



Московский государственный университет имени М.В. Ломоносова

Факультет Вычислительной математики и кибернетики

Кафедра Алгоритмических языков

Отчет по заданию практикума по объектно-ориентированному
программированию

Система управления инвестиционным портфелем

Выполнил:

студент 424 группы

Яндутов Алексей Владимирович

Москва, 2021

Оглавление

Уточнение постановки задачи	3
Диаграмма классов	5
Текстовые спецификации интерфейса	5
Class Model	5
Class Market	6
Class MarketAsset, Deposit	7
Class InvestFund	8
Диаграмма объектов	9
Инструментальные средства	9
Описание файловой структуры программы	9
Пользовательский интерфейс	10

Уточнение постановки задачи

Рассматривается работа инвестиционного фонда, который осуществляет различные вложения денежных средств с целью получения прибыли. В игре возможны следующие вложения:

- срочные депозиты банков
Обладают заранее известной фиксированной доходностью и сроком, на который вкладывается депозит
- драгоценные металлы
Имеют низкую волатильность как низкорискованный актив
- государственные облигации
Обладают наиболее низким риском и волатильностью, но предоставляют выплату купонов каждый ход с заранее известной доходностью, которая каждый ход может меняться
- акции различных предприятий
Имеют высокую волатильность по сравнению с другими активами

Создаваемая система реализует экономическую игру, пользователь которой – менеджер, управляющий работой инвестиционного фонда. В начале игры страховая компания обладает некоторым капиталом (например, 100 тыс. у.е.), и известен базовый спрос (т.е. количество потенциальных покупателей) на страховки каждого вида, при заданных начальных условиях страховок. Игрок-менеджер компании может изменить эти условия на новые, а также установить срок их действия (например, 6 месяцев).

Игра моделирует работу фонда в течение M месяцев ($12 \leq M \leq 30$). Шаг моделирования – один месяц, в конце каждого месяца выполняется:

1. подсчет доходности по всем составляющим инвестиционного портфеля, определение общей суммы прибыли и процента доходности за этот месяц;
2. выплата государству налога на доход фонда (например, 17% от суммы прибыли);
3. учет новых поступивших денежных средств фонда (в частности, за счет продажи населению паев фонда);
4. учет расходов фонда (например, в случае возврата паев их держателями);
5. реструктуризация портфеля с учетом изменённой (на шагах 3 и 4) общей суммы капитала фонда и с учетом изменений во внешней конъюнктуре (например, новой стоимости акций).

Операции 1 и 2 выполняются автоматически, операция 5 – игроком-менеджером, а операции 3 и 4 могут выполняться как автоматически, так и игроком. Обычно поступление новых денежных средств в фонд (шаг 3) зависит от его доходности: чем выше доходность за месяц, тем больше спрос на паи фонда и наоборот – падение доходности влечет за собой обратную продажу (возврат) паев (шаг 4).

