| Activity No. 4.2 | |
|---|---|
| Pointers | |
| **Course Code:** CPE 007 | **Program:** Computer Engineering |
| **Course Title:** Programming Logic and Design | **Date Performed: 9/18/2025** |
| **Section: CPE11S1** | **Date Submitted: 9/18/2025** |
| **Name(s): Cenar, Marqui Joshua** | **Instructor: Engr. Jimlord M. Quejado** |
| **6. Output** | |

```cpp
#include <iostream>
using namespace std;

int main() {
    const int size = 10;
    int scores[size] = {95, 85, 78, 88, 92, 80, 75, 80, 89, 91};

    for (int i = 0; i < size; i++) {
        cout << scores[i] << " ";
    }

    cout << endl << endl;
    for (int i = 0; i < size; i++)
        cout << "address of element " << i << ": " << &scores[i] << endl;

    cout << endl << endl;
    int *scorePtr;
    scorePtr = &scores[0];

    cout << "the address of the array[0]: " << scorePtr << endl;
    cout << "the dereferenced pointer: " << *scorePtr << endl;
    cout << endl << endl;

    int numBytes = sizeof(scores);
    cout << "The number of bytes of the array is: " << numBytes << endl;

    return 0;
}
```

```
95 85 78 88 92 80 75 80 89 91

address of element 0: 0x7ff7b72989c0
address of element 1: 0x7ff7b72989c4
address of element 2: 0x7ff7b72989c8
address of element 3: 0x7ff7b72989cc
address of element 4: 0x7ff7b72989d0
address of element 5: 0x7ff7b72989d4
address of element 6: 0x7ff7b72989d8
address of element 7: 0x7ff7b72989dc
address of element 8: 0x7ff7b72989e0
address of element 9: 0x7ff7b72989e4



the address of the array[0]: 0x7ff7b72989c0
the dereferenced pointer: 95



The number of bytes of the array is: 40


Process finished with exit code 0
```

## 7. Supplementary Activity

- I learned from this activity how to store elements sequentially and how the pointers can be used to access both the values and addresses in an array.
  On line 5 and 6 I initialized an array called "scores" and its size "score[size]" with a value of 10.
  Line 8-9 I created a for loop to print out or display all the elements of the array one by one.
  Line 13-14 I created another for loop to print out or display the memory address of each array element using "&scores[i]".
  Line 17-18 I used a pointer "scorePtr" set to point the first element of the array (&scores[0]).
  Line 20-22 The program prints out or displays the address stored in the pointer, which is the address of the first element and also prints out the value stored at that address using the dereference operator "*scorePtr".
  Line 24-25 I initialized numBytes = sizeof(scores) to calculate the total number of bytes taken by the whole array in memory and it prints out "The number of bytes of the array is: ".

## 8. Conclusion

- For my first time figuring out how to use pointers in an array, I think I learned it smoothly but through trial and error. While doing this activity I figured out how arrays and pointers work together inside of a program and was able to accomplish the required output. With more practice I could possibly create a more clear explanation and program about this topic.