



Государственное бюджетное образовательное учреждение высшего образования
Московской области

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ

КОЛЛЕДЖ КОСМИЧЕСКОГО МАШИНОСТРОЕНИЯ И ТЕХНОЛОГИЙ

РАССМОТРЕНО цикловой комиссией «__» _____ 2020 года	ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № <u>18</u> по модулю <u>ПМ.01 «Разработка программных модулей программного обеспечения для компьютерных систем»</u> специальность <u>09.02.03</u> курс <u>3</u> группы <u>П1-17, П2-17</u>	УТВЕРЖДЕНО Зам. директора по учебной работе _____ 2020 г.
---	---	--

1. По текстовому описанию задачи построить диаграмму прецедентов. **ПК 1.6**
2. По описанию и диаграмме прецедентов построить диаграмму классов. **ПК 1.1**
3. Разработать определения классов по диаграмме. **ПК 1.2**
4. Разработать проект с классами для выполнения задания. **ПК 1.2**
5. Показать отладку одного из модулей при разработке. **ПК 1.3**
6. Подготовить тестовые наборы и провести тестирование одного из модулей. **ПК 1.4**
7. Определить значение временной сложности O для одного из методов. Указать значение O для оптимального кода. **ПК 1.5**

Нормативное время выполнения - 120 минут

Задание для разработки. Туристский клуб

Туристский клуб организует одно – и многодневные пешеходные туры в группах с руководителем по разным маршрутам и разной категории сложности.

Таблицы: Маршруты (Код маршрута, название, продолжительность, категория сложности), Руководители (Код руководителя, ФИО, телефон), Группы (Код группы, название группы, код маршрута, код руководителя, количество туристов, дата отправления).

Требуется:

- определить перечень групп на маршруте;
- построить сравнительную таблицу количества туристов по уровню сложности маршрутов.

Преподаватель, А. Туратин

1. По текстовому описанию задачи построить диаграмму прецедентов.
На рисунке 1 указана диаграмма прецедентов. По ней можно понять, какой функционал будет иметь для работников клуба, или для руководителей групп.

