НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра автоматизованих систем обробки інформації і систем управління

**ЛАБОРАТОРНА РОБОТА № 5**

з дисципліни «Об’єктно-орієнтоване програмування»

**на тему: «**STD та інші можливості С++**»**

Виконав:

Студент 2-го курсу групи ІП-72

Кенгерлі Ельмар Фаігович

Київ – 2018 рік

Зміст

[Ціль роботи 4](#_Toc532739625)

[Постановка задачі 5](#_Toc532739626)

[UML-діаграма класів 6](#_Toc532739627)

[Код програми 7](#_Toc532739628)

[Приклади роботи 10](#_Toc532739629)

[Висновки 11](#_Toc532739630)

## Ціль роботи

Вивчити особливості роботи зі стандартною бібліотекою STD.

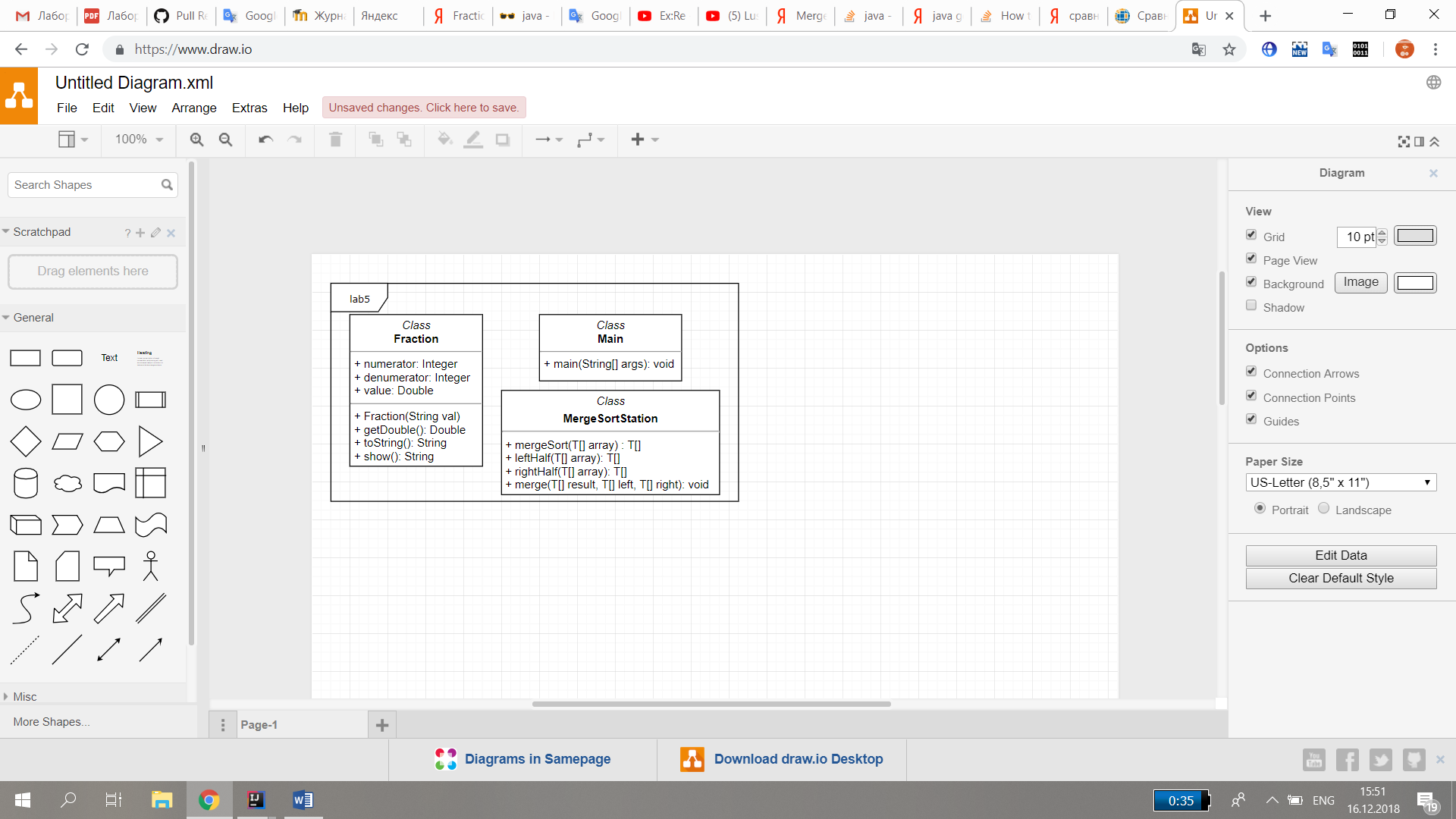
## Постановка задачі

Створити шаблон класу, в якому міститься масив об’єктів заданого типу, крім базових типів, у шаблоні необхідно передбачити використання користувацьких типів, таких як Fraction і String. Для користувацьких типів перегрузити усі необхідні для маніпуляцій операції.

