# Green University of Bangladesh
# Department of Computer Science and Engineering (CSE)
### Faculty of Sciences and Engineering
### Semester: (Summer, Year:2022), B.Sc. in CSE (Day)

### LAB REPORT NO 01
### Course Title: Data Structure Lab
### Course Code: CSE 106          Section: DA

**Lab Experiment Name:**

Search, replace, remove and insert elements to an existing array.

## Student Details

| Name | ID |
|---|---|
| Obaydur Rahman | 213902018 |

**Lab Date**                    : 15-06-2022
**Submission Date**             : 21-06-2022
**Course Teacher's Name**       : Farhana Akter Sunny

## Lab Report Status
**Marks:** ……………………………          **Signature:** ....................
**Comments:** ............................................          **Date:**...............................

## 1. TITLE OF THE LAB EXPERIMENT

Search, replace, remove and insert elements to an existing array.

## 2. AIM

1. Add an element to any index without losing the old value, and shift the array 1 step left from after the index. (Similar to array push, but we can do it on any index).
2. Remove an element by value or index, and shift the array 1 step right from after the removed index.
3. Search specific element and display it.
4. Replace value of an index with new value.

## 3. DESIGN

We will define and allocate memory for the array with one more element than we need, and after that we will have an extra space to insert new entry and shift other entries.

## 4. TEST RESULT / OUTPUT

```
opu@opu:/mnt/z/university/cse-106$ gcc main.c -o main && ./main
Choose an option:
1. Insert
2. Replace
3. Remove by index
4. Remove by value
5. Search

Enter your option: 1

Enter the index: 5
Enter the value: 12
Old: [ 0 ⇒ 0, 1 ⇒ 1, 2 ⇒ 2, 3 ⇒ 3, 4 ⇒ 4, 5 ⇒ 5, 6 ⇒ 6, 7 ⇒ 7, 8 ⇒ 8, 9 ⇒ 9 ]
New: [ 0 ⇒ 0, 1 ⇒ 1, 2 ⇒ 2, 3 ⇒ 3, 4 ⇒ 4, 5 ⇒ 12, 6 ⇒ 5, 7 ⇒ 6, 8 ⇒ 7, 9 ⇒ 8, 10 ⇒ 9 ]
opu@opu:/mnt/z/university/cse-106$
```

## 5. ANALYSIS AND DISCUSSION

With this process we can manipulate an array. This method is fast and easy but as we have allocated memory for only one extra entry, we won't be able to add more than one value to the array. We could allocate more memory but it is not ideal solution. Because we might not use that much memory or we might need more memory. What we could have done is, for each addition we could have copied the old array and created a new array with the new value inserted at its place and freed the old memory. In that way we could have used the actual amount of memory we need.