Assignment One - Mahnoor Shahid

How to Run the Code:

1. First, create a header file: Make sure that the functions readFile, findInteger, modifyInteger, addInteger, and removeIntegerOrSetZero are declared and defined in functions.h and their implementation file (functions.cpp).

2. Next, prepare the input file: Create a text file named input.txt. This file should contain the integers to be read into the array.

3. Then, compile the code: Use a C++ compiler to compile the main program and the implementation file containing the function definitions. After that, run the program.

4. Finally, the program will execute. The program will prompt you to enter a number to search for, and it will display whether the number was found in the array. It will then ask for an index and a new value to modify the array. You will be prompted to add a new integer to the array. Finally, the program asks you to either remove an element from the array or set it to 0 at a specified index.

ScreenShots - Question One

1. You can tell the program to look for a number, and it will find it. However, if you give it a number that isn't in the list, it will say that the number is not found.

   Number Found:

   ```
   Enter a number to find: 98
   Number 98 found at index 97.
   ```

   Number Not Found:

   ```
   Enter a number to find: 103
   Number 103 not found in the array.
   ```

2. You can modify the numbers. If you select an index to update and provide a new value, it will replace the old value at that index with the new one. It will display both the old value and the new value.

   ```
   Enter the index to modify: 23
   Enter the new value: 90
   Old value at index 23 was 24. The new value is 90.
   ```

3. You can add an integer to the end of the array. Simply input a number and it will be added. This also increases the size and shows the new size of the array.

```
Enter a new number to add to the array: 134
Added new number 134 to the array. The new size is now 101.
```

4. You can either remove the index or set it to 0. First, enter the index number. Then you can either enter 1 or 0. If you enter 1, it will remove that index and the size will decrease. If you enter 0, it will set that index to 0, and the size will stay the same.

   Input 1

```
Enter the index to remove or set to 0: 45
Enter 1 to remove or 0 to set to 0: 1
After removing, the array size is now 100.
```

   Input 0

```
Enter the index to remove or set to 0: 45
Enter 1 to remove or 0 to set to 0: 0
After setting to 0, the array size is 101.
```

Screenshots - Question 2

1. After adding error handling to the updated modify integer function, you must enter a valid index that is within the allowed range. If the index is not valid, the program will continue to prompt the user until a correct index is provided. After that, the program will request a new value. If the value entered is not an integer, the program will keep asking for a valid integer.

```
Enter the index to modify (0 to 99): 101
Invalid index. Please enter a valid index between 0 and 99.
Enter the index to modify (0 to 99): 99
Enter the new value: k
Invalid input. Please enter a valid integer.
Enter the new value: 132
Old value at index 99 was 100. The new value is 132.
```

2. After adding error handling to the function that adds an integer to the end of the array, you must enter a valid integer. If the user inputs anything other than an integer, it will keep prompting them. Additionally, the new value must be unique and cannot already be in the array.

```
Enter a new number to add to the array: m
Invalid input. Please enter a valid integer.
Enter a new number to add to the array: 98
The number 98 already exists in the array. Please enter a different number.
Enter a new number to add to the array: 105
Added new number 105 to the array. The new size is now 101.
```