

Московский государственный университет имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра алгоритмических языков



Отчет по заданию практикума

Моделирование работы животноводческой фермы

Выполнил: студент 424 группы
Воронцов Александр Алексеевич

Москва, 2022

Содержание

1	Уточнение постановки задачи	3
2	Диаграмма основных классов	4
3	Спецификации интерфейса классов	5
3.1	Класс Graph	5
3.2	Класс Modeling	5
3.3	Класс Recount	6
4	Диаграмма объектов	7
5	Инструментальные средства	8
6	Файловая структура	9
7	Пользовательский интерфейс	10

Глава 1

Уточнение постановки задачи

Рассматривается моделирование работы животноводческой фермы. Пользователь программы - владелец фермы, имеет возможность ввести текущее количество животных (коров или овец), например:

- 70 голов молодняка,
- 90 голов взрослых животных,
- 85 старых животных.

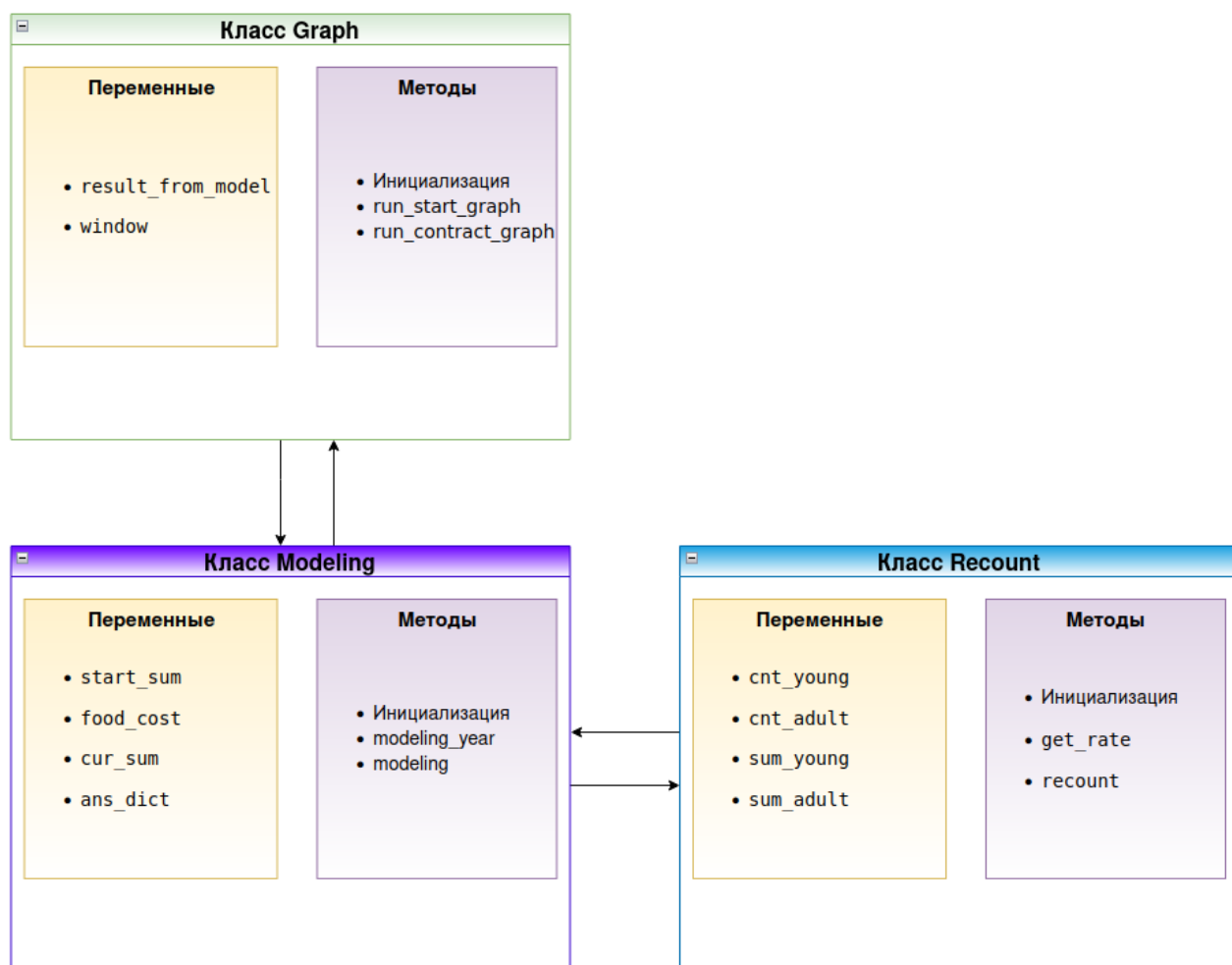
Также пользователь может ввести текущий денежный капитал, например, 100 тыс. у.е., для закупки кормов для животных.