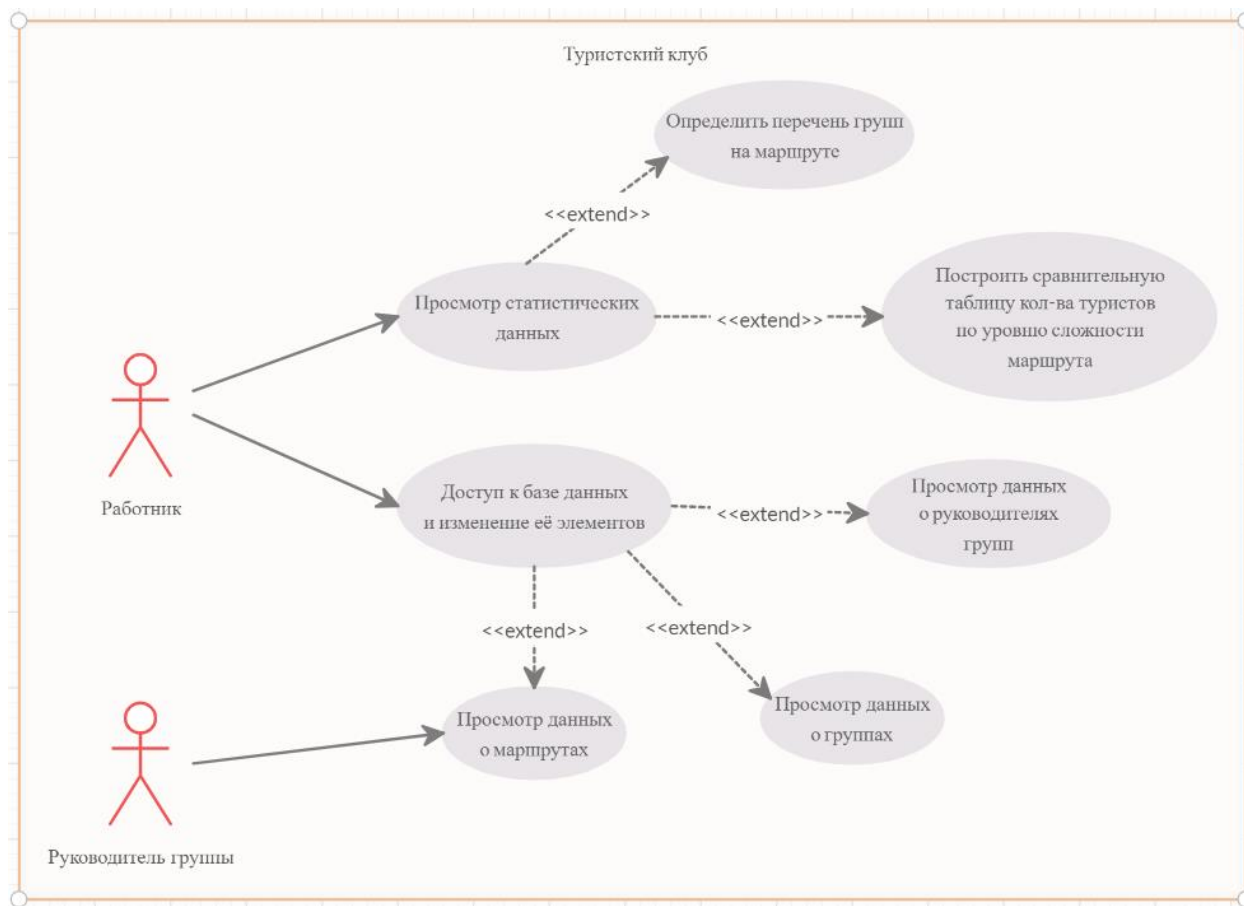


Рисунок 1. Диаграмма прецедентов

2. По описанию и диаграмме прецедентов построить диаграмму классов.

На рисунке 2 показана диаграмма классов, по которой, предположительно будет создаваться программа, если не возникнет иных способов реализации задачи.

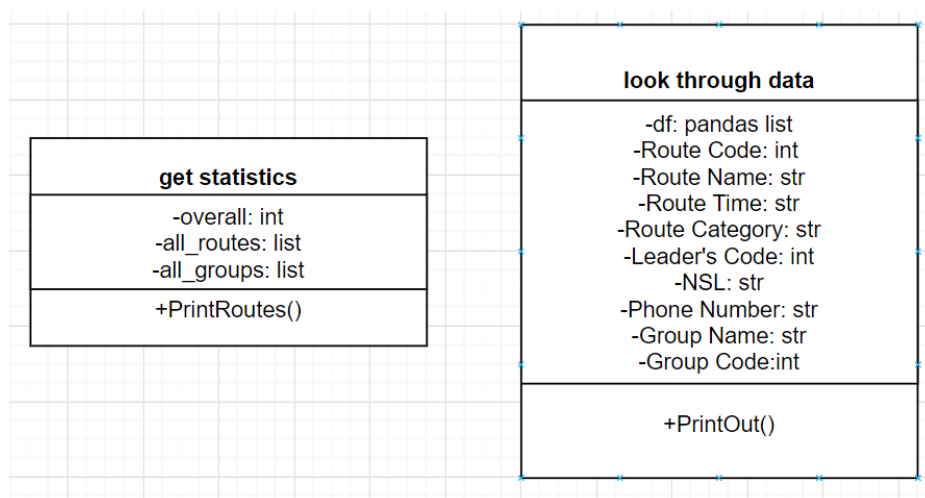


Рисунок 2. Диаграмма классов

3. Разработать определения классов по диаграмме.