Розробити для даного класу алгоритми сортування злиттям.

Продемонструвати усі дії з об’єктом даного класу на декількох типів даних.

## UML-діаграма класів



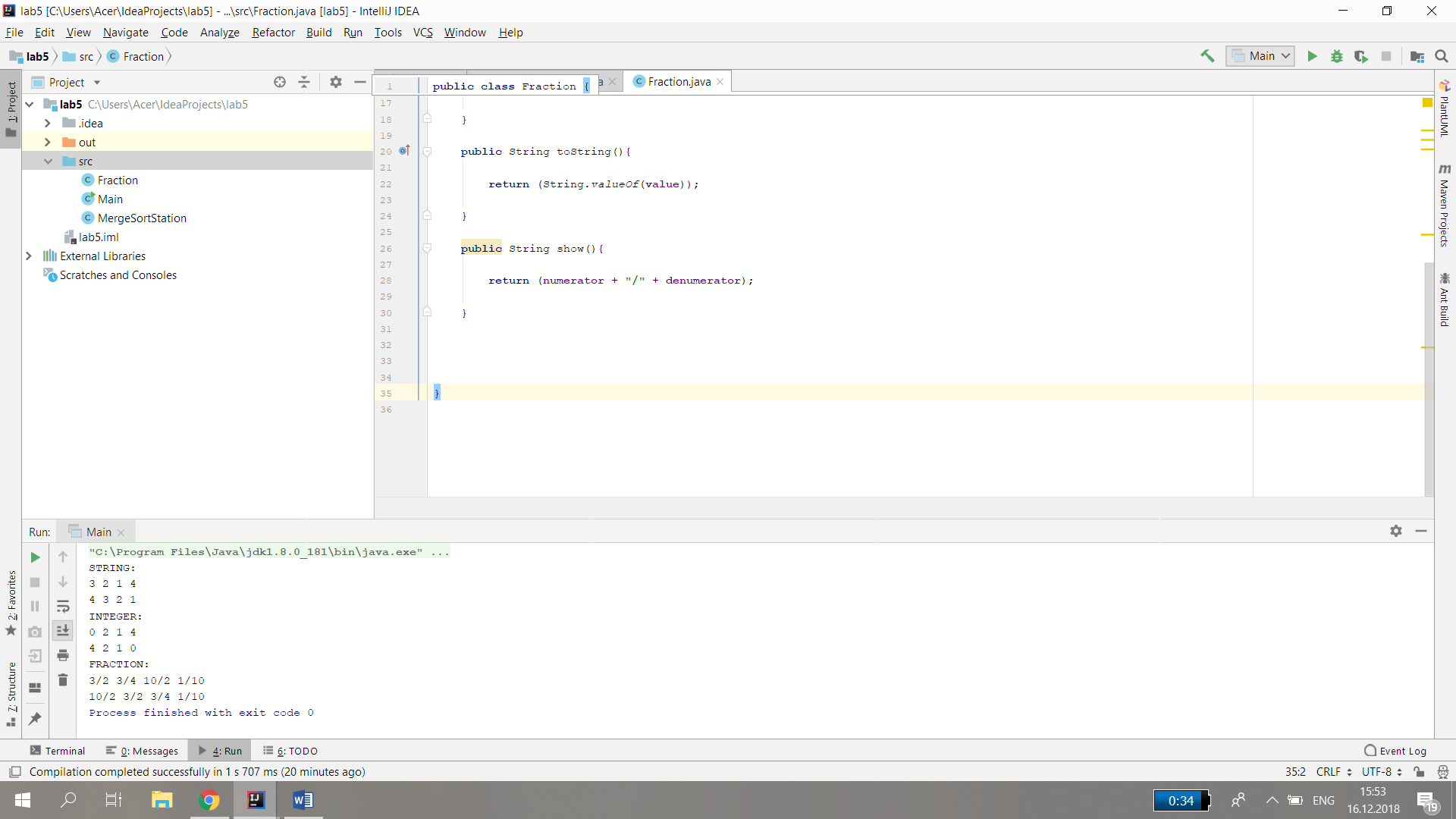
## Код програми

**public class** Main {  
  
 **public static void** main(String[] args) {  
  
 *// Створюємо масиви String, Integer та Fraction* String stringArr[] = {**"3"**,**"2"**,**"1"**,**"4"**};  
 Integer intArr[] = {0, 2, 1, 4};  
 Fraction frArr[] = {  
 **new** Fraction(**"3/2"**), **new** Fraction(**"3/4"**),  
 **new** Fraction(**"10/2"**), **new** Fraction(**"1/10"**)  
 };  
  