Система моделирования реализует экономическую игру, результатом которой, является контракт с товарной биржей сроком на K лет ($3 \leq K \leq 5$), по которому обязуется ежегодно покупать кормов на определенную сумму, и ежегодно продавать определенное число голов молодняка, взрослых и старых животных. Животные разного возраста продаются по разной цене, которая фиксируется в контракте; цена может меняться год от года. В случае невыполнения обязательств по контракту владелец фермы должен выплатить определенную неустойку (оговариваемую в контракте), за каждое непроданное животное.

Цель моделирования – выявление стратегий контрактации и производства на ферме, позволяющих владельцу фермы приумножить общий капитал по окончании срока контракта.

Глава 2

Диаграмма основных классов



Глава 3

Спецификации интерфейса классов

Ниже представлены классы и их описания.

3.1 Класс Graph

Данный класс, отвечает за визуализацию. Методы класса:

- `__init__(self, theme_color='Purple')` - конструктор с возможностью регулировки цвета окна.
- `create_start_window(self)` - функция для создания стартового окна.
- `ans_layout_create(self, ind, result_from_modeling)` - вспомогательная функция для отрисовки окна контракта.
- `create_contract_window(self, result_from_modeling)` - основная функция для отрисовки окна контракта.
- `show_window(self, window)` - функция для отображения созданного окна.
- `run_start_graph(self)` - функция раннер для стартового окна.
- `run_contract_graph(self, result_from_modeling)` - функция раннер для окна контракта.

3.2 Класс Modeling

Класс, необходим для моделирования работы фермы. Методы класса:

- `__init__(self, animals_info, farm_info)` - конструктор, принимающий на вход информацию о животных, а также о различных коэффициентах (например, `survival_rate_young` - коэффициент выживаемости молодняка).
- `get_cost(self, animal_type, profit_rate=1.4)` - вспомогательная функция для определения цены каждого животного. В условиях отсутствия внешних факторов (таких как кризис) доходность должна составлять 40%.
- `coverage_amount_young(self, cnt_young_next_year, coverage_amount)` - функция для определения количества голов молодняка для продажи.

- `coverage_amount_adult(self, cnt_adult_next_year, coverage_amount)` - функция для определения количества голов взрослых животных для продажи.
- `coverage_amount_old(self, cnt_old_next_year, coverage_amount)` - функция для определения количества голов старых животных для продажи.
- `random_death(self)` - функция для генерации неблагоприятного события. Процент погибающих животных равен 5-20% от общего поголовья.
- `modeling_year(self)` - главная функция моделирования для одного года.
- `modeling(self, n)` - функция раннер для моделирования на n лет.

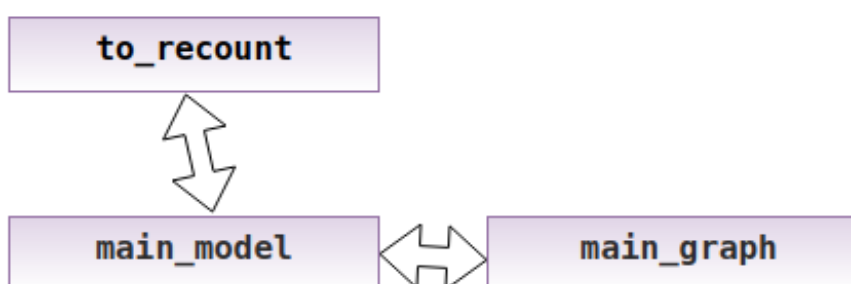
3.3 Класс Recount

Текущий класс предназначен для регулирования цен на продажу. В ходе моделирования могут получиться цены, отличающиеся друг от друга более чем на порядок. Так, например, предлагаемая стоимость старого животного может составить всего 2 тыс. у.е., в то время как стоимость молодого животного может превышать 50 тыс. у.е., что не соответствует реальности. Методы класса:

- `__init__(self, animals_info)` - конструктор, принимающий на вход информацию о животных, в том числе их цены.
- `get_rate(self, cnt)` - вспомогательная функция для определения порогового коэффициента усреднения.
- `recount(self)` - главная функция для пересчёта цен.

Глава 4

Диаграмма объектов



Глава 5

Инструментальные средства

В ходе программы были использованы следующие инструменты.

- Язык программирования: Python 3.8
- Среда разработки: PyCharm
- Используемые библиотеки: PySimpleGUI и random.

Глава 6

Файловая структура

В программе используется несколько модулей. Ниже представлены описания каждого из них.

- `graph.py` - модуль класса `Graph`, в нём перечислены все методы данного класса.
- `modeling.py` - модуль класса `Modeling`, в нём перечислены все методы данного класса.
- `recount.py` - модуль класса `Recount`, в нём перечислены все методы данного класса.
- `main.py` - главный модуль, который запускает всю программу.

Глава 7

Пользовательский интерфейс

При запуске программы пользователю предлагается ввести необходимую информацию:

Животные
Молодые
Количество 70
Взрослые
Количество 20
Старые
Количество 05
Текущий капитал 150000
Стоимость корма необходимого взрослому животному в течении года 1000
Рассчитать оптимальный контракт

После моделирования появляется окно с оптимальным контрактом. Причём пользователю предлагается на стратегию от 1 до 6 лет:

Первый год	Второй год	Третий год	Четвёртый год	Пятый год	Шестой год
Товарная биржа обязуется продать корма на сумму 99333 у.е. Владелец фермы обязуется продать: 1 голова молодняка по цене не ниже 27955 у.е. 0 голов взрослых животных по цене не ниже 0 у.е. 16 голов старых животных по цене не ниже 9783 у.е. В случае невыполнения обязательств владелец обязан выплатить неустойку в размере не более 10892 у.е. за каждое непродавшее животное Доход владельца фермы составит 95166 у.е.	Товарная биржа обязуется продать корма на сумму 146000 у.е. Владелец фермы обязуется продать: 9 голов молодняка по цене не ниже 4932 у.е. 0 голов взрослых животных по цене не ниже 0 у.е. 33 голов старых животных по цене не ниже 4027 у.е. В случае невыполнения обязательств владелец обязан выплатить неустойку в размере не более 4221 у.е. за каждое непродавшее животное Доход владельца фермы составит 116464 у.е.	Товарная биржа обязуется продать корма на сумму 148000 у.е. Владелец фермы обязуется продать: 3 головы молодняка по цене не ниже 6724 у.е. 1 голова взрослых животных по цене не ниже 17681 у.е. 21 голова старых животных по цене не ниже 6272 у.е. В случае невыполнения обязательств владелец обязан выплатить неустойку в размере не более 6724 у.е. за каждое непродавшее животное Доход владельца фермы составит 159763 у.е.	Товарная биржа обязуется продать корма на сумму 147000 у.е. Владелец фермы обязуется продать: 3 головы молодняка по цене не ниже 7069 у.е. 2 головы взрослых животных по цене не ниже 12806 у.е. 22 головы старых животных по цене не ниже 6537 у.е. В случае невыполнения обязательств владелец обязан выплатить неустойку в размере не более 7069 у.е. за каждое непродавшее животное Доход владельца фермы составит 203396 у.е.	Товарная биржа обязуется продать корма на сумму 136000 у.е. Владелец фермы обязуется продать: 3 головы молодняка по цене не ниже 7111 у.е. 1 голова взрослых животных по цене не ниже 18162 у.е. 21 голова старых животных по цене не ниже 6595 у.е. В случае невыполнения обязательств владелец обязан выплатить неустойку в размере не более 7111 у.е. за каждое непродавшее животное Доход владельца фермы составит 244695 у.е.	Товарная биржа обязуется продать корма на сумму 144166 у.е. Владелец фермы обязуется продать: 3 головы молодняка по цене не ниже 7196 у.е. 2 головы взрослых животных по цене не ниже 12788 у.е. 21 голова старых животных по цене не ниже 6664 у.е. В случае невыполнения обязательств владелец обязан выплатить неустойку в размере не более 7196 у.е. за каждое непродавшее животное Доход владельца фермы составит 207662 у.е.

Если первоначального капитала недостаточно и моделирование приведёт к дефолту, то выводится предупреждающее окно:

К сожалению, стартового капитала недостаточно для оптимального контракта
Попробуйте накопить больше денег для взаимодействия с товарной биржей