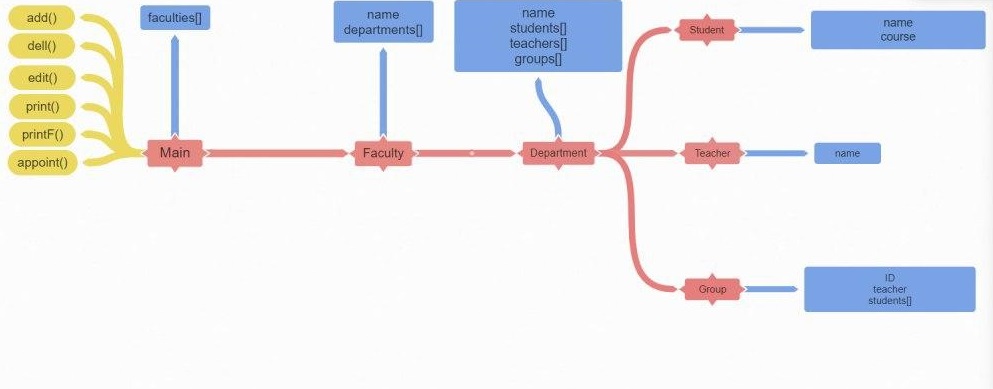
**Lab1**

By Thoruk Igor and Gusar Kyrylo

**Постановка задачі**

Взаємодія з програмою відбувається за допомогою консольного інтерфейсу.

Дерево класів виглядає наступним чином:



**Розподіл задач**

**Тхорук Ігор:**

Основний клас

Javadoc

Ввід/вивід даних

Класи: Faculty,Group.

**Гусар Кирило:**

Алгоритми сортування

Класи: Base, Teacher, Department, Student.

**Структура програми**

Для кожного об’єкту існує свій клас:

* Faculty – факультет. Кожен факультет має ім'я і список кафедр.
* Department – кафедра. Кожна кафедра має: ім'я, список студентів, вчителів та груп.
* Group – група. Кожна група має свій id, вчителя, а також список студентів групи.
* Student – студент. Кожен студент має ім'я та курс.
* Teacher – вчитель. Кожен вчитель має ім’я і предмет.
* Main – основний клас програми.
* Base – клас бази даних. Нажаль, цей функціонал до кінця реалізувати не вийшло, тому клас не використовується.

**Інструкція**

Інтерфейс програми зроблений в вигляді консольної стрічки, тому взаємодія користувача з програмою відбувається за допомогою консольних команд.

**Команда для створення нового об'єкту:**

add [/O] [назва/ID] [/R] [курс]

/O - обєкт(faculty, department, student, teacher, group)

/R - шлях([faculty]/[department]

**Команда видалення об'єкту:**

del [/O] [назва/ID] [/R]

/O - обєкт(faculty, department, student, teacher, group)

/R - шлях([faculty]/[department]

**Команда для редагування існуючого об’єкту:**

edit [/O] [назва/ID] [/R] [нова назва/ID] [новий курс]

/O - обєкт(faculty, department, student, teacher, group)

/R - шлях([faculty]/[department]

**Команда виведення інформації про об’єкт:**

print [/O] [назва/ID] [/R] [/S]

/O - обєкт(faculty, department, student, teacher, group, course)

/R - шлях([faculty]/[department]

/S - сортувати за(name, course)

**Команда виводить студентів або вчителів не відносячи їх до кафедр.**

printF [назва факультету] [/O] [/S]

/O - обєкт(student, teacher)

/S - сортувати за(name, course)

**Kоманда назначення студентів та викладачів до групи**

**Повторне назначення видаляє з групи, а викладачі переназначаються:**

appoint [/O] [назва] [/R] [ID]

/R - шлях([faculty]/[department]

/O - обєкт(student, teacher)

Проблеми з якими ми зіткнулись та їх вирішення

1. Відловлення та виправлення багів. Напевно найтяжча проблема з якою довелось зіткнутись. Я думаю тут взагалі простих рішень немає. В нашому випадку допомогли мануальні тести. При виявленні багу перевірявся алгоритм і виправлялись логічні порушення.
2. Серіалізація та десеріалізація.Цю тему ми ще не вчили, тому була спроба зробити збереження і зчитування даних вручну, тими методами які ми вивчали. Збереження зробити вийшло(хоча не відомо наскільки воно адекватно працює), а от зі зчитуванням виникли складнощі. Розбирати json формат без бібліотек виявилось складно. Була думка використовувати regExp, але навіть з ним код виявився надто складним. Реалізувати зчитування так і не вдалось, тому від ідеї збереження даних довелось відмовитись.