C/C++ PROGRAMMING (IT116IU)

Lab 6 - Introduction to C/C++ Programming

Your Name: Vu Kien Quoc
Your ID: ITITIU21295

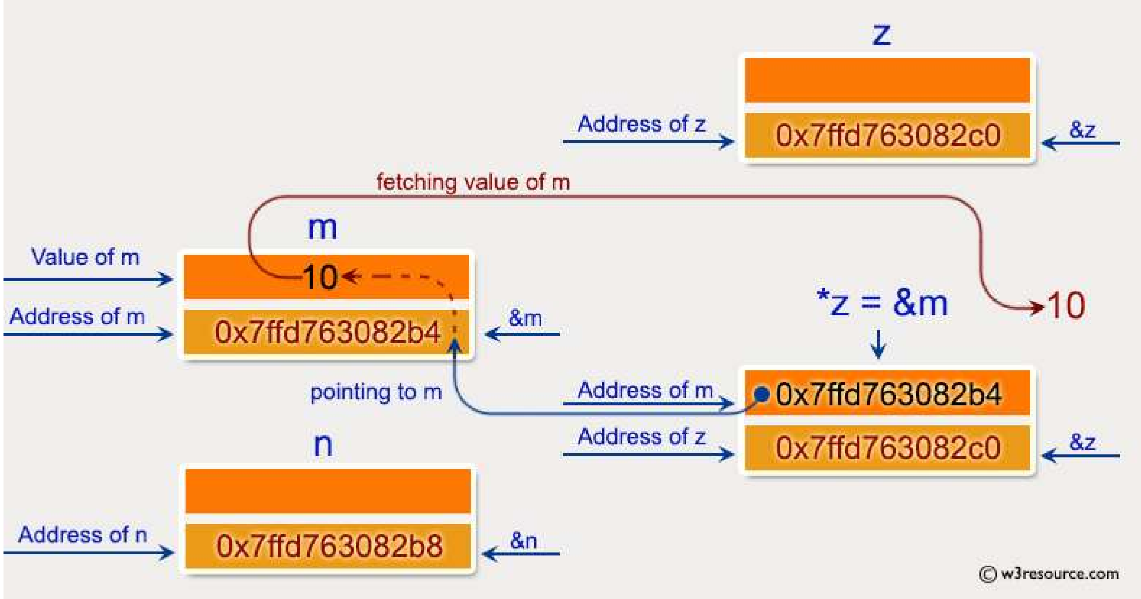
Due date: Please check on Blackboard

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.

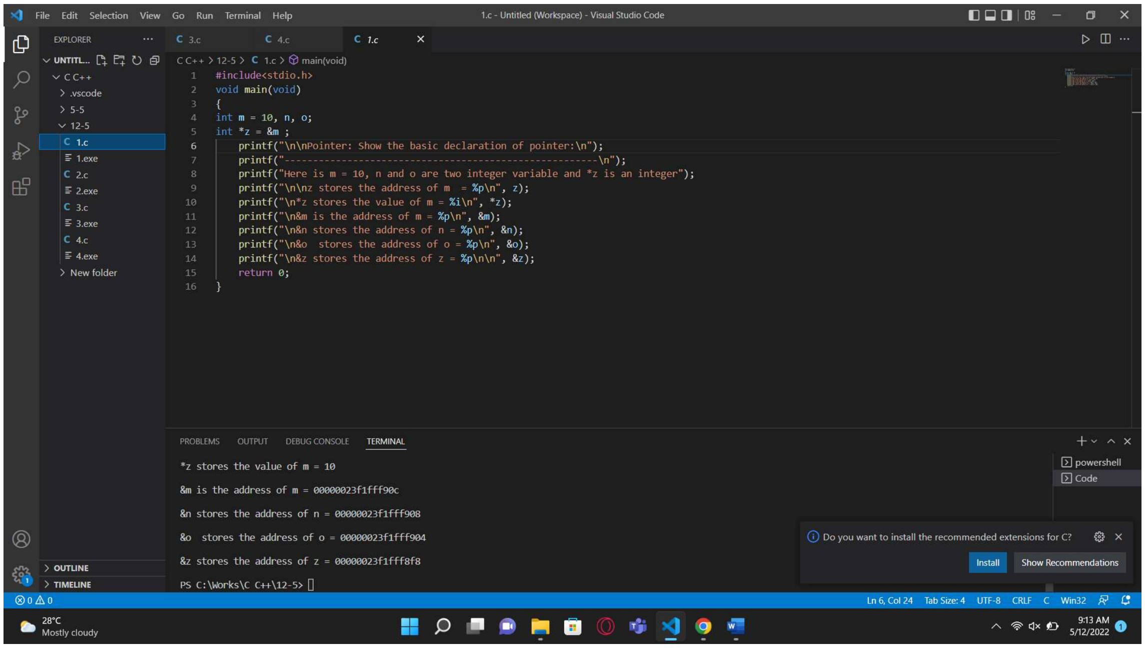
Lab Assignments

Question 1. Write a program in C to show the basic declaration of pointer.