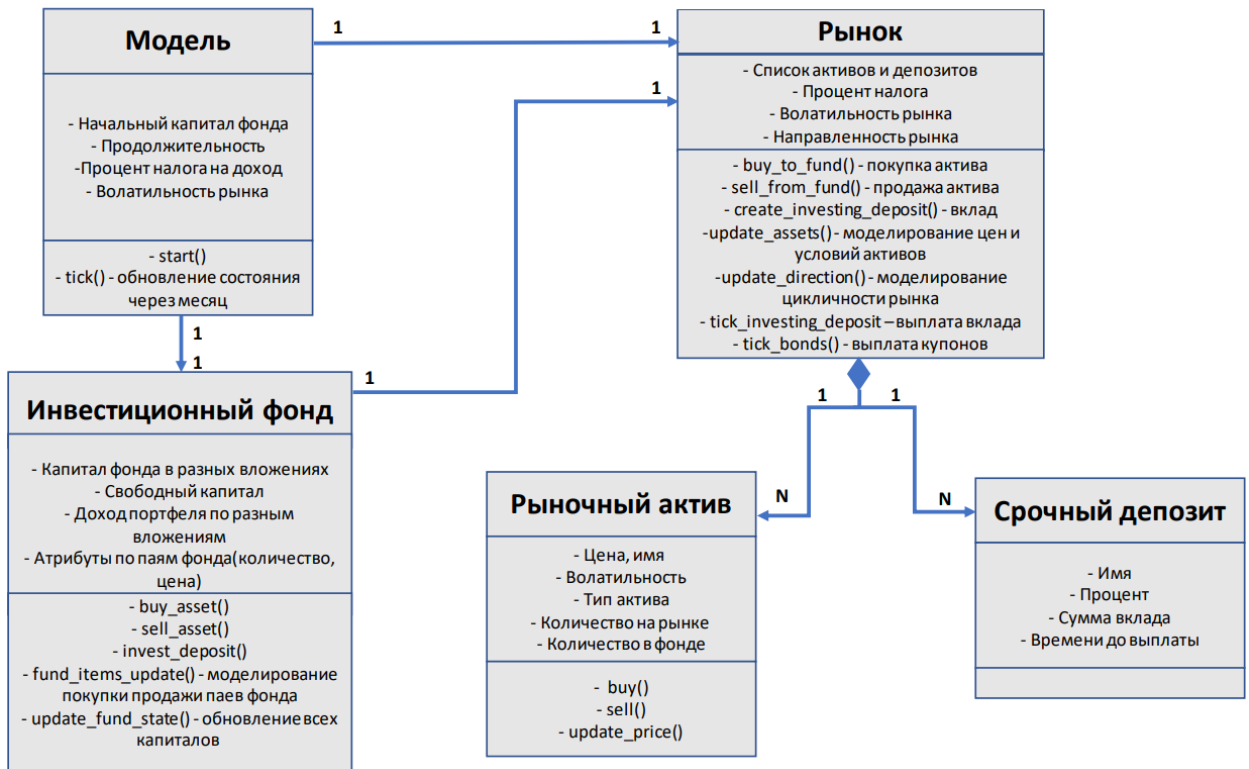
Реструктуризация инвестиционного портфеля может включать, например, продажу части акций или покупку новых, а также вложения в новые депозиты.

При подсчете доходности фонда за месяц (шаг 1) учитывается, что доходность депозита и облигаций известна заранее, а доходность акций и драгоценных металлов определяется внешней конъюнктурой (текущей ценой). Текущую цену следует моделировать как цену прошлого месяца, скорректированную случайной величиной, изменяющейся в некотором диапазоне по определенному вероятностному закону. Аналогичным образом можно моделировать изменение процентных ставок по депозитам по окончании их срока и другие показатели внешней конъюнктуры.

Цель моделирования – выявление пропорций инвестиционного портфеля, позволяющих устойчиво получать прибыль и наращивать общий капитал инвестиционного фонда. В изменяемые параметры целесообразно включить число M , размер исходного капитала инвестиционного фонда, первоначальную структуру инвестиционного портфеля, процент налога на доход, а также диапазоны разброса случайных величин, от которых зависит изменение внешней конъюнктуры.

На каждом шаге игры игроку-менеджеру должны быть доступны все данные о состоянии дел инвестиционного фонда: суммарный капитал, общий доход и доход по отдельным статьям портфеля и др., а также информация о внешней конъюнктуре: цена акций, процентные ставки по депозитам и т.п. По окончании игры можно предусмотреть вывод дополнительных статистических данных о работе фонда, например, уровень продажи и возврата паев фонда.

Диаграмма классов



Текстовые спецификации интерфейса

Class Model

```

class Model():

    ''' Основной класс моделирования '''

    self.start_capital # стартовый капитал

    self.game_duration # продолжительность игры

    self.tax # налог

    self.volatility # волатильность

    self.month_counter # счетчик месяцев

    self.screen_manager # атрибут менеджера окон (для графики)

    self.actions_list # список произошедших действий в игре

    def start(self):

        ''' Старт моделирования '''
  
```

```
def tick(self):  
  
    ''' Переход к следующему ходу '''
```

Class Market

```
class Market():  
  
    ''' Класс, описывающий биржевой рынок '''  
  
    self.market_volatility # величина волатильности рынка  
    self.direction # направленность рынка  
    self.actions_list # список проведенных операций  
    self.tax_perc # налог на доход  
    self.direction_period # длительность цикла рынка  
    self.assets # словарь активов  
    self.deposits # словарь депозитов  
    self.fund_deposits # вклады от фонда  
    self.assets_history # список купленных активов с ценами и количеством  
  
    def update_assets(self):  
  
        ''' Обновление активов при окончании хода'''  
  
    def buy_to_fund(self, name, num):  
  
        ''' Покупка актива'''  
  
    def sell_from_fund(self, name, num):  
  
        ''' Продажа актива фонда'''  
  
    def create_investing_deposit(self, deposit_name, sum):  
  
        ''' Создание депозита-вложения от фонда '''  
  
    def tick_invested_deposits(self):  
  
        ''' Обновление состояния вложений при окончании месяца '''
```

```

def tick_bonds(self):
    ''' Передача фонду выплат по всем облигациям фонда при окончании
    месяца '''

def update_direction(self, month):
    ''' Простое моделирование цикличности рынка '''

