Министерство образования Республики Беларусь
Учреждение образования
“Белорусский государственный университет информатики и радиоэлектроники”

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ
Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №1
Проектирование программного обеспечения в интеллектуальных системах

Выполнил: ст. гр. 121701
Воронцов Р.Г.

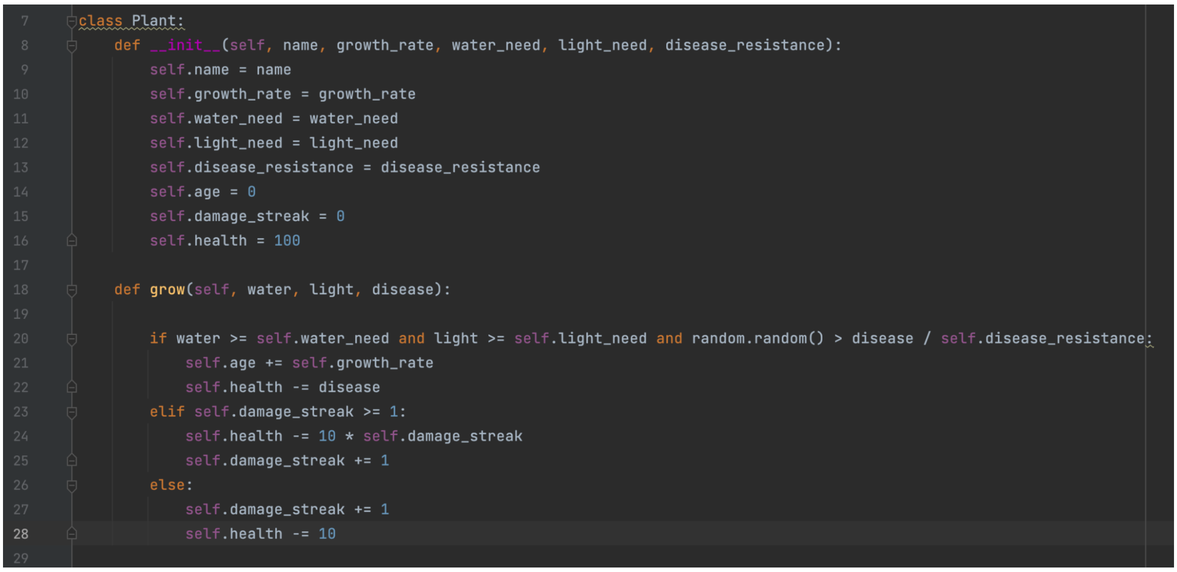
Проверил: Бутрин С.В.

