**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №1

# по дисциплине: «Технологии разработки программного обеспечения»

на тему: «**Развитие представлений о разработке программ. Объектно-ориентированное программирование в *Java*»**

Выполнил: студент группы ИТП-21

Бурлаков А. О.

Проверил: преподаватель

Малиновский И. Л.

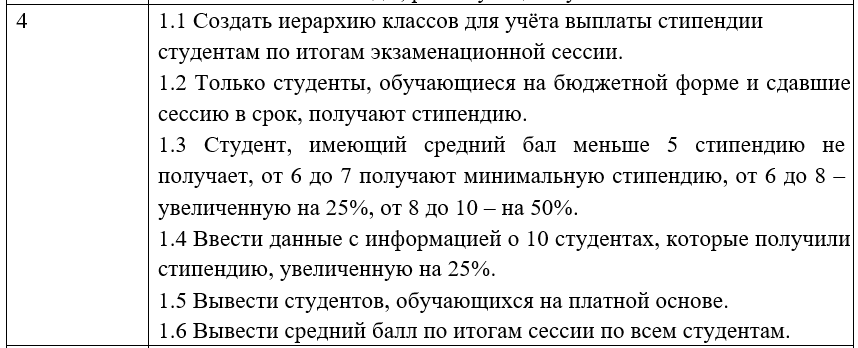
Гомель 2024

**Цель работы:**

1. Разработать *UML*-диаграмму иерархии классов, согласно варианта (таблица 1).
2. При наименовании компонентов руководствоваться соглашением о наименовании ([https://www.oracle.com/technetwork/java/codeconventions-150003.pdf)](https://www.oracle.com/technetwork/java/codeconventions-150003.pdf).
3. При описании иерархии использовать наследование и композицию.
4. На основе *UML*-диаграммы разработать иерархию классов на языке *Java*.
5. Весь код должен быть снабжен элементами документирования

[(https://www.jetbrains.com/help/idea/working-with-code-documentation.html)](https://www.jetbrains.com/help/idea/working-with-code-documentation.html).

1. Разработанную иерархию поместить в .*jar* файл для дальнейшего использования в качестве библиотечных классов.
2. Создать консольное приложение для демонстрации работы созданных классов.
3. Составить отчет о проделанной работе.



На рисунке 1 предоставлена UML-диаграмма программы

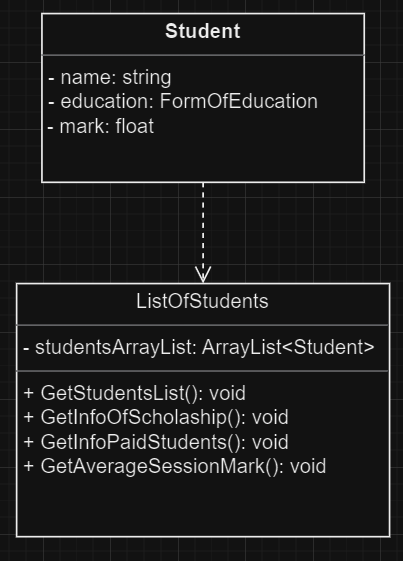


Рисунок 1 – UML-диаграмма

На рисунке 2 предоставлено выполнение задания для варианта №4. Текст программы находится в приложении А

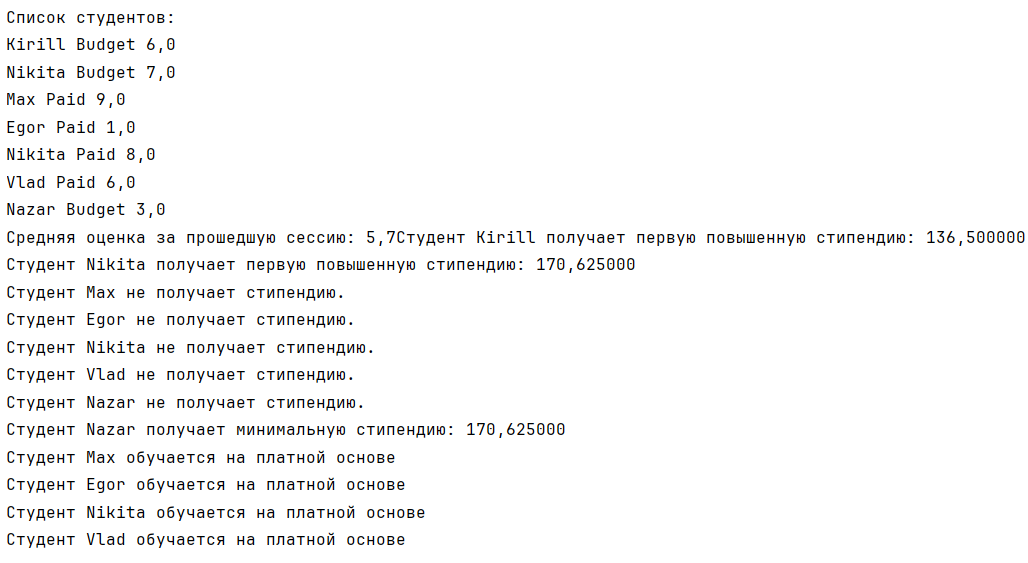


Рисунок 2 – Выполнение программы

**Вывод:** в ходе выполнения лабораторной работы разработали иерархию дальнейшего использования в качестве библиотечных классов. Создали консольное приложение для демонстрации работы созданных классов.