```

Class MarketAsset, Deposit

```

class MarketAsset():
    ''' Класс рыночного актива'''

    self.dict_asset_vol # волатильность рынка
    self.name # имя актива
    self.type # тип актива
    self.num_in_fund # кол-во в фонде
    self.percent # процент ежемесячных выплат
    self.price # цена
    self.num_in_market # кол-во на рынке

    def update_price(self, market_direction):
        ''' Обновление цены актива'''

    def buy(self, num):
        ''' Покупка данного актива '''

    def sell(self, num):
        ''' Продажа актива '''

class Deposit():
    '''Класс депозита.

    Логика отличается от активов, поэтому в отличие от активов нет
    аналогичных методов.
    '''

    self.name # имя депозита
    self.time # срок
    self.perc # процент
    self.sum # сумма вклада
    self.time_left # осталось месяцев до выплаты

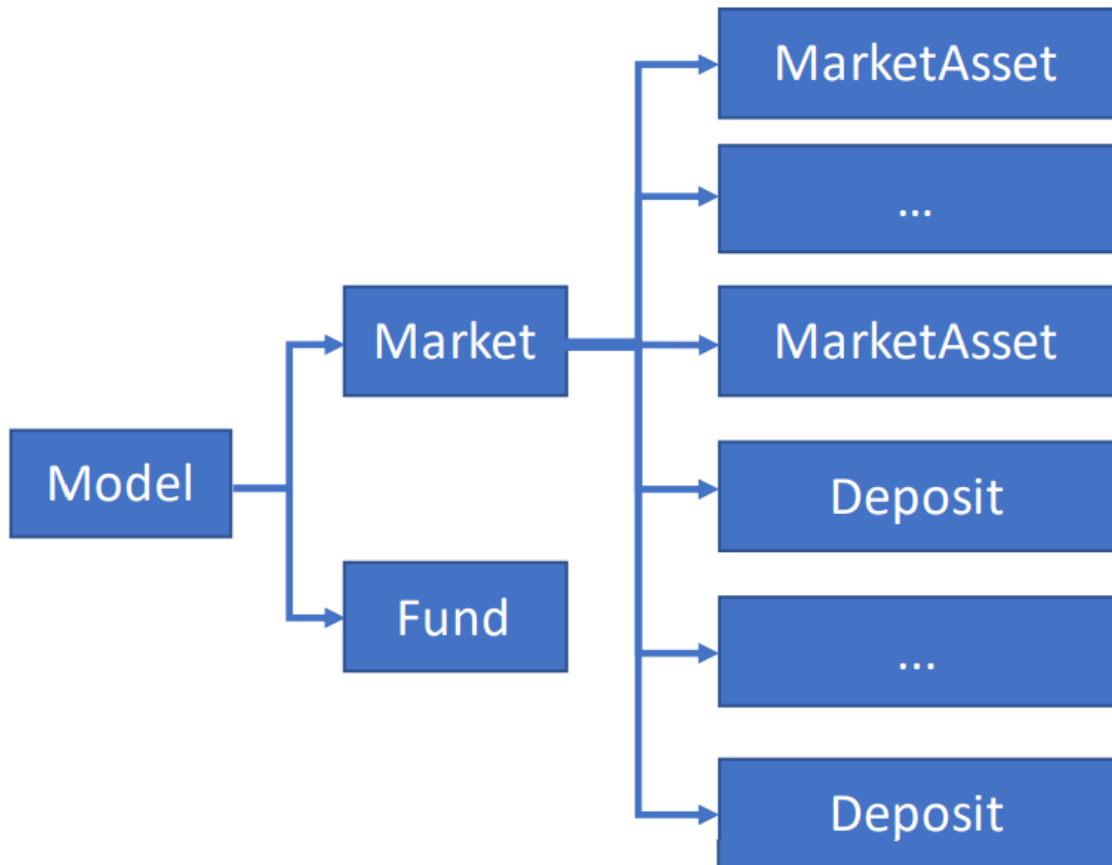
```