Expected Output :

```
 Pointer : Show the basic declaration of pointer :
-------------------------------------------------------
 Here is m=10, n and o are two integer variable and *z is an
integer

 z stores the address of m  = 0x7ffd40630d44

 *z stores the value of m = 10

 &m is the address of m = 0x7ffd40630d44

 &n stores the address of n = 0x7ffd40630d48

 &o  stores the address of o = 0x7ffd40630d4c

 &z stores the address of z = 0x7ffd40630d50
```

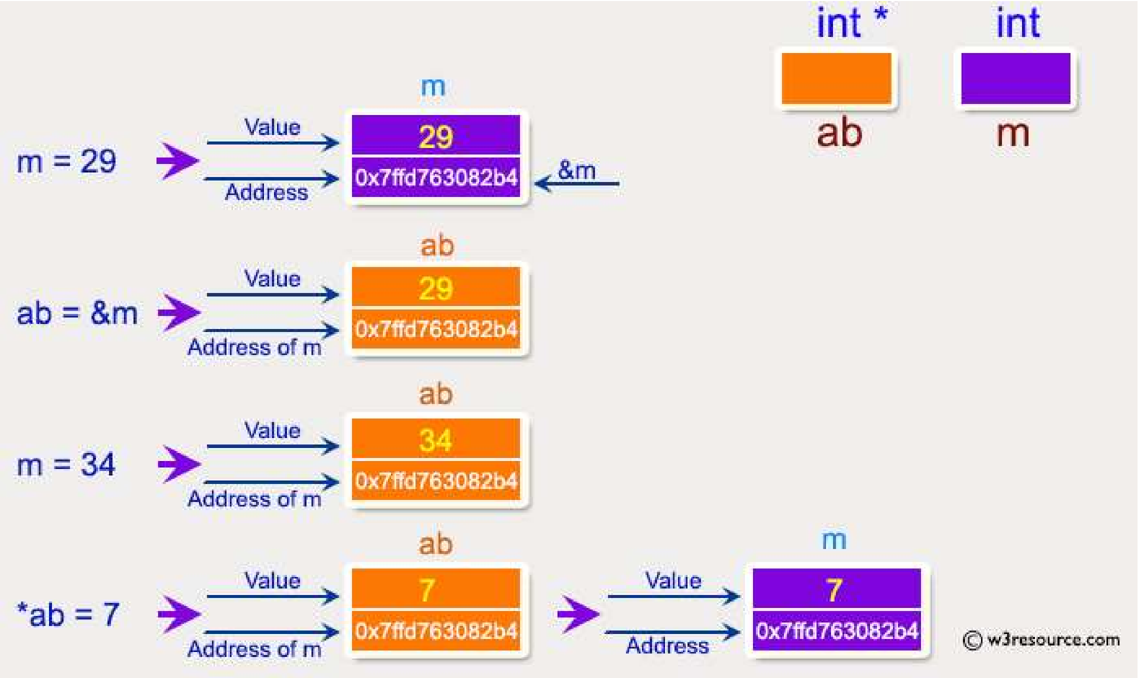


```c
#include<stdio.h>
void main(void)
{
int m = 10, n, o;
int *z = &m ;
    printf("\n\nPointer: Show the basic declaration of pointer:\n");
    printf("-------------------------------------------------------\n");
    printf("Here is m = 10, n and o are two integer variable and *z is an integer");
    printf("\n\nz stores the address of m  = %p\n", z);
    printf("\n*z stores the value of m = %i\n", *z);
    printf("\n&m is the address of m = %p\n", &m);
    printf("\n&n stores the address of n = %p\n", &n);
    printf("\n&o  stores the address of o = %p\n", &o);
    printf("\n&z stores the address of z = %p\n\n", &z);
    return 0;
}
```

```
*z stores the value of m = 10

&m is the address of m = 00000023f1fff90c

&n stores the address of n = 00000023f1fff908

&o  stores the address of o = 00000023f1fff904

&z stores the address of z = 00000023f1fff8f8

PS C:\Works\C C++\12-5> 
```