Минск 2023

Задание №4 - Модель садового участка

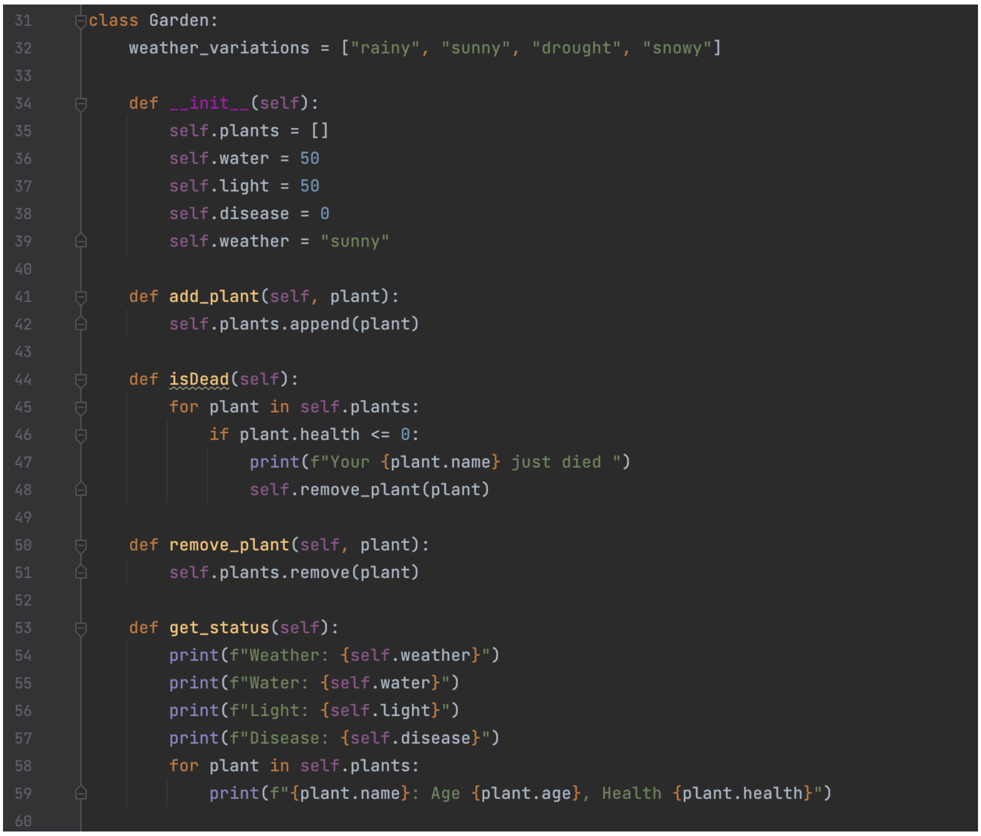
Ход работы:

Описание класса растения (Plant): метод init(self, name, growth_rate, water_need, light_need, disease_resistance) – конструктор класса; метод grow(self, water, light, disease) – рост растения



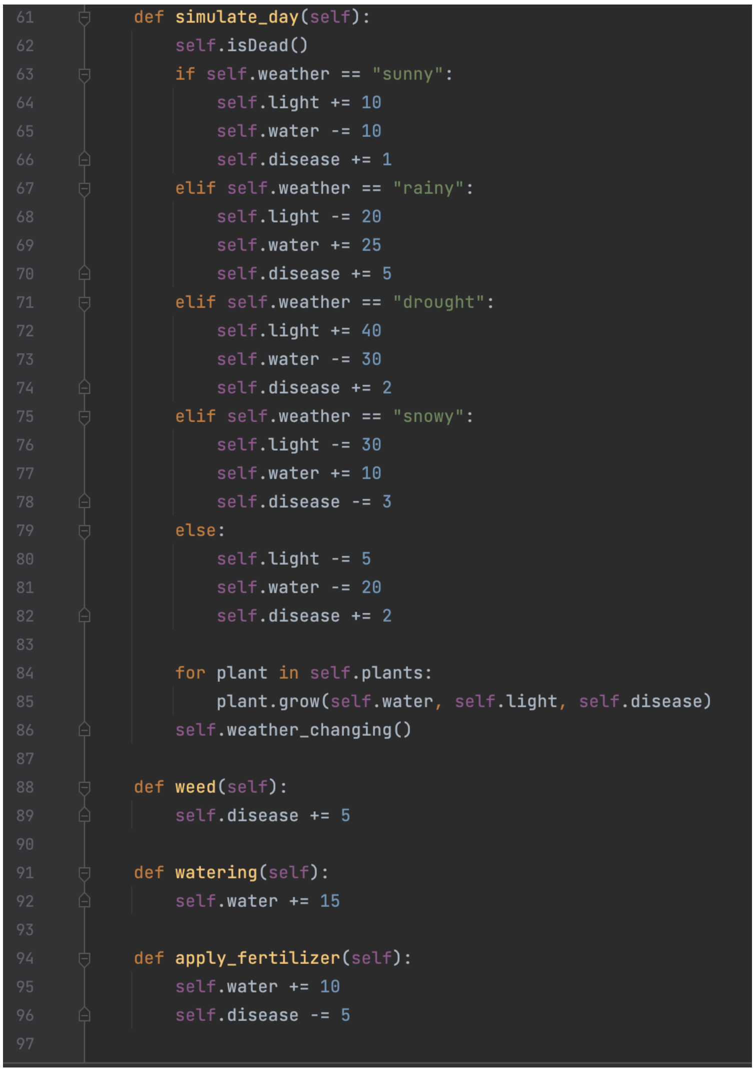
```python
class Plant:
    def __init__(self, name, growth_rate, water_need, light_need, disease_resistance):
        self.name = name
        self.growth_rate = growth_rate
        self.water_need = water_need
        self.light_need = light_need
        self.disease_resistance = disease_resistance
        self.age = 0
        self.damage_streak = 0
        self.health = 100

    def grow(self, water, light, disease):

        if water >= self.water_need and light >= self.light_need and random.random() > disease / self.disease_resistance:
            self.age += self.growth_rate
            self.health -= disease
        elif self.damage_streak >= 1:
            self.health -= 10 * self.damage_streak
            self.damage_streak += 1
        else:
            self.damage_streak += 1
            self.health -= 10

