

# Assignment 03

## Object Oriented Programming

### Spring 2023

**Q1: Operator Overloading for String Class** Your goal is to overload the operators for “String” class that you have implemented in the last assignment. You will need to write three files (string.h, string.cpp and stringMain.cpp). Your implemented class must fully provide the definitions of following class (interface) functions . Please also write down the test code to drive your class implementation. Please note that we will be running your code against our test code and any segmentation faults or incorrect result will result in loss of marks.

---

```
1 class String{
2 // think about the private data members...
3 public:
4 // provide definitions of following functions...
5 String(); // default constructor
6 String(char *str); // initializes the string with constant cstring
7 String(const String &); // copy constructor to initialize the string from existing
  ↳ string
8 String(int x); // initializes a string of pre-defined size
9 char operator[](int i); // returns the character at index [x]
10 String& operator+(String str); // append a String at the end of string
11 String& operator-(String substr); // removes the substr from the string
12 bool operator!(); // returns true if string is empty..
13 String& operator=(const String&); // Copy one string to another ...
14 bool operator==(const String&); // returns true if two strings are equal
15 int operator()(char); // returns the index of character being searched.
16 String operator*(int a); // multiplies the string by i times and return the string.
  ↳ Remember the Python functionality for *
17 int length(); // returns the length of string
18 ~String(); // destructor...
19 };
20 istream& operator<<(istream& input, const String); // Inputs the string
21 ostream& operator>>(ostream& output, const String); // Outputs the string
```

---

**Q2: Implementation of Array Class** Your goal is to overload the operators for “Array” class that you have implemented in the last assignment. You will need to write three files (array.h, array.cpp and arrayMain.cpp). Your implemented class must fully provide the definitions of following class (interface) functions . Please also write down the test code to drive your class implementation. Please note that we will be running your code against our test code and any segmentation faults or incorrect result will result in loss of marks.

```
1  class Array{
2  // think about the private data members...
3  public:
4  // provide definitions of following functions...
5  Array();// a default constructor
6  Array(int size);// a parametrized constructor initializing an Array of predefined
   → size
7  Array(int *arr, int size);// initializes the Array with an existing Array
8  Array(const Array &);// copy constructor
9  int operator[](int i);// returns the integer at index [i] after checking the out of
   → range error
10 void operator=(const Array);//copy the array
11 Array operator+(const Array);//adds two Array
12 Array operator-(const Array);//subtracts two Array
13 void operator++();//adds one to each element of Array
14 Array& operator--(int);//subtracts one from each element of array
15 bool operator==(const Array);//returns true if two arrays are same
16 bool operator!();// returns true if the Array is empty
17 void operator+=(const Array&);//adds two Array
18 void operator-=(const Array&);//subtracts two Array
19 int operator()(int idx, int val);// erases the value val at idx. Returns 1 for a
   → successful deletion and -1 if idx does not exists or is invalid. Shift the
   → elements after idx to the left.
20 ~Array();// destructor...
21 };
22 istream& operator<<(istream& input, const Array);//Inputs the Array
23 ostream& operator>>(ostream& ouput, const Array); //Outputs the Array
```

**Q3: Implementation of Matrix Class** Your goal is to overload the operators for “Matrix” class that you have implemented in the last assignment. You will need to write three files (matrix.h, matrix.cpp and matrixMain.cpp). Your implemented class must fully provide the definitions of following class (interface) functions . Please also write down the test code to drive your class implementation. Please note that we will be running your code against our test code and any segmentation faults or incorrect result will result in loss of marks.

---

```
1 class Matrix{
2 // think about the private data members...
3 // the matrix should store real numbers
4 public:
5 //include all the necessary checks before performing the operations in the functions
6 Matrix();// a default constructor
7 Matrix(int, int);// a parametrized constructor
8 Matrix(const Matrix &);// copy constructor
9 void operator()(int i, int j, float val);//set value at (i,j)
10 Matrix& operator=(const Matrix &);//assigns (copies) a Matrix. Returns the same
11 bool operator==(const Matrix &);//Compares two matrices
12 Matrix operator+(const Matrix &);//adds two Matrices and returns the result
13 Matrix operator-(const Matrix &);//subtracts two Matrices and returns the result
14 Matrix operator*(const Matrix &);//multiplies two Matrices elementwise and returns
   ↳ the result
15 Matrix& operator++(int);//add one to every element
16 void operator+=(const Matrix&);//adds two Matrices
17 void operator-=(const Matrix&);//subtracts two Matrices
18 ~Matrix();
19 }
20 istream& operator<<(istream& input, const Matrix);//Inputs the Array
21 ostream& operator>>(ostream& ouput, const Matrix); //Outputs the Array
```

---