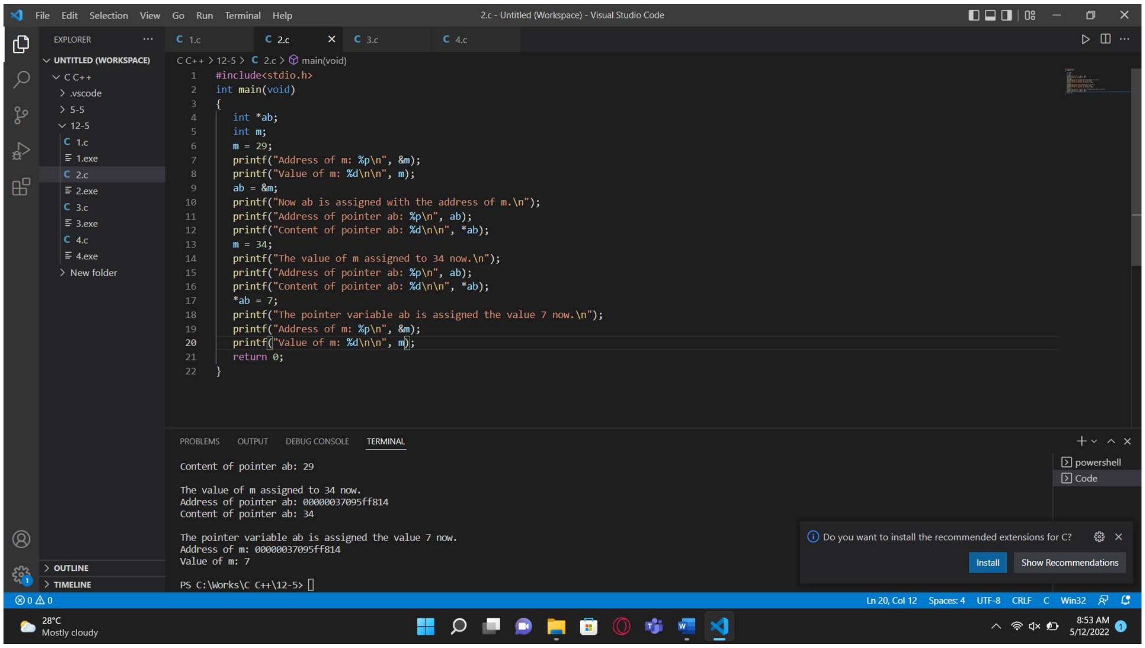
Do you want to install the recommended extensions for C?

Question 2. Write a program in C to demonstrate how to handle the pointers in the program.



Expected Output :

```
Address of m : 0x7ffcc3ad291c
 Value of m : 29

 Now ab is assigned with the address of m.
 Address of pointer ab : 0x7ffcc3ad291c
 Content of pointer ab : 29

 The value of m assigned to 34 now.
 Address of pointer ab : 0x7ffcc3ad291c
 Content of pointer ab : 34

 The pointer variable ab is assigned with the value 7 now.
 Address of m : 0x7ffcc3ad291c
 Value of m : 7
```