Class InvestFund

```
class InvestFund():
    '''
    Класс инвестиционного фонда, которым управляет пользователь
    '''
    self.market # ссылка на рынок
    self.actions_list # история операций
    self.start_capital # стартовый капитал
    self.free_capital # свободный капитал
    self.capital_in_deposits # капитал в депозитах
    self.capital_in_assets # капитал в активах
    self.start_num_items # начальное кол-во паев
    self.total_fund_items # общее число паев
    self.price_fund_item # цена пая
    self.item_profit # прибыль с паев
    self.assets_profit # прибыль с активов
    self.deposit_profit # прибыль с депозитов

    def update_fund_state(self):
        ''' Обновление капитала фонда '''
    def buy_asset(self, name, num):
        ''' Покупка актива '''
    def sell_asset(self, name, num):
        ''' Продажа актива '''
    def fund_items_update(self, old_capital):
        ''' Моделирование покупки/продажи рынком паев фонда '''
    def invest_deposit(self, name, sum):
        ''' Инвестирование в депозит '''
    def get_depo_income(self, income):
        ''' Вызывается при окончании срока депозита '''
    def get_bond_income(self, income):
        ''' Вызывается при получении выплаты от облигаций '''
```


Диаграмма объектов



Инструментальные средства

- Язык программирования – Python 3.7.;
- Среда разработки – Visual Studio Code;
- Используемая графическая библиотека – Kivy, KivyMD.

Описание файловой структуры программы

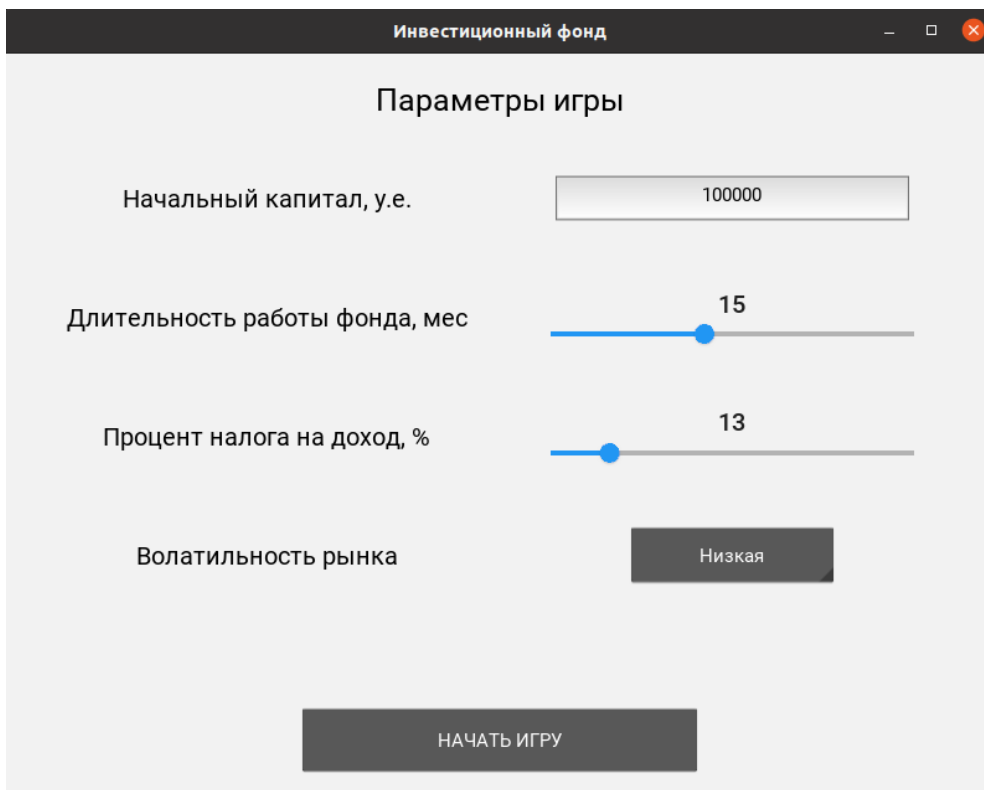
Корневой каталог:

- main.py
- Папка classes:
 - fund_class.py - класс инвестиционного фонда
 - market_class.py - класс рынка

- model.py - класс модели
- Папка GUI:
 - start_window.py - стартовое окно с вводом параметров
 - main_window.py - основное окно игры, на котором представлена информация фонда, активы на рынке, депозиты и различные кнопки
 - asset_window.py - окно покупки/продажи активов
 - deposit_window.py - окно вклада в депозит
 - history_window.py - окно с историей операций и изменения капитала фонда
- graphic.kv - файл со слоями и виджетами графики на языке фреймворка Kivy Design Language

Пользовательский интерфейс

1) При запуске программы появляется окно:



Пользователь вводит начальные условия для моделирования и нажимает на кнопку “Начать игру”.

