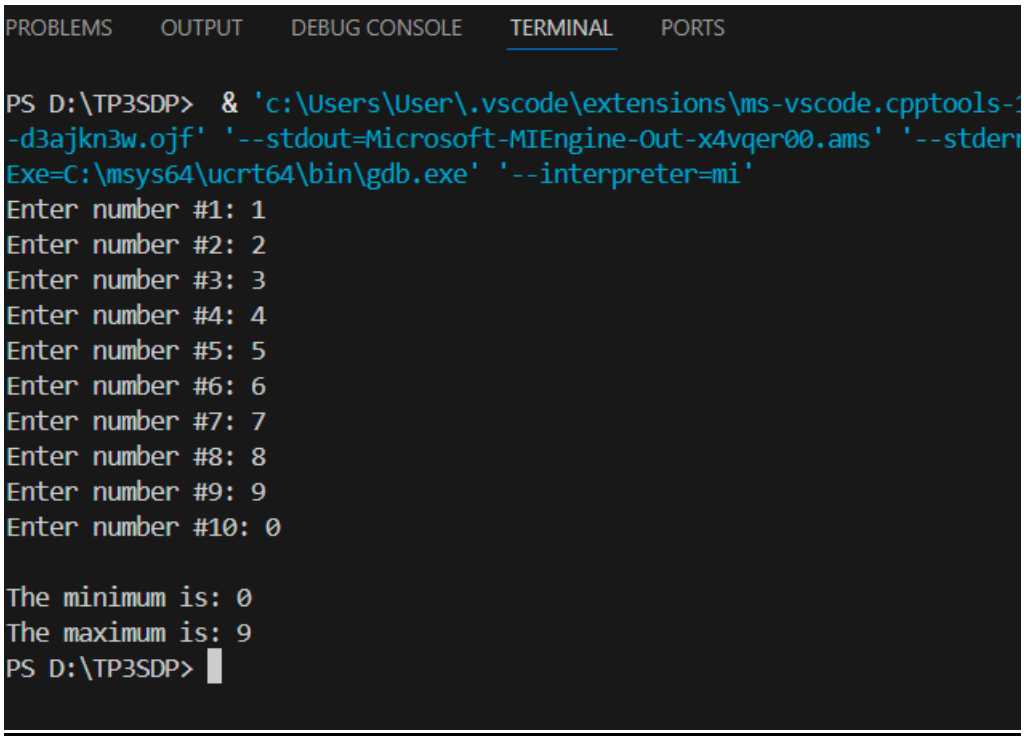
- Here, we assume the first number (a1) is the smallest.
- Then we compare it with all the other numbers one by one.
- **if (min > a2) min = a2;**
- **if (min > a3) min = a3;**
- **if (min > a4) min = a4;**
- **if (min > a5) min = a5;**
- **if (min > a6) min = a6;**
- **if (min > a7) min = a7;**
- **if (min > a8) min = a8;**
- **if (min > a9) min = a9;**
- **if (min > a10) min = a10;**
  - Each if checks if another number is smaller than the current min.
  - If true, it updates min with that smaller number.
  - After all comparisons, min will hold the smallest value entered by the user.

- **Finding the Maximum Value**
- **max = a1;**
  - Now we assume the first number is the largest.
  - Then we compare it with the others.
- **if (max < a2) max = a2;**
- **if (max < a3) max = a3;**
- **if (max < a4) max = a4;**
- **if (max < a5) max = a5;**
- **if (max < a6) max = a6;**
- **if (max < a7) max = a7;**
- **if (max < a8) max = a8;**
- **if (max < a9) max = a9;**
- **if (max < a10) max = a10;**
  - Each if checks if another number is greater than the current max.

- If true, it updates max with that larger number.
- After checking all, max will store the largest value among all inputs.

- **Display the Results**
- **printf("\nThe minimum is: %d\n", min);**
- **printf("The maximum is: %d\n", max);**
  - These lines print out the smallest and largest numbers found.
  - %d is used to display an integer value.

Result:



Problem5- Write a C program to solve the quadratic equation. $ax^2 + bx + c = 0$.
Ask a user to input the coefficients a, b and c. The program calculates delta and find roots of the equation. Display the result (roots of the equation) on screen.

INPUT:

Input a: 1
Input b: 4
Input c: 3

OUTPUT:

The delta value of this equation equals 4.
So this equation has two roots: $x1 = -1$ and $x2 = -3$

Remark: Make sure you verify the result from the program whether it is correct if we do it manually by hand.