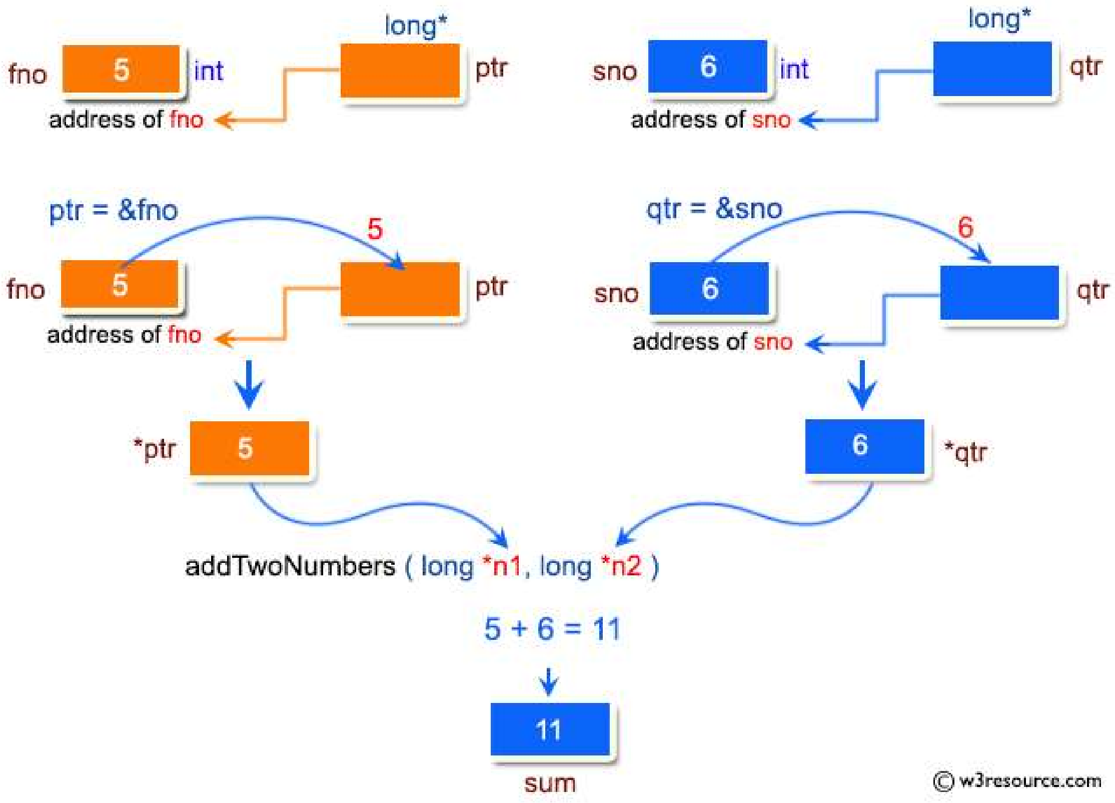


```c
#include<stdio.h>
int main(void)
{
    int *ab;
    int m;
    m = 29;
    printf("Address of m: %p\n", &m);
    printf("Value of m: %d\n\n", m);
    ab = &m;
    printf("Now ab is assigned with the address of m.\n");
    printf("Address of pointer ab: %p\n", ab);
    printf("Content of pointer ab: %d\n\n", *ab);
    m = 34;
    printf("The value of m assigned to 34 now.\n");
    printf("Address of pointer ab: %p\n", ab);
    printf("Content of pointer ab: %d\n\n", *ab);
    *ab = 7;
    printf("The pointer variable ab is assigned the value 7 now.\n");
    printf("Address of m: %p\n", &m);
    printf("Value of m: %d\n\n", m);
    return 0;
}
```

```
Content of pointer ab: 29

The value of m assigned to 34 now.
Address of pointer ab: 00000037095ff814
Content of pointer ab: 34

The pointer variable ab is assigned the value 7 now.
Address of m: 00000037095ff814
Value of m: 7

PS C:\Works\C C++\12-5> 
```

2.c - Untitled (Workspace) - Visual Studio Code
File Edit Selection View Go Run Terminal Help
1.c 2.c 3.c 4.c
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Do you want to install the recommended extensions for C?
Install
Show Recommendations

Question 3. Write a program in C to add numbers using call by reference.



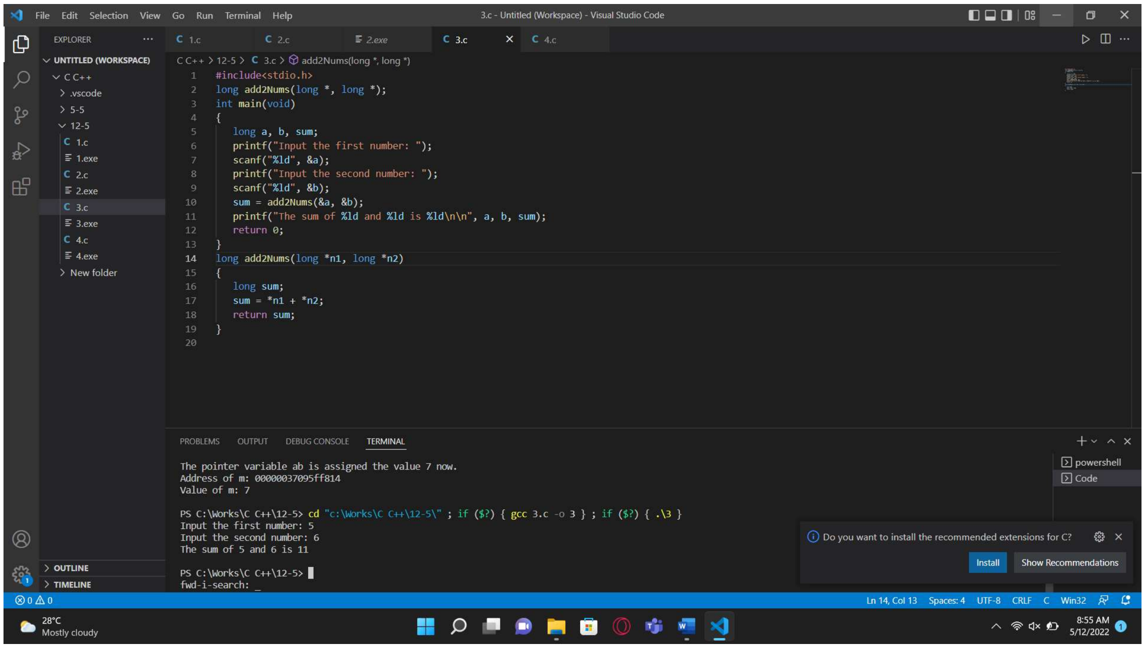
Test Data :