- 2) Далее открывается основное окно моделирования, в котором находится вся информация про капитал инвестиционного фонда, а также информация состоянии паев фонда, предлагаемые активы на биржевом рынке и список срочных депозитов.

Инвестиционный фонд

Капитал фонда

Общий капитал

В активах

В депозите

Свободных

Всего, руб

Всего, шт

Паи фонда

100000

0

0

100000

Общий доход, руб

0

Месяц 1/15

Закончить ход

История фонда

Начать заново

Список рыночных активов

Название	Тип	Цена	Кол-во в фонде	Выплата, %/мес
ChillUp	Акция	19665	0	0
Yandex	Акция	13894	0	0
ОФЗ 26230	Гос. облигация	1000	0	2.6
ОФЗ 29012	Гос. облигация	1000	0	2.6
ОФЗ 26666	Гос. облигация	1000	0	2.8
Gold	Металл	3941	0	0
Silver	Металл	12860	0	0

Срочные депозиты

Название	Срок, мес	Процент	Денег вложено	До выплаты, мес
Депозит 1	3	4	0	0
Депозит 2	8	12	0	0

Пользователь может взаимодействовать с программой с помощью следующих кнопок и действий:

- “Закончить ход” - моделирование 1 месяца работы рынка и инвестиционного фонда и переход к следующему ходу. Динамика изменения капитала в различных типах вложениях отображается цветом, а также показывается процент изменения по сравнению со стартовым капиталом

Инвестиционный фонд

Капитал фонда

Общий капитал

В активах

В депозите

Свободных

Всего, руб

Всего, шт

Паи фонда

953905

378286

300000

275619

Всего, руб

97

-46095(4.61)

-21746(2.17)

+800(0.08)

Общий доход, руб

-25149(2.51)

Месяц 6/11

Закончить ход

История фонда

Начать заново

Список рыночных активов

Название	Тип	Цена	Кол-во в фонде	Выплата, %/мес
BioMed	Акция	50632	3	0
ChillUp	Акция	45795	0	0
Yandex	Акция	1706	30	0
ОФЗ 26230	Гос. облигация	997	0	2.6
ОФЗ 29012	Гос. облигация	1001	0	2.4
ОФЗ 26666	Гос. облигация	996	100	3.0
Gold	Металл	15122	5	0

Срочные депозиты

Название	Срок, мес	Процент	Денег вложено	До выплаты, мес
Вложение 1	3	4	300000	1
Депозит 1	3	4	0	0
Депозит 2	8	12	0	0

- “Начать заново” - окончание моделирование и переход к стартовому окну выбору параметров для начала новой игры с новыми начальными условиями
- Для покупки либо продажи интересующего актива необходимо нажать на его название в таблице “Список рыночных активов”. При этом всплывает окно покупки/продажи актива с информацией о нем.

Инвестиционный фонд

Покупка/продажа активов

Вернуться

Название:	Yandex
Тип:	Акция
Цена:	13894
В фонде, шт:	0
Свободных денег в фонде:	100000
Купить, шт	<input type="text" value="Введите число от 0 до 7"/>
Продать, шт	<input type="text" value="0"/>

Подтвердить и вернуться

- Для вложения в депозит необходимо нажать на название депозита в таблице “Срочные депозиты”, в этом случае вызывается следующее окно для создания вклада. Созданное вложение отобразится в этой же таблице и будет показано, сколько осталось до выплаты
- “История фонда” - открывает окно с историей изменения капитала фонда и произведенных операциях.

Инвестиционный фонд

История операций фонда

Вернуться

Месяц 2
Покупка BioMed: кол-во - 3 шт., цена: 50075, потрачено: 150225
Начисление выплаты купона по облигациям: +30
Приход паев: 6 шт, +63234 у.е.
Месяц 3
Начисление депозита по: Вложение 0(вклад: 20000, срок: 3, процент: 4, получено: 20800)
Начисление выплаты купона по облигациям: +30
Приход паев: 14 шт, +157318 у.е.
Месяц 4
Вложение в депозит по: Депозит 1 - вклад: 300000, срок: 3, процент: 4

Во всех виджетах, где требуется ввод информации есть валидация вводимых значений, а также подсказки для допустимых диапазонов ввода