**MATRIX CLASS**

Matrix Class supports the following operations:

1. Matrix/vector Addition
2. Matrix/vector Subtraction
3. Matrix/vector Multiplication
4. Element Wise Multiplication
5. Scalar Multiplication
6. Reshape
7. Transpose

Matrix is a template class which takes int, float or double as template typename. The private members consist of rows, columns and a container to store elements of the matrix.

To initialize a matrix assignment operator is overloaded to take values.

```
matrix<float> c(3, 2), d(3, 2);
c = {1.0, 2.0, 3.0, 4.0, 5.0, 4.0};
d = {3.0, 4.0, 5.0, 6.0, 7.0, 8.0};
```

To print the values of the matrix a method is created called printContainer().

```
c.printContainer();
1 2
3 4
5 4
```

Matrix addition can be performed between two or more matrices by using + operator. Similarly, matrix subtraction can be performed using - operator.

```
matrix<float> x = c + d;
matrix<float> x = c - d;
```

Matrix transpose is available with Transpose function and reshape function is used to reshape the matrix but matrix size remains the same.

```
matrix<float> y = x.Transpose();
matrix<int> test_reshape(2, 6);
test_reshape.reshape(4, 3);
```

Matrix multiplication is performed with operator *

```
    matrix<float> matmul_a(2, 3), matmul_b(3, 2);
    matmul_a = {1, 2, 3, 4, 5, 6};
    matmul_b = {1, 2, 3, 4, 3.2, 7.5};

    matrix<float> matmul_c = matmul_a * matmul_b;
```

Element wise multiplication is done using elementwise_multiply(mat) function. Scalar multiplication also supported using * operator.

```
matrix<float> elem_mul = 2 * matmul_c.elementwise_multiply(matmul_c) * 2;
```

main.cpp contains the example and usage of matrix class and matrix.h contains the class definition and member functions.