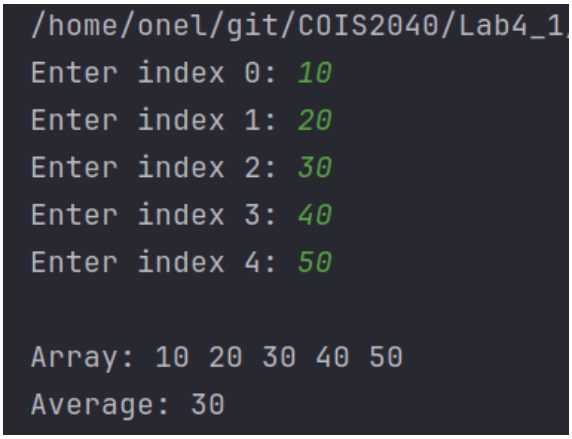
2040 Lab 4 – Answer Document

Part 1 – Arrays

3. Run your program, and show the output.



5. Show the output and your function call.

```cpp
using namespace std;

double CalculateAverage(double numberArray[], int arraySize);

int main()
{
    double numbers[5];

    for (int i = 0; i < 5; i++)
    {
        cout << "Enter index " << i << ": ";
        cin >> numbers[i];
    }
    cout << "\nArray: ";
    for (double val : numbers)
    {
        cout << val << " ";
    }
    const double average = CalculateAverage(numbers, arraySize: 5);
    cout << "\nAverage: " << average << endl;
}

double CalculateAverage(double numberArray[], int arraySize)
{
    double sum = 0;
    for (int i = 0; i < arraySize; i++)
    {
        sum += numberArray[i];
    }
    return sum / arraySize;
}
```

Part 2 – C Strings (strings as arrays)

2. What changes? Why does this happen?

We are receiving a letter-assigned character array, with a character at the specified index.

3. Show the output when changing the value in the brackets to:

- **str1[2] :** l
- **str1[-1] :** w
- **str1[10] :** (Empty)

4. Capture the output

Why does s2 not output what's expected? How can you fix this?

Since char array s2 is not terminated with '\0' will keep printing until finds '\0' in memory

5. Output the new string, and capture your output

```
int main()
{
    char s1[] = {'C', 'O', 'I', 'S', '\0'};
    char s2[] = {'2', '0', '4', '0', '\0'};

    cout << "s1 = " << s1 << endl;
    cout << "s2 = " << s2 << endl;

    char combined[] = "";
    strcpy(combined, s1);
    strcat(combined, s2);

    cout << "combined = " << combined << endl;
}
```

```
in
Lab4_2 ×
/home/onel/git/COIS2040/Lab4_2/cmake-build-debug/Lab4_2
s1 = COIS
s2 = 2040
combined = COIS2040

Process finished with exit code 0
```

6. Why? How do you fix it?

The word 'Cats' contains 4 characters plus a null-terminator. Simply making the array size 5 will prevent errors.

Part 3 – The string class

2. What is the difference between the two inputs?

`cin >>` reads a string until the first whitespace, while `getline(cin, val)` captures the entire input line.

3. What is the difference? What did the commented line do?

Without clearing the input buffer, the leftover newline from the first input is used by std::getline

4. Capture your output.

```cpp
int main()
{
    string s4, s5;

    cout << "Enter value for s4: " << endl;
    getline([&]cin, [&]s4);

    cout << "Enter value for s5: " << endl;
    getline([&]cin, [&]s5);

    string s6 = s4 + s5;
    cout << "Combined s6: " << s6 << endl;
    cout << "Length: " << s6.length() << endl;
}
```

```
Lab4_2 ×
⋮
/home/onel/git/COIS2040/Lab4_2/cmake-build-debug/Lab4_2
Enter value for s4:
onur
Enter value for s5:
onel
Combined s6: onuronel
Length: 8
```