Code:

problem5.c > main()

```c
#include<stdio.h>
#include<math.h>
    int main(){
        int a ,b ,c;
        int x1,x2;
        int delta;
        printf("Input a:");
        scanf("%d",&a);
        printf("Input b:");
        scanf("%d",&b);
        printf("Input c:");
        scanf("%d",&c);
        delta=b*b-4*a*c;
        printf("The delta value of this equation%d",delta);
        if (delta>0){
            x1=(-b+sqrt(delta))/2*a;
            x2=(-b-sqrt(delta))/2*a;
            printf("So the equation have two root x1=%dand x2=%d",x1,x2);
        } else if (delta==0){
            x1=x2=-b/2*a;
            printf("So the equation have the same root x1=x2=%d",x1);
        } else if (delta<0){
            printf("So the equation have no root");
        }
    }
```

- **#include<stdio.h>**

  - This line includes the **Standard Input/Output library**.
  - It allows the program to use functions like printf() (to display messages) and scanf() (to read user input).

- **#include<math.h>**

  - This includes the **Math library**, which allows you to use mathematical functions like sqrt() to calculate square roots.

- **int main(){**

  - This marks the **start of the main function**, where the program begins running.

- **int a, b, c;**
- **int x1, x2;**
- **int delta**;

  - a, b, and c ⟶ store the coefficients of the quadratic equation ax2+bx+c=0ax^2 + bx + c = 0ax2+bx+c=0
  - x1, x2 ⟶ will store the roots (solutions) of the equation.
  - delta ⟶ stores the discriminant, calculated as Δ=b2−4ac\Delta = b^2 - 4acΔ=b2−4ac.

- **User Input:**
- **printf("Input a:");**
- **scanf("%d",&a);**
- **printf("Input b:");**
- **scanf("%d",&b);**
- **printf("Input c:");**
- **scanf("%d",&c);**

  - printf() asks the user to enter the values of a, b, and c.
  - scanf("%d", &a) reads the integer value entered by the user and stores it in the variable a.
    (The symbol & means “address of”, telling the program where to store the value.)

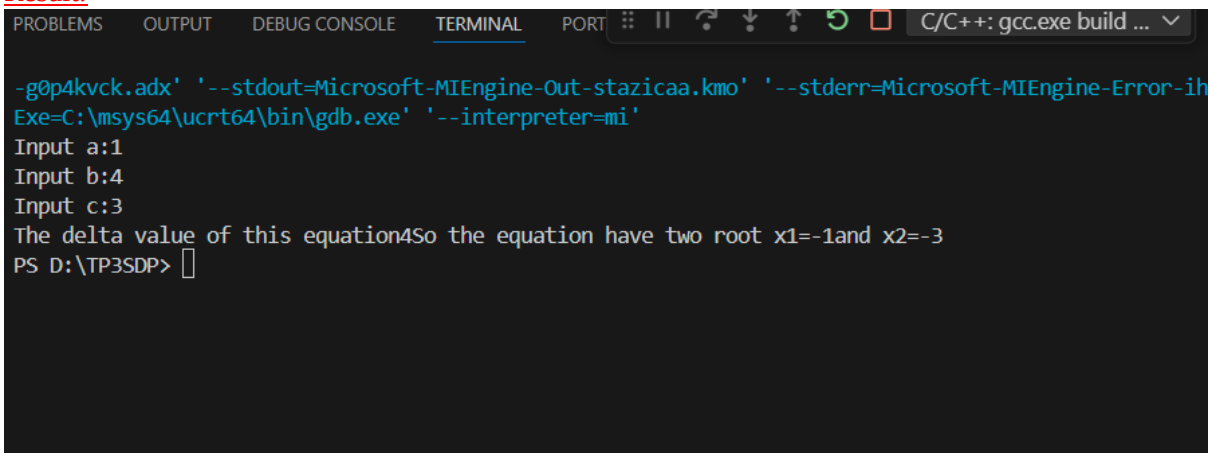
- Compute Delta:
- delta = b*b - 4*a*c;
- printf("The delta value of this equation%d", delta);

  - delta (Δ) is calculated using the formula:
    - Δ=b2−4ac\Delta = b^2 - 4acΔ=b2−4ac
  - printf() displays the value of delta.

