

# Lab #3

CS-2050

September 16, 2022

## 1 Requirements

In this lab, you will cover pointer arithmetic by offsetting and accessing via that offset an integer array. You might find it useful to re-use your Lab 2 code (or the solution), as some of the functions in this lab are similar to the previous lab. **Make sure to read the requirements carefully.**

### 1.1 makeArray

```
int * makeArray(int size);
```



**Info:** This function will take an integer representing the size of array to allocate. It will then allocate the array, and if successful, initialize it with a random number from 0 to 9 (inclusive) in each index. If the array was successfully allocated the function will return a pointer to it, otherwise the function will return NULL.

### 1.2 offsetArray

```
int * offsetArray(int *array, int offset);
```



**Info:** This function will take an integer array, and an offset to apply to the array. It will return a pointer to the array which is offset by the given amount, such that `array[offset]` is the first index in the array.

### 1.3 printNines

```
void printNines(int *array, int size, int offset);
```



**Info:** This function will take an integer array, and the offset of the array. It will print the index of all indices which are occupied by the number 9. (IE: in the array { 9, 0, 7, 9 } with no offset, it would print "0, 3".

### 1.4 freeArray

```
void freeArray(int *array, int offset);
```



**Info:** This function will take a pointer to an integer array, as well as the offset of the array pointer. It will free the memory allocated to the array.

## Notice



### Grading: 7 points

1. Write required *makeArray* function
  - \* 2 points
2. Write required *offset* function
  - \* 1 point
3. Write required *print* function
  - \* 2 points
4. Write required *free* function
  - \* 2 points



### Notice:

1. All of your lab submissions must compile under GCC using the *-Wall* and *-Werror* flags to be considered for a grade.
2. You are expected to provide proper documentation in every lab submission, in the form of code comments. For an example of proper lab documentation and a clear description of our expectations, see the lab policy document.