|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| *Федеральное государственное бюджетное образовательное учреждение высшего образования* ***«МИРЭА – Российский технологический университет»***  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**Дисциплина «Программирование на языке Джава»**

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №6**

Выполнил студент группы ИНБО-02-20 Баринов.И.В.

Принял Степанов П.В.

Практические работы выполнены «\_\_\_»\_\_\_\_\_\_\_2021г.

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_»\_\_\_\_\_\_\_2021г.

Отметка о выполнении

**Москва – 2021 г.**

## **Задание**

Целью данной практической работы освоить на практике сортировки различными методами.

## **Ход Работы**

В ходе выполнения работы были получены следующие исходные коды:

package Practise6.test;  
  
public class Exercise1 {  
 public static void selectionSort(int[] arr){  
 /\*По очереди будем просматривать все подмножества  
 элементов массива (0 - последний, 1-последний,  
 2-последний,...)\*/  
 for (int i = 0; i < arr.length; i++) {  
 /\*Предполагаем, что первый элемент (в каждом  
 подмножестве элементов) является минимальным \*/  
 int min = arr[i];  
 int min\_i = i;  
 /\*В оставшейся части подмножества ищем элемент,  
 который меньше предположенного минимума\*/  
 for (int j = i+1; j < arr.length; j++) {  
 //Если находим, запоминаем его индекс  
 if (arr[j] < min) {  
 min = arr[j];  
 min\_i = j;  
 }  
 }  
 /\*Если нашелся элемент, меньший, чем на текущей позиции,  
 меняем их местами\*/  
 if (i != min\_i) {  
 int tmp = arr[i];  
 arr[i] = arr[min\_i];  
 arr[min\_i] = tmp;  
 }  
 }  
 }  
  
 public static void main(String[] args) {  
 int [] Student= {1,2,3,4,5};  
 int [] iDNumber= {1,6,5,321,64,876,324,12,345,86};  
 *selectionSort*(iDNumber);  
 for(int i = 0; i< iDNumber.length; i++) {  
 System.*out*.println(iDNumber[i]);  
 }  
 }  
}

----

package Practise6.test;  
  
import java.util.Arrays;  
  
public class MergeSort {  
 public static void mergeSort(int[] a,int left,int right){  
 if(right<left)  
 return;  
 int mid=(left+right)/2;  
 *mergeSort*(a,left,mid);  
 *mergeSort*(a,mid+1,right);  
 *merge*(a,left,mid,right);  
 }  
 public static void merge(int[]a,int left,int mid,int right){  
 int lenghtleft=mid-left+1;  
 int lenghtright=right-mid;  
  
 int leftAr[]=new int [lenghtleft];  
 int rightAr[]=new int [lenghtright];  
  
 for(int i=0;i<lenghtleft;i++){  
 leftAr[i]=a[left+i];  
 }  
 for(int i=0;i<lenghtright;i++){  
 rightAr[i]=a[mid+i+1];  
 }  
 }  
}

----

package Practise6.test;  
  
import java.util.Arrays;  
import java.util.Comparator;  
  
  
public class SortingStudentsByGPA extends Student {  
  
  
 public static void main(String[] args) {  
 Student[] students = new Student[5];  
 Comparator<Student> scomp = new StudentGPAComparator();  
 students[0] = new Student(64352, 245);  
 students[1] = new Student(5234, 223);  
 students[2] = new Student(1, 222);  
 students[3] = new Student(666, 290);  
 students[4] = new Student(777, 255);  
 String[] strs = {"sdfds","sdfsdf","aaaa"};  
 int[] x = {3434,5,2,1,-9999};  
 Arrays.*sort*(students, scomp);  
 for (int i = 0; i < 5; i++) {  
 System.*out*.println(students[i].getiD() + " " + students[i].getGPA());  
 }  
 }  
}  
  
class StudentGPAComparator implements Comparator<Student>{  
 public int compare(Student a,Student b){  
 if(a.getGPA()>b.getGPA())  
 return 1;  
 else if(a.getGPA()<b.getGPA())  
 return -1;  
 else return 0;  
 }  
}

----

package Practise6.test;  
import java.util.Arrays;  
public class Student {  
 private int iD;  
 private int GPA;  
  
 public Student(int iD,int GPA) {  
 this.iD = iD;  
 this.GPA=GPA;  
 }  
  
 public Student(int i, int gpa, String fio) {  
  
 }  
  
 public int getiD() {  
 return iD;  
 }  
  
 public void setiD(int iD) {  
 this.iD = iD;  
 }  
  
 public int getGPA() {  
 return GPA;  
 }  
  
 public Student() {  
 }  
  
 public void setGPA(int GPA) {  
 this.GPA = GPA;  
 }  
}

## **Вывод**

Получены знания по работе с сортировками различными методами

GitHub: https://github.com/BarinovIvan/Practice6.test