- **If delta > 0**
- x1 = (-b + sqrt(delta)) / 2*a;
- x2 = (-b - sqrt(delta)) / 2*a;
- printf("So the equation have two root x1=%d and x2=%d", x1, x2);

- If delta == 0
  - x1 = x2 = -b / 2*a;
  - printf("So the equation have the same root x1=x2=%d", x1);
- **if delta < 0**
  - printf("So the equation have no root");

Result:



Problem6- Write a C program to tell how much tax a person should pay based on his/her salary and gender. The program ask a user for gender, name, and salary (US dollar). The tax is computed according to the following rules:

-pay tax 10%, for male and his salary is more than 400 USD.
-pay tax 5%, for male and his salary is from 200 USD to 400 USD.
-pay tax 5%, for females and her salary is more than 500 USD.
-pay tax 3%, for females and her salary is from 250 USD to 500 USD.
-Otherwise, no need to pay tax.

INPUT:

Enter gender: F
What is your name?: Sokha
Your salary in USD: 450

Output:

Hi Dara, based on your given information, the tax salary that
You need to pay: 13.5 USD.

Code:

problem1.c | problem2.c | problem6.c | C/C++: gcc.exe build ... | i love you.e

problem6.c > ...

```c
#include<stdio.h>
int main(){
char gender;
char name[20];
float salary;
float tax;
printf("Enter gender(M/F) :");
scanf("%c",&gender);
printf("What is your name?");
scanf("%s",&name);
printf("Your salary in USA:");
scanf("%f",&salary);
if (gender=='M'|| gender=='m'){
    if(salary>400)
    tax=salary*0.1;
   else if (salary<=200&&salary<=400)
   tax=salary*0.05;
} else if (gender=='F'||gender=='f'){
    if(salary>500)
    tax=salary*0.05;
    else if (salary>=250&&salary<=500)
    tax=salary*0.03;
} else{
    printf("Otherwise,no need to play tax");
}
printf(" Hi %s,based on your information,the tax salary that\n You need to pay :%.2f",name,tax);


}
```

- **#include<stdio.h>**
    - These lines tell the compiler to use **standard input/output functions** like printf() and scanf().

- **Variable declarations**

- char gender;
- char name[20];
- float salary;
- float tax;

You create:

- gender: a character variable (M or F).
- name[20]: a string to store your name (max 19 letters).
- salary: a floating number for your income.
- tax: a floating number to store the tax you'll calculate later.

- **Asking the user for gender**

- **printf("Enter gender(M/F) :");**
- **scanf("%c",&gender);**

    - The program asks the user:
    - Enter gender (M/F):
    - Then stores your answer (like M or F) inside gender.

- **Asking for the name**

- printf("What is your name?");
- scanf("%s",&name);

It asks:

- What is your name?
- Then saves your name into name.

- **Asking for salary**

- printf("Your salary in USA:");
- scanf("%f",&salary);

It asks:

- Your salary in USA:
- and saves it as a floating-point number in the variable salary.

- **Start checking gender and salary**

Now the program decides **how much tax** you must pay depending on your gender and income.

- ***For males:***

```
if (gender=='M'|| gender=='m'){
   if(salary>400)
      tax=salary*0.1;
   else if (salary<=200&&salary>=400)
      tax=salary*0.05;
}
```

- If the gender is **M or m**, it checks your salary.
    - If salary > 400 → tax = 10% of salary.
    - If salary is between 200 and 400 → tax = 5%.

For females:

```
else if (gender=='F'||gender=='f'){
   if(salary>500)
      tax=salary*0.05;
   else if (salary>=250&&salary<=500)
      tax=salary*0.03;
}
```

- If the gender is **F or f**, it checks your salary.
    - If salary > 500 → tax = 5%

