



Labs 11

I. Да се състави интерфейс, пакетен достъп `IOInterface` за четене/запис, формат: (четене/запис) `void read(); void write();`

II. Да се състави клас `Applicant`, пакетен достъп

1. Частни членове

- име - низ,
- дата на раждане - `GregorianCalendar`
- среден успех от кандидатстване - реално число:

2. Експлицитен конструктор

3. Методи четене/запис и стрингова интерпретация:

3.1. Четене/презапис на дата на раждане

3.2. Четене/презапис на име

3.3. Четене/презапис на успех

3.4. Стрингова интерпретация

III. Да се състави клас `Student`, пакетен достъп, наследник на `Applicant`, имплементиращ `Comparable`

1. Частни членове:

- факултетен номер - низ,

2. Експлицитен конструктор

3. Методи:

3.1. четене/презапис на фак.номер

3.2. за равенство по фак. номер

3.3. стрингова интерпретация

3.4. сравнение за по-малко - по избор

IV. Да се състави публичен клас `StudentGroup` за съхраняване на сортирана колекция от студенти (от клас `Student`), имплементиращ интерфейса `IOInterface`

1. Частни членове:

колекция от студенти - тип интерфейс `Set`

имената на входния и изходен файлове - низ

2. Експлицитен конструктор с имената на входния и изходен файлове

Прочита данните от входния файл в колекцията. Формат:

Име<SP>успех<SP>фак.ном<SP>дата

Пример: Simeon 5.5 61362100 23/11/1983 - записи във входния файл

Pesho 4.5 61362101 25/03/1989

Stefan 6.0 61362130 20/12/1995

3. Методи

3.1. За четене/запис на имената на файловете:

3.2. създаване на списък със сортирани по дата на раждане студенти-
връща `ArrayList` като резултат

3.2. имплементиране на интерфейсите методи (четене/запис) `void read()`, `void write()`

3.2. Стрингова интерпретация-създава стринг от данните на колекцията.

4. Главна функция:

4.1. Създава обект от IV, по зададен входен и изходен файл, извежда обекта.

4.2. Записва в изходния файл

4.3. Сортира данните по дата, извежда.