Input the first number : 5

Input the second number : 6

Expected Output :

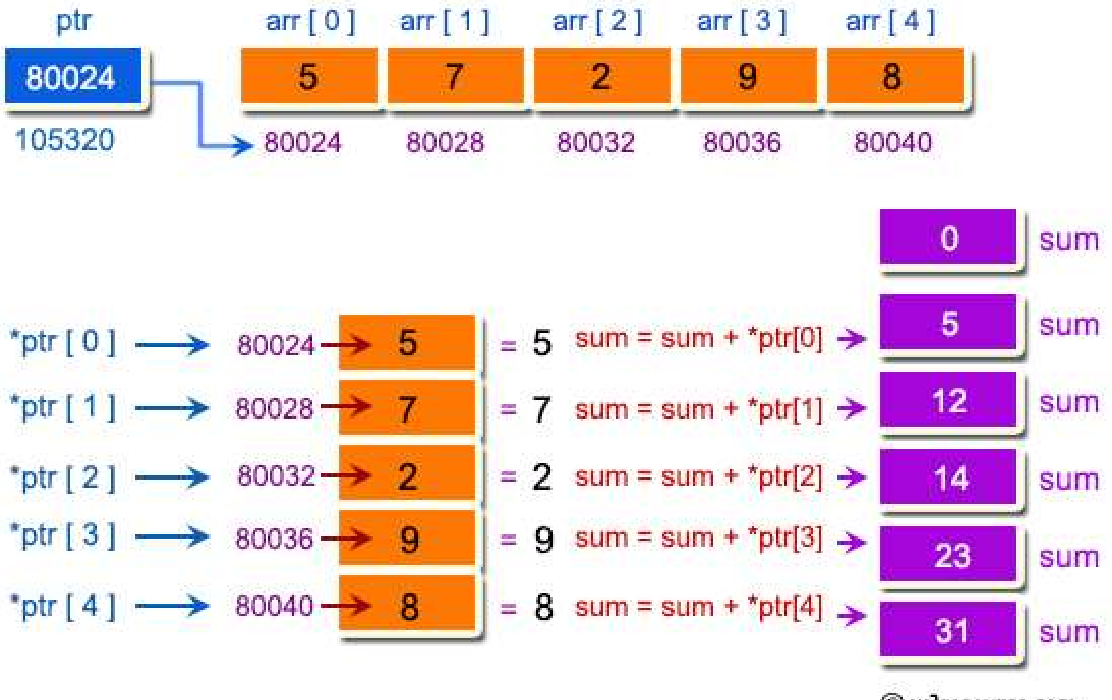
```
 The sum of 5 and 6  is 11
```



```c
#include<stdio.h>
long add2Nums(long *, long *);
int main(void)
{
    long a, b, sum;
    printf("Input the first number: ");
    scanf("%ld", &a);
    printf("Input the second number: ");
    scanf("%ld", &b);
    sum = add2Nums(&a, &b);
    printf("The sum of %ld and %ld is %ld\n\n", a, b, sum);
    return 0;
}
long add2Nums(long *n1, long *n2)
{
    long sum;
    sum = *n1 + *n2;
    return sum;
}
```

```
The pointer variable ab is assigned the value 7 now.
Address of m: 00000037095ff814
Value of m: 7

PS C:\Works\C C++\12-5> cd "c:\Works\C C++\12-5\" ; if ($?) { gcc 3.c -o 3 } ; if ($?) { .\3 }
Input the first number: 5
Input the second number: 6
The sum of 5 and 6 is 11

PS C:\Works\C C++\12-5> 
fwd-i-search: _
```