  
 *// Перевіримо правильність роботи на типі даних  
 // String* System.***out***.println(**"STRING:"**);  
  
 **for** (String a : stringArr) {  
 *// Масив ДО: 3 2 1 4* System.***out***.print(a + **" "**);  
 }  
  
 System.***out***.print(**"\n"**);  
  
 *// Створюємо сортувальну станцію типу String, сортуємо масив* MergeSortStation<String> station1 = **new** MergeSortStation<>();  
 stringArr = station1.mergeSort(stringArr);  
  
 **for** (String a : stringArr) {  
 *// Масив ПІСЛЯ: 4 3 2 1* System.***out***.print(a + **" "**);  
 }  
  
  
  
  
 *// Перевіримо правильність роботи на типі даних  
 // Integer* System.***out***.println(**"\nINTEGER:"**);  
  
 **for** (**int** a : intArr) {  
 *// Масив ДО: 0 2 1 4* System.***out***.print(a + **" "**);  
 }  
  
 System.***out***.print(**"\n"**);  
  
 *// Створюємо сортувальну станцію типу Integer, сортуємо масив* MergeSortStation<Integer> station2 = **new** MergeSortStation<>();  
 intArr = station2.mergeSort(intArr);  
  
 **for** (**int** a : intArr) {  
 *// Масив ПІСЛЯ: 4 2 1 0* System.***out***.print(a + **" "**);  
 }  
  
  
  
 *// Перевіримо правильність роботи на типі даних  
 // FRACTION* System.***out***.println(**"\nFRACTION:"**);  
  
 **for** (Fraction a : frArr) {  
 *// Масив ДО: 3/2 3/4 10/2 1/10* System.***out***.print(a.show() + **" "**);  
 }  
  
 System.***out***.print(**"\n"**);  
  
 *// Створюємо сортувальну станцію типу Fraction, сортуємо масив* MergeSortStation<Fraction> station3 = **new** MergeSortStation<>();  
 frArr = station3.mergeSort(frArr);  
  
 **for** (Fraction a : frArr) {  
 *// Масив ПІСЛЯ: 10/2 3/2 3/4 1/10* System.***out***.print(a.show() + **" "**);  
 }  
  
 }  
}

**public class** MergeSortStation<T> {  
  
 **public** T[] mergeSort(T[] array) {  
 **if** (array.**length** > 1) {  
 T[] left = leftHalf(array);  
 T[] right = rightHalf(array);  
  
 mergeSort(left);  
 mergeSort(right);  
  
 merge(array, left, right);  
 }  
 **return** array;  
 }  
  
 **public** T[] leftHalf(T[] array) {  
 **int** size1 = array.**length** / 2;  
 T[] left = (T[]) **new** Object[size1];  
  
 **for** (**int** i = 0; i < size1; i++) {  
 left[i] = array[i];  
 }  
  
 **return** left;  
 }  
  
 **public** T[] rightHalf(T[] array) {  
 **int** size1 = array.**length** / 2;  
 **int** size2 = array.**length** - size1;  
 T[] right = (T[]) **new** Object[size2];  
 **for** (**int** i = 0; i < size2; i++) {  
 right[i] = array[i + size1];  
 }  
 **return** right;  
 }  
  
 **public void** merge(T[] result, T[] left, T[] right) {  
 **int** i1 = 0;  
 **int** i2 = 0;  
  
 **for** (**int** i = 0; i < result.**length**; i++) {  
 **if** (i2 >= right.**length** || (i1 < left.**length** && ((left[i1].toString()).compareTo(right[i2].toString())) > 0)) {  
 result[i] = left[i1];  
 i1++;  
 } **else** {  
 result[i] = right[i2];  
 i2++;  
 }  
 }  
 }  
  
}

**public class** Fraction {  
  
 **int numerator**;  
 **int denumerator**;  
 **double value**;  
  
 Fraction ( String val ){  
 String dr[] = val.split(**"/"**);  
 **this**.**denumerator** = Integer.*valueOf*(dr[1]);  
 **this**.**numerator** = Integer.*valueOf*(dr[0]);  
 **value** = ((**double**)**numerator**/(**double**)**denumerator**);  
 }  
  
 **public double** getDouble(){  
  
 **return** ((**double**)**numerator**/(**double**)**denumerator**);  
  
 }  
  
 **public** String toString(){  
  
 **return** (String.*valueOf*(**value**));  
  
 }  
  
 **public** String show(){  
  
 **return** (**numerator** + **"/"** + **denumerator**);  
  
 }  
}

## Приклади роботи



## Висновки

У ході даної лабораторної роботи було створено шаблон класу, який сортує масиви базових типів даних, та користувацьких String і Fraction. Отримана програма пройшла всі підібрані тести, коректно сортує передані масиви елементів заданого типу.