Assignment 4.3	
Pointers	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed:9/22/25**
Section:CPE11S1	**Date Submitted:9/22/25**
Name(s): Niel Vincent B. Condino	**Instructor: Engr. Jimlord M. Quejado**

6. Output

What is a pointer in C++?
A pointer is a variable that stores the memory address of another variable as its value.
How does a pointer differ from a regular variable?
A regular variable stores a direct data value, such as an integer or string, while a pointer is a special kind of variable that stores the memory address of another variable.
What operator is used to get the address of a variable?
The "&" operator is used to get an address of a variable.
What operator is used to access the value stored at a pointer's address?
The "*" operator is used to access the value stored at a pointer's address.
Why are pointers important in C++? Give two uses.
It is useful when storing and getting the memory addresses and values of a variable. It is also useful managing memory in case of big data storage.

7. Supplementary Activity

Identify the Output

For each code snippet, predict the output without compiling:

1.

```
int x = 42;

int *ptr = &x;

cout << *ptr;
```

Output : 42

2.

```
int a = 5, b = 10;

int *p = &a;

p = &b;

cout << *p;
```

Output: 10

3.

```
int arr[3] = {10, 20, 30};

int *p = arr;

cout << *p;
```

Output: 10

4.

```
int arr[4] = {2, 4, 6, 8};

int *p = arr;

p++;

cout << *p;
```

Output : 4

5.

```
int arr[3] = {5, 15, 25};

int *p = arr;

cout << *(p + 2);
```

Output:25

Error Spotting

Identify and fix the error(if any) in the codes below.

1.

```
int arr[3] = {1, 2, 3};

int *p = &arr;
```

Error: second line, the pointer stores the address of the whole array instead of an element of it. Can be fixed by replacing "&arr" with "&arr[0]" or just "arr".

Fixed:

```
int arr[3] = {1, 2, 3};

int *p = &arr[0];
```

2.

```
int arr[5];

int *p;

p = arr[2];
```

Error: The pointer does not store the address of arr[2] but the value itself. Can be fixed by adding the "&" operator before the "arr[2]"
Fixed:

```
int arr[5];

int *p;

p = &arr[2];
```

3.

```
int arr[4] = {10, 20, 30, 40};

cout << *arr[2];
```

Error: arr[2] 's value is already an integer so applying the "*" operator causes an error*. *Initializing a pointer or printing it directly will solve this error.*

Fixed:

```
int arr[4] = {10, 20, 30, 40};

int *p = &arr[2];

cout << *p;
```

8. Conclusion

The output part made me learn more by searching about them on other sites. The different questions made me remember more how a pointer works,the difference of it from a normal variable, its operators, and their purpose. The last question made me search deeper about the topic because I only know one of its purposes and that is the first answer I put there. However, after searching I have many more purposes for pointers and picked one of them and put it as one of my answers.The supplementary on the other hand is pretty useful for training how to use and debug on the topic of pointers. The predicting output part made me search on youtube on how it works so I can technically predict correctly without compiling it. From it, it also helped on the debugging part of the activity. I learned the proper syntax for pointers so that I can spot the errors within the snippet and how to fix them. Overall, I think I did pretty alright for this activity. I learned more about how to apply pointers in c++. Although I still need to learn more about the topic because I did not get the answers without sometimes searching more about pointers andI feel there is still more to learn about pointers.

9. Assessment Rubric