

APS106: FUNDAMENTALS OF COMPUTER PROGRAMMING

LAB # 7 - MONDAY, MARCH 10, 2:00 – 4:00

This lab will test your ability to declare arrays, initialize arrays, and compute arrays using function calls. The extension of your source file must be .c, not .cpp (e.g. lab7.c).

BACKGROUND: Consider a 1D array of numbers; for example:

```
int array[10] = {1,2,3,4,5,6,7,8,9,10};
```

Now consider a function `Shift` that shifts all elements of the array by some value and moves elements that fall off the end of the array onto the beginning. For example, `Shift(array,3)` would shift all elements three positions to the right (and move the last three elements to the beginning):

```
{8,9,10,1,2,3,4,5,6,7}
```

A subsequent `Shift(array,2)` would shift all elements a further two positions to the right:

```
{6,7,8,9,10,1,2,3,4,5}
```

PROBLEM: Write a function `main()` that allocates an `array[SIZE]`, and initializes the array as:

```
for (i=0; i<SIZE; i++)  
    array[i] = i+1;
```

Then write two other functions:

```
void Shift(int array[], int num); /* shift the array num positions*/  
void printArray(int array[]);    /* print the array */
```

From `main()`, call `Shift` and `printArray` from within a loop, and show that five shifts of two elements each results in the array returning to its original configuration:

```
1 2 3 4 5 6 7 8 9 10  
9 10 1 2 3 4 5 6 7 8  
7 8 9 10 1 2 3 4 5 6  
5 6 7 8 9 10 1 2 3 4  
3 4 5 6 7 8 9 10 1 2  
1 2 3 4 5 6 7 8 9 10
```

HINT: In the function `Shift`, allocate a second temporary array into which you can shift values, and once done, copy the elements of the temporary array back into the original one.