После создания диаграммы классов по диаграмме прецедентов и описании задачи было совершено заключение, что диаграмма классов является нечетко определённой. Было решено создать программу с тремя классами для работы с каждой таблицей и более легкой работы. Классы будут реализованы в следующем порядке:

- a. Класс для работы с таблицей “Маршруты”
- b. Класс для работы с таблицей “Руководители”
- c. Класс для работы с таблицей “Группы”

В каждом классе будет содержаться метод вывода таблицы.

4. Разработать проект с классами для выполнения задания.

Код проекта представлен в файле “classes main.py”. Код так же будет приведён далее. Для правильной работы так же необходимы файлы “Tourism” и “Group_Data” с расширением .xlsx.

Код программы из файла “classes main.py”

```
import pandas as pd
#класс для работы с таблицей “Маршруты”
class Table1:
    def __init__(self, event_type_code, type_name):
        self.event_type_code = event_type_code
        self.type_name = type_name

    def PrintOut1():
        df1 = pd.read_excel('Tourism.xlsx')
        print(df1[['Route Code', 'Route Name', 'Route Time', 'Route
Category']])
#класс для работы с таблицей “Руководители”
class Table2:
    def __init__(self, event_type_code, type_name):
        self.event_type_code = event_type_code
        self.type_name = type_name

    def PrintOut2():
        df2 = pd.read_excel('Tourism.xlsx')
        print(df2[["Leader's Code", 'NSL', 'Phone Number']])

#класс для работы с таблицей “Группы”
class Table3:
    def __init__(self, event_code, event_type_code, event_name, start_time,
full_time, done_tick):
        self.event_code = event_code
```

```

        self.event_type_code = event_type_code
        self.event_name = event_name
        self.start_time = start_time
        self.full_time = full_time
        self.done_tick = done_tick

    def PrintOut3():
        df3 = pd.read_excel('Group_Data.xlsx')
        print(df3)

    def PrintRoutes():
        df3 = pd.read_excel('Group_Data.xlsx')
        print(df3[["Group Name", "Route Code"]])

# функция для цикла с главным меню
def MainMenu():
    print('Выберите действие:')
    print('1: Вывести первую таблицу')
    print('2: Вывести вторую таблицу')
    print('3: Вывести третью таблицу')
    print('4: Вывести список групп и их маршрутов')
    print('0: Выход')
    cycle = input('>')
    return cycle

cycle = MainMenu()
# цикл выбора пунктов в главном меню
while cycle != '0':
    if cycle == '1':
        Table1.PrintOut1()
    elif cycle == '2':
        Table2.PrintOut2()
    elif cycle == '3':
        Table3.PrintOut3()
    elif cycle == '4':
        Table3.PrintRoutes()
    else:
        print('Неизвестное действие, используйте предложенный выбор.')
    cycle = MainMenu()

```

5. Показать отладку одного из модулей при разработке.

При запуске программы, происходила ошибка компиляции. В окне запуска появлялся текст, отображённый на рисунке 3.

```
Traceback (most recent call last):
  File "C:/Users/Prog/Desktop/exam/classes_main.py", line 59, in <module>
    Table3.PrintOut3()
  File "C:/Users/Prog/Desktop/exam/classes_main.py", line 32, in PrintOut3
    df3 = pd.read_excel('Group Data.xlsx')
  File "D:\Anaconda\lib\site-packages\pandas\io\excel\_base.py", line 304, in read_excel
    io = ExcelFile(io, engine=engine)
  File "D:\Anaconda\lib\site-packages\pandas\io\excel\_base.py", line 821, in __init__
    self._reader = self._engines[engine](self._io)
  File "D:\Anaconda\lib\site-packages\pandas\io\excel\_xlrd.py", line 21, in __init__
    super().__init__(filepath_or_buffer)
  File "D:\Anaconda\lib\site-packages\pandas\io\excel\_base.py", line 353, in __init__
    self.book = self.load_workbook(filepath_or_buffer)
  File "D:\Anaconda\lib\site-packages\pandas\io\excel\_xlrd.py", line 36, in load_workbook
    return open_workbook(filepath_or_buffer)
  File "D:\Anaconda\lib\site-packages\xlrd\_init_.py", line 111, in open_workbook
    with open(filename, "rb") as f:
FileNotFoundError: [Errno 2] No such file or directory: 'Group Data.xlsx'

Process finished with exit code 1
```