**ПРИЛОЖЕНИЕ А**

**Текст программы**

**Main.java**

import LibClass.\*;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
  
 ListOfStudents listOfStudents = null;  
 ArrayList<Student> students = new ArrayList<Student>();  
 Scanner in = new Scanner(System.*in*);  
  
 System.*out*.print("Введите количество студентов: ");  
 float numberOfStudents = in.nextFloat();  
  
 for(int i = 0; i < numberOfStudents; i++)  
 {  
 System.*out*.print("Введите имя студента: ");  
 String name = in.next();  
 System.*out*.print("Введите форму обучения" +  
 " студента(Budget/Paid): ");  
 String formEducation = in.next();  
  
 FormOfEducation formOfEducation;  
 switch (formEducation)  
 {  
 case ("Budget"):  
 formOfEducation = FormOfEducation.*Budget*;  
 break;  
  
 case ("Paid"):  
 formOfEducation = FormOfEducation.*Paid*;  
 break;  
  
 default:  
 return;  
 }  
 System.*out*.print("Введите оценку по итогам" +  
 " сессии студента: ");  
 int mark = in.nextInt();  
*//* students.add(new Student(name, formOfEducation, mark));  
 }  
  
 listOfStudents = new ListOfStudents(students);  
  
 listOfStudents.GetStudentsList();  
 listOfStudents.GetAverageSessionMark();  
 listOfStudents.GetInfoOfScholarship();  
 listOfStudents.GetInfoPaidStudents();  
 }  
}

**Student.java**

package LibClass;  
  
public class Student {  
 private String name;  
  
 public String GetName()  
 {  
 return name;  
 }  
 private FormOfEducation education;  
  
 public FormOfEducation GetFormOfEducation()  
 {  
 return education;  
 }  
 private float mark;  
  
 public float GetMark()  
 {  
 return mark;  
 }  
  
 public Student(){}  
 public Student(String name, FormOfEducation education, float mark)  
 {  
 this.name = name;  
 this.education = education;  
 this.mark = mark;  
 }  
}

**ListOfStudents.java**

package LibClass;  
  
import java.util.ArrayList;  
  
public class ListOfStudents extends Student  
{  
 private ArrayList<Student> studentsArrayList;  
  
 public ListOfStudents(ArrayList<Student> StudentsArrayList)  
 {  
 studentsArrayList = new ArrayList<Student>();  
  
 for(Student students : StudentsArrayList)  
 {  
 studentsArrayList.add(new Student(students.GetName(),  
 students.GetFormOfEducation(), students.GetMark()));  
 }  
 }  
 public void GetStudentsList()  
 {  
 System.*out*.println("Список студентов:");  
 for(Student student : studentsArrayList)  
 {  
 System.*out*.printf("%s %s %.1f\n",student.GetName(),student.GetFormOfEducation().toString(),  
 student.GetMark());  
 }  
 }  
 public void GetInfoOfScholarship()  
 {  
 float scholarShip = 109.2F;  
 for(Student students : studentsArrayList)  
 {  
 if(students.GetFormOfEducation() == FormOfEducation.*Budget*)  
 {  
 switch ((int) students.GetMark())  
 {  
 case 1:  
 case 2:  
 case 3:  
 System.*out*.printf("Студент %s не получает" +  
 " стипендию.\n", students.GetName());  
 case 4:  
 case 5:  
 System.*out*.printf("Студент %s получает" +  
 " минимальную стипендию: %f\n", students.GetName(), scholarShip);  
 break;  
 case 6:  
 case 7:  
 scholarShip = scholarShip + scholarShip \* 25 / 100;  
 System.*out*.printf("Студент %s получает" +  
 " первую повышенную стипендию: %f\n", students.GetName(), scholarShip);  
 break;  
 case 8:  
 case 9:  
 case 10:  
 scholarShip = scholarShip + scholarShip \* 50 / 100;  
 System.*out*.printf("Студент %s получает" +  
 " вторую повышенную стипендию: %f\n", students.GetName(), scholarShip);  
 break;  
 default:  
 return;  
 }  
 }  
 else  
 {  
 System.*out*.printf("Студент %s не получает" +  
 " стипендию.\n", students.GetName());  
 }  
 }  
 }  
  
 public void GetInfoPaidStudents()  
 {  
 for(Student students : studentsArrayList)  
 {  
 if(students.GetFormOfEducation() == FormOfEducation.*Paid*)  
 {  
 System.*out*.printf("Студент %s обучается" +  
 " на платной основе\n", students.GetName());  
 }  
 }  
 }  
  
 public void GetAverageSessionMark()  
 {  
 int index = 0;  
 float averageMark = 0;  
 for(Student students : studentsArrayList)  
 {  
 index++;  
 averageMark += students.GetMark();  
 }  
 System.*out*.printf("Средняя оценка за прошедшую" +  
 " сессию: %.1f",averageMark = averageMark / index);  
 }  
}

**FormOfEducation.java**

package LibClass;  
  
public enum FormOfEducation  
{  
 *Budget*, *Paid*}