- If salary is between 250 and 500 → tax = 3%

- *Otherwise (not M or F)*

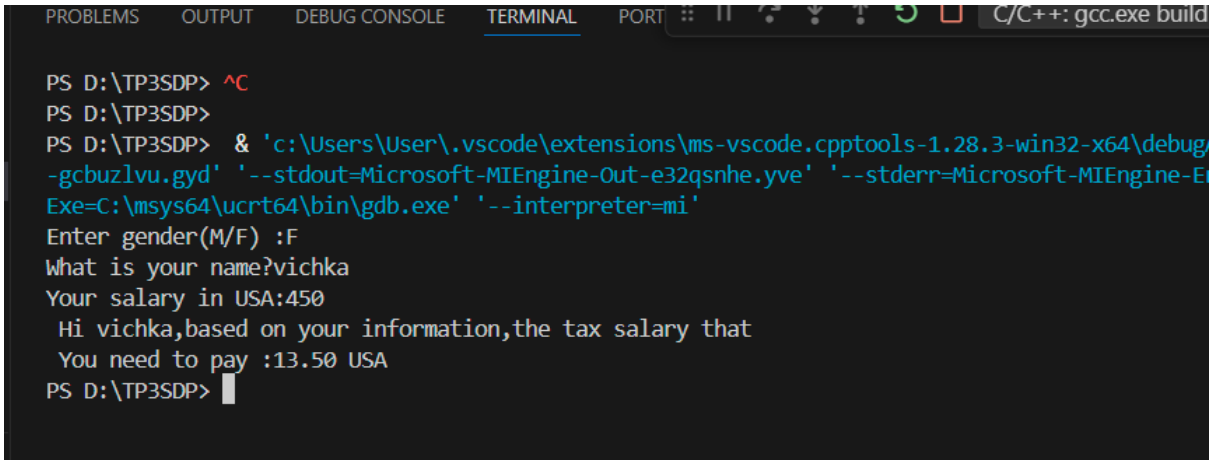
```
else{
   printf("Otherwise,no need to play tax");
}
```

- If the user didn't type M or F, it prints:
- Otherwise, no need to play tax
  (But this should say **"pay tax"**, not *play tax* 😄)

Printing the result

- printf(" Hi %s,based on your information,the tax salary that\n You need to pay :%.2f",name,tax);

Result:



Problem7- Write a C program to compute the average score of a student over 3 subjects Math, English and Computer. Then tell which grade this student could get. The grading is based on the following rules:

Average score:	**Grade**
86 - 100	A
81 - 85	B
70 - 80	C
61 - 69	D
50 - 60	E
Below 50	F

INPUT:
Math score: 75.5
English score: 90
Language score: 85.5
Output:
Average score is: 83.66
This student gets grade B.

Code:

problem1.c | problem2.c | problem6.c | problem5.c | problem6.exe

problem7.c > main()

```c
#include<stdio.h>
int main(){
    float math, english, computer;
    float average;
    printf("Math score: ");
    scanf("%f", &math);
    printf("English score: ");
    scanf("%f", &english);
    printf("Computer score: ");
    scanf("%f", &computer);
    average = (math + english + computer) / 3;
    printf("Average score is: %.2f\n", average);
    if (average >= 86 && average <= 100) {
        printf("This student gets grade A\n");
    }
    else if (average >= 81 && average <= 85) {
        printf("This student gets grade B\n");
    }
    else if (average >= 70 && average <= 80) {
        printf("This student gets grade C\n");
    }
    else if (average >= 61 && average <= 69) {
        printf("This student gets grade D\n");
    }
    else if (average >= 50 && average <= 60) {
        printf("This student gets grade E\n");
    }
    else {
        printf("This student gets grade F\n");
    }
}

```

Input scores

- **printf("Math score: ");**
- **scanf("%f", &math);**
  - Displays a message asking for the **Math** score,
    then uses scanf() to store the user's input in the variable math.

Same thing happens for:

- **printf("English score: ");**
- **scanf("%f", &english);**

and

- **printf("Computer score: ");**
- **scanf("%f", &computer);**

✅ After these 3 steps, the program now has all the subject scores.

Calculate the average

- **average = (math + english + computer) / 3;**
  - This formula adds all three scores together, then divides by 3 to get the **average score**.
  - Example:
  - If math = 80, english = 70, computer = 90
    → average = (80 + 70 + 90) / 3 = 80.00

Show the average

- **printf("Average score is: %.2f\n", average);**
  - This prints the average score with **2 decimal places**.
  - Example output:
  - Average score is: 80.00
  - **Decide the grade using if and else if**

Now, the program checks which range the average score is in.

- **if (average >= 86 && average <= 100) {**
- **printf("This student gets grade A\n");**
- **}**
  - If the average is between 86 and 100 → print **grade A**.
- **else if (average >= 81 && average <= 85) {**
- **printf("This student gets grade B\n");**
- **}**
  - If average is between 81 and 85 → print **grade B**.
- else if (average >= 70 && average <= 80) {
- printf("This student gets grade C\n");
- }
  - If average is between 70 and 80 → print **grade C**.

- **else if (average >= 61 && average <= 69) {**
- **printf("This student gets grade D\n");**
- **}**
    - If average is between 61 and 69 → print **grade D**.

- **else if (average >= 50 && average <= 60) {**
- **printf("This student gets grade E\n");**
- **}**
    - If average is between 50 and 60 → print **grade E**.

- **else {**
- **printf("This student gets grade F\n");**
- **}**
    - If the average is below 50 → print **grade F** (fail).

Result:

