C/C++ PROGRAMMING (IT116IU)

Lab 4 - Introduction to C/C++ Programming

Your Name: Vũ Kiến Quốc
Your ID: ITITIU21295

Due date: Please check on Blackboard

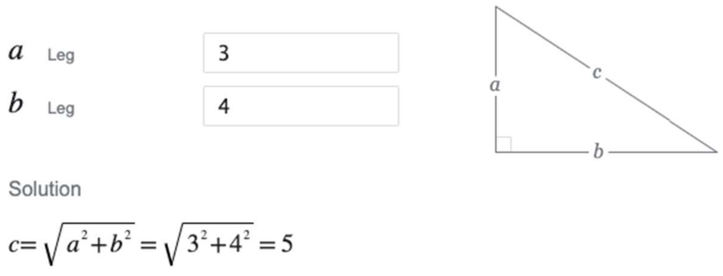
Instruction

Please follow the steps:

1. For each question, please make your code clean and make sure that your code is runnable.
2. Open the provided problem sets (.docx file). For each exercise, please capture screenshots of your work and then paste them into the problem sets (.docx file). DO NOT create a new answer file!
   Please convert this .docx file to .pdf file
3. Submit these files (source code and problem set files) to Blackboard before the deadline.
4. There are a total of 7 Lab Assignments in this course. 3/7 Lab Assignments will be randomly selected to score (~10% of your final score).
5. The final lab exam will be 10% of your final score.

Lab Assignments

Question 1. Define a function called hypotenuse that calculates the length of the hypotenuse of a right triangle when the other two sides are given. Use this function in a program to determine the length of the hypotenuse for each of the following triangles. The function should take two arguments of type double and return the hypotenuse as a double. Test your program with the side values specified in the following figure.



Solution

$$c = \sqrt{a^2 + b^2} = \sqrt{3^2 + 4^2} = 5$$

Triangle	Side 1	Side 2
1	3.0	4.0
2	5.0	12.0
3	8.0	15.0

E.g.:

Enter the sides of the triangle: 3.0 4.0

Hypotenuse: 5.0

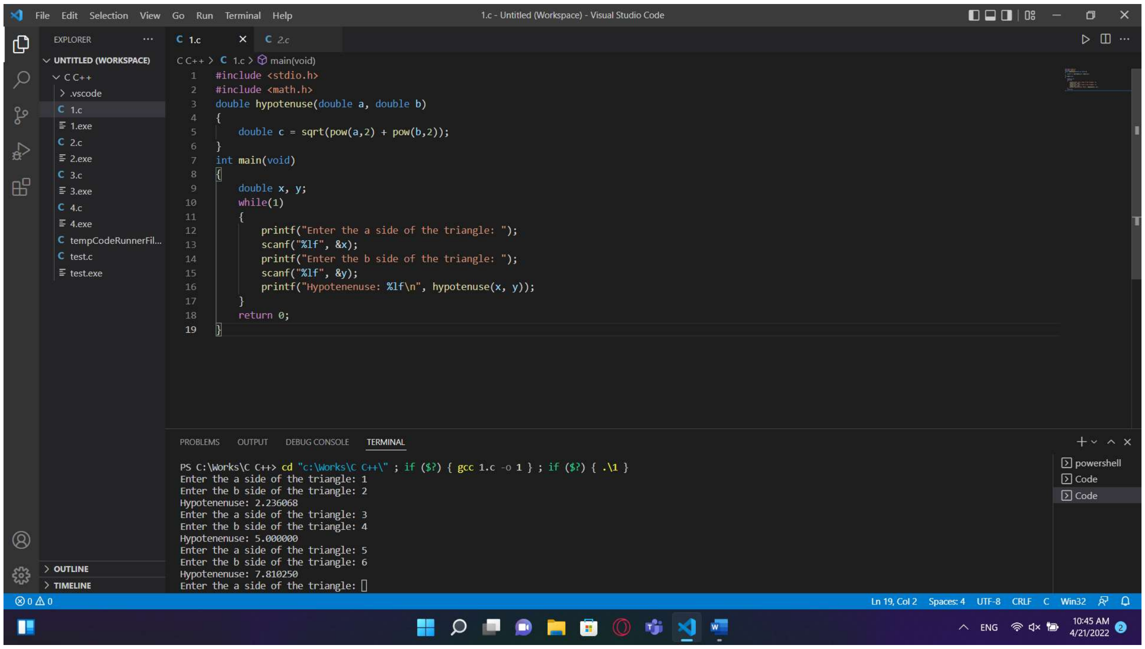
Enter the sides of the triangle: 5.0 12.0

Hypotenuse: 13.0

Enter the sides of the triangle: 8.0 15.0

Hypotenuse: 17.0

Please capture screenshots of your work then paste them here



```c
#include <stdio.h>
#include <math.h>
double hypotenuse(double a, double b)
{
    double c = sqrt(pow(a,2) + pow(b,2));
}
int main(void)
{
    double x, y;
    while(1)
    {
        printf("Enter the a side of the triangle: ");
        scanf("%lf", &x);
        printf("Enter the b side of the triangle: ");
        scanf("%lf", &y);
        printf("Hypotenenuse: %lf\n", hypotenuse(x, y));
    }
    return 0;
}
```

```
PS C:\Works\C C++> cd "c:\Works\C C++\" ; if ($?) { gcc 1.c -o 1 } ; if ($?) { .\1 }
Enter the a side of the triangle: 1
Enter the b side of the triangle: 2
Hypotenenuse: 2.236068
Enter the a side of the triangle: 3
Enter the b side of the triangle: 4
Hypotenenuse: 5.000000
Enter the a side of the triangle: 5
Enter the b side of the triangle: 6
Hypotenenuse: 7.810250
Enter the a side of the triangle: 
```

Question 2. Write a program that inputs a series of integers and passes them one at a time to function even which uses the remainder operator to determine if an integer is even. The function should take an integer argument and return 1 if the integer is even and 0 otherwise.

E.g.:

Enter an integer: 7

7 is not an even integer

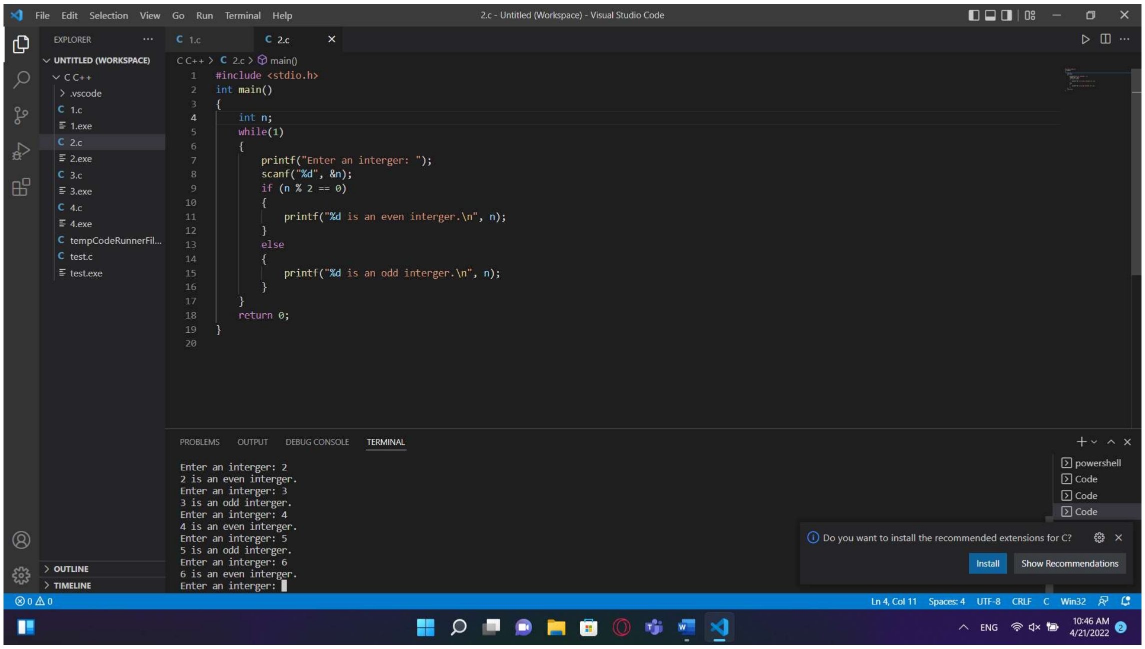
Enter an integer: 6

6 is an even integer

Enter an integer: 10000

10000 is an even integer

Please capture screenshots of your work then paste them here



```
#include <stdio.h>
int main()
{
    int n;
    while(1)
    {
        printf("Enter an interger: ");
        scanf("%d", &n);
        if (n % 2 == 0)
        {
            printf("%d is an even interger.\n", n);
        }
        else
        {
            printf("%d is an odd interger.\n", n);
        }
    }
    return 0;
}
```

```
Enter an interger: 2
2 is an even interger.
Enter an interger: 3
3 is an odd interger.
Enter an interger: 4
4 is an even interger.
Enter an interger: 5
5 is an odd interger.
Enter an interger: 6
6 is an even interger.
Enter an interger: 
```

Do you want to install the recommended extensions for C?

Question 3. Write a C program that plays the game of "guess the number" as follows: Your program chooses the number to be guessed by selecting an integer at random in the range 1 to 1000. The program then types:

```
I have a number between 1 and 1000.
Can you guess my number?
Please type your first guess.
```

The player then types a first guess. The program responds with one of the following:

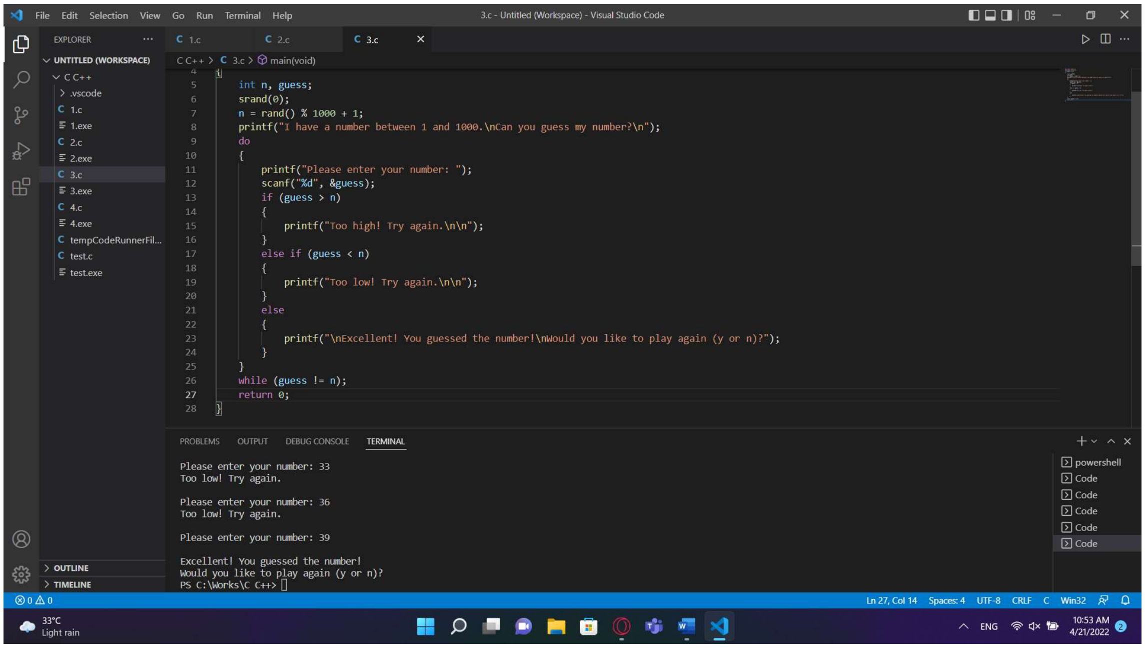
```
1. Excellent! You guessed the number!
   Would you like to play again (y or n)?
2. Too low. Try again.
3. Too high. Try again.
```

If the player's guess is incorrect, your program should loop until the player finally gets the number right. Your program should keep telling the player Too high or Too low to help the player “zero in” on the correct answer. [Note: The searching technique employed in this problem is called binary search.]

E.g.:

```
I have a number between 1 and 1000. Can you guess my number?
Please type your first guess.
? 500

Too low. Try again. ? 750
Too high. Try again. ? 625

Too low. Try again. ? 687
Too high. Try again. ? 656

Too low. Try again. ? 671
Too low. Try again. ? 678

Too high. Try again. ? 675
Too high. Try again. ? 673

Too high. Try again. ? 672

Excellent! You guessed the number! Would you like to play again? Please type ( 1=yes, 2=no )? 2
```

Please capture screenshots of your work then paste them here

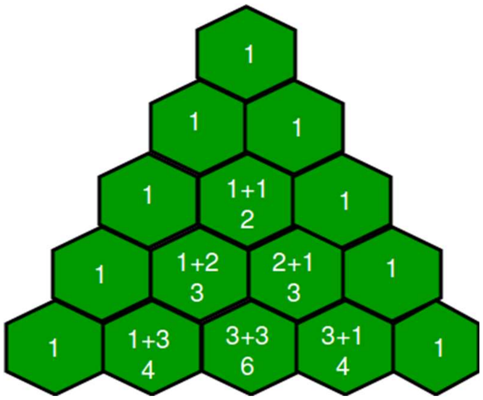


```c
{
    int n, guess;
    srand(0);
    n = rand() % 1000 + 1;
    printf("I have a number between 1 and 1000.\nCan you guess my number?\n");
    do
    {
        printf("Please enter your number: ");
        scanf("%d", &guess);
        if (guess > n)
        {
            printf("Too high! Try again.\n\n");
        }
        else if (guess < n)
        {
            printf("Too low! Try again.\n\n");
        }
        else
        {
            printf("\nExcellent! You guessed the number!\nWould you like to play again (y or n)?");
        }
    }
    while (guess != n);
    return 0;
}
```

```
Please enter your number: 33
Too low! Try again.

Please enter your number: 36
Too low! Try again.

Please enter your number: 39

Excellent! You guessed the number!
Would you like to play again (y or n)?
PS C:\Works\C C++> 
```

Question 4. Pascal’s triangle is a triangular array of the binomial coefficients. Write a function that takes an integer value n as input and prints first n lines of the Pascal’s triangle. Following are the first 6 rows of Pascal’s Triangle.

```
1

1  1

1  2  1

1  3  3  1

1  4  6  4  1

1  5  10  10  5  1
```



Number of entries in every line is equal to line number. For example, the first line has “1”, the second line has “1 1”, the third line has “1 2 1”,.. and so on. Every entry in a line is value of a Binomial Coefficient. The value of ith entry in line number line is C(line, i).

We know that **i**th entry in a line number line is Binomial Coefficient C(line, i) and all lines start with value 1. The idea is to calculate C(line, i) using C(line, i-1). It can be calculated in O(1) time using the following.

```
C(line, i)   = line! / ( (line-i)! * i! )

C(line, i-1) = line! / ( (line - i + 1)! * (i-1)! )

We can derive following expression from above two expressions.

C(line, i) = C(line, i-1) * (line - i + 1) / i
```

<u>E.g.</u>

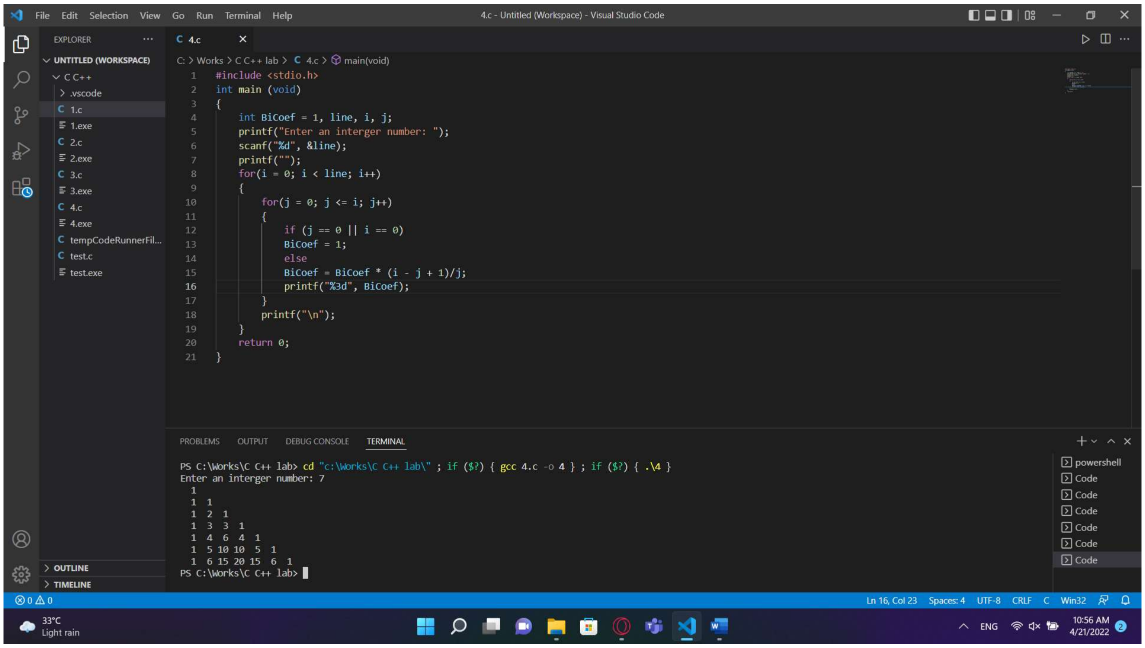
Input an integer number:

7

Output:

```
1

1  1

1  2  1

1  3  3  1

1  4  6  4  1

1  5  10  10  5  1

1  6  15  20  15  6  1
```

Please capture screenshots of your work then paste them here



```c
#include <stdio.h>
int main (void)
{
    int BiCoef = 1, line, i, j;
    printf("Enter an interger number: ");
    scanf("%d", &line);
    printf("");
    for(i = 0; i < line; i++)
    {
        for(j = 0; j <= i; j++)
        {
            if (j == 0 || i == 0)
            BiCoef = 1;
            else
            BiCoef = BiCoef * (i - j + 1)/j;
            printf("%3d", BiCoef);
        }
        printf("\n");
    }
    return 0;
}
```

```
PS C:\Works\C C++ lab> cd "c:\Works\C C++ lab\" ; if ($?) { gcc 4.c -o 4 } ; if ($?) { .\4 }
Enter an interger number: 7
  1
  1  1
  1  2  1
  1  3  3  1
  1  4  6  4  1
  1  5 10 10  5  1
  1  6 15 20 15  6  1
PS C:\Works\C C++ lab> 
```