Question 4. Write a program in C to compute the sum of all elements in an array using pointers



ptr
80024
105320
arr [0]
arr [1]
arr [2]
arr [3]
arr [4]
5
7
2
9
8
80024
80028
80032
80036
80040
0
sum
*ptr [0] → 80024 → 5 = 5 sum = sum + *ptr[0] → 5 sum
*ptr [1] → 80028 → 7 = 7 sum = sum + *ptr[1] → 12 sum
*ptr [2] → 80032 → 2 = 2 sum = sum + *ptr[2] → 14 sum
*ptr [3] → 80036 → 9 = 9 sum = sum + *ptr[3] → 23 sum
*ptr [4] → 80040 → 8 = 8 sum = sum + *ptr[4] → 31 sum

© w3resource.com

Test Data :

Input the number of elements to store in the array (max 10) : 5

Input 5 number of elements in the array :

element - 1 : 2

element - 2 : 3

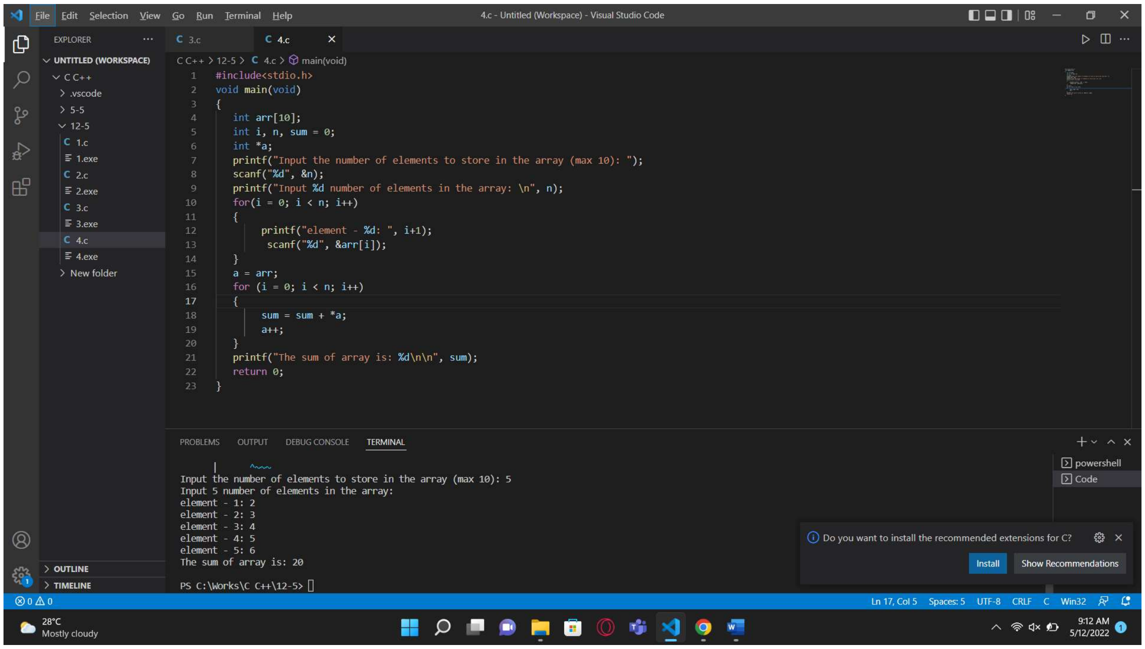
element - 3 : 4

element - 4 : 5

element - 5 : 6

Expected Output :

```
The sum of array is : 20
```



```
#include<stdio.h>
void main(void)
{
   int arr[10];
   int i, n, sum = 0;
   int *a;
   printf("Input the number of elements to store in the array (max 10): ");
   scanf("%d", &n);
   printf("Input %d number of elements in the array: \n", n);
   for(i = 0; i < n; i++)
   {
        printf("element - %d: ", i+1);
         scanf("%d", &arr[i]);
   }
   a = arr;
   for (i = 0; i < n; i++)
   {
        sum = sum + *a;
        a++;
   }
   printf("The sum of array is: %d\n\n", sum);
   return 0;
}
```

```
Input the number of elements to store in the array (max 10): 5
Input 5 number of elements in the array:
element - 1: 2
element - 2: 3
element - 3: 4
element - 4: 5
element - 5: 6
The sum of array is: 20

PS C:\Works\C C++\12-5> 
```

Do you want to install the recommended extensions for C?