Lesson 5 Lab

Task 1: define a **program** which defines an int array with length 10, gives it some initial values, and then prints out the values from the array along with each value's index

```
#include <stdio.h>

int main()
{
    int a[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    for (int x = 0; x < 10; x++)
    {
        printf("Value: %d Index: %d \n", a[x], x);
    }
    return 0;
}
```

Task 2: Write a program which reads in a string name, Eg, "Tom", and then prints out a greeting "Hello Tom!"

```
#include <stdio.h>

int main()
{
    char name[100];
    scanf("%s", name);
    printf("Hello %s!", name);
    return 0;
}
```

Task 3: Write a program which reads in a string from the user, and then prints a version without the first and last char, so for "Hello" prints "ell".
The string length will be at least 2.

input: Hello → print: ell
input: salute → print: alut
input: coding → print: odin

```
#include <stdio.h>
```

```
int main()
{
   char name[100] = "";
   scanf("%s", name);

   int length = 0;
   while (name[length] != '\0') length++;

   for (int i = 1; i < length - 1; i++)
   {
      printf("%c", name[i]);
   }
   return 0;
}
```

Task 4: Write a **function** (named "nonStart") which takes in two char arrays, and then prints their concatenation, except omit the first char of each. The strings will be at least length 1. Eg,

nonStart("Hello", "There") → print: "ellohere"
nonStart("java", "code") → print: "avaode"
nonStart("shotl", "java") → print:"hotlava"

```
void nonStart(char first[], char second[])
{
    int firstLength = 0;
    while (first[firstLength] != '\0') firstLength++;

    int secondLength = 0;
    while (first[secondLength] != '\0') secondLength++;

    for (int x = 1; x < firstLength; x ++)
    {
        printf("%c", first[x]);
    }
    for (int y = 1; y < secondLength; y ++)
```

```
    {
        printf("%c", second[y]);
    }
}
```

Task 5: Write a **function** which takes in an input integer array and the length of the array. The function is to reverse the array values. Eg, input [1, 2, 3] -> [3, 2,1]
[Hint: This function should not print, it changes values in the array.]

```
void reverse(int numbers[], size_t length)
{
    for (int x = 0; x < length / 2; x++) {
        int tmp = numbers[x];
        numbers[x] = numbers[length - x - 1];
        numbers[length - x - 1] = tmp;
    }
}
```