Рисунок 3. До применения средств отладки

В результате анализа была выявлена причина ошибки. Ошибку содержала 32 строка (класс “Table3”, метод “PrintOut3”). Из-за неверно указанного названия колонки, оператор библиотеки pandas не мог найти нужное название, и выдавал ошибку. После исправления, метод работал исправно.

```
Выберите действие:
1: Вывести первую таблицу
2: Вывести вторую таблицу
3: Вывести третью таблицу
4: Вывести список групп и их маршрутов
0: Выход
>3
      Group Code Group Name ... Number of People Date of Departure
0         1      Rota ...           12      2019-01-13
1         2    Ostrov ...           15      2020-06-25
2         3    Ogonek ...           20      2020-03-18
3         4    Zvezda ...            7      2010-05-11
4         5    Kometa ...           20      2019-09-16

[5 rows x 6 columns]
Выберите действие:
1: Вывести первую таблицу
2: Вывести вторую таблицу
3: Вывести третью таблицу
4: Вывести список групп и их маршрутов
0: Выход
>0

Process finished with exit code 0
```

Рисунок 4. После применения средств отладки

6. Подготовить текстовые наборы и провести тестирование одного из модулей.

После передачи действия в главное меню, запускается бесконечный цикл, прерывающийся подачей символа “0”. Реагирует цикл только на символы “1”, “2”, “3”, “4” и “0”. При попытке ввода иных символов цикл переходит на новую итерацию, не производя никаких иных действий. Было принято решение создать подсказку, оповещающую, что на введенный символ реакции не будет.

```
D:\Anaconda\python.exe "C:/Users/Prog/Desktop/exam/classes main.py"
Выберите действие:
1: Вывести первую таблицу
2: Вывести вторую таблицу
3: Вывести третью таблицу
4: Вывести список групп и их маршрутов
0: Выход
>6aпл
Выберите действие:
1: Вывести первую таблицу
2: Вывести вторую таблицу
3: Вывести третью таблицу
4: Вывести список групп и их маршрутов
0: Выход
>
```

Рисунок 5. До применения предложенного решения

```
D:\Anaconda\python.exe "C:/Users/Prog/Desktop/exam/classes main.py"
Выберите действие:
1: Вывести первую таблицу
2: Вывести вторую таблицу
3: Вывести третью таблицу
4: Вывести список групп и их маршрутов
0: Выход
>6666
Неизвестное действие, используйте предложенный выбор.
Выберите действие:
1: Вывести первую таблицу
2: Вывести вторую таблицу
```

Рисунок 6. После применения предложенного решения

7. Определить значение временной сложности O для одного из методов.
Указать значение O для оптимального кода.

В коде файла “classes main.py”, методы всех классов не имеют циклов, от этого они являются линейными и выполняется за некоторое постоянное время, определяемое вычислительной мощностью компьютера, на котором программа запускается. Алгоритмы с таким константным временем всегда выражаются как:

$O(1)$

Рассмотрим метод “PrintOut1” класса “Routes”, из файла “classes main.py”.

Код метода PrintOut1.

```
def PrintOut1():  
    df1 = pd.read_excel('Tourism.xlsx')  
    print(df1[['Route Code', 'Route Name', 'Route Time', 'Route  
Category']])
```

Алгоритм построчного выполнения при вызове этого метода будет иметь константное время. То есть Big O будет равным $O(1)$.

Для всей программы значение O будет равным:

$O(7 + n)$,

где 7 – все методы с линейным алгоритмом, а n – цикл while с потенциально бесконечным количеством итераций.