```

Описание класса сада (Garden): init(self) – конструктор класса; add_plant(self, plant) – добавление растения в сад; isDead(self) – смерть растения; remove_plant(self, plant) – удаление растения из сада; get_status(self) – вывод текущего состояния сада



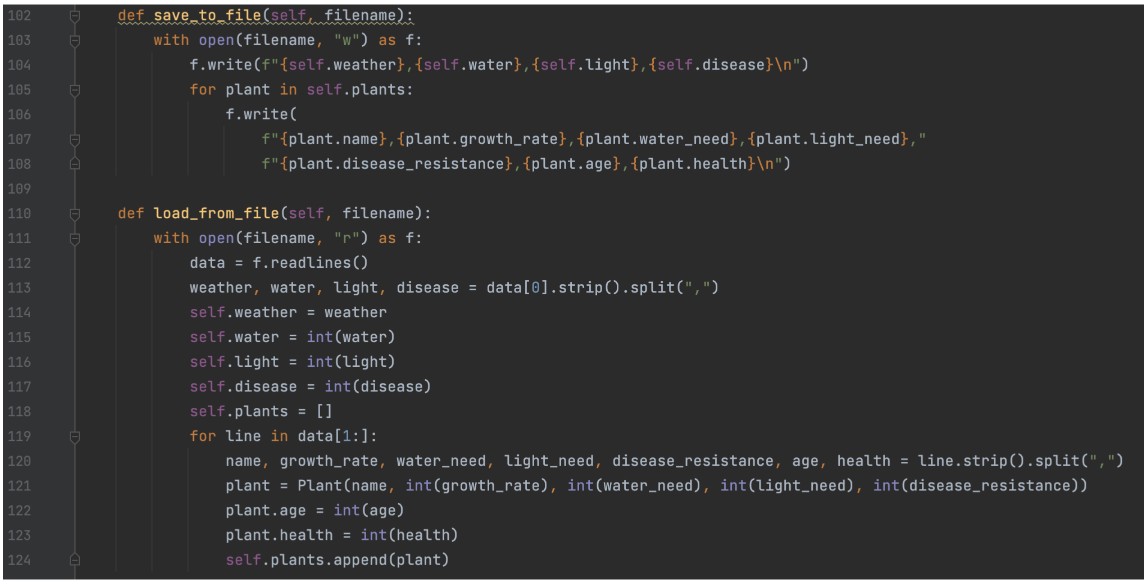
```python
class Garden:
    weather_variations = ["rainy", "sunny", "drought", "snowy"]

    def __init__(self):
        self.plants = []
        self.water = 50
        self.light = 50
        self.disease = 0
        self.weather = "sunny"

    def add_plant(self, plant):
        self.plants.append(plant)

    def isDead(self):
        for plant in self.plants:
            if plant.health <= 0:
                print(f"Your {plant.name} just died ")
                self.remove_plant(plant)

    def remove_plant(self, plant):
        self.plants.remove(plant)

    def get_status(self):
        print(f"Weather: {self.weather}")
        print(f"Water: {self.water}")
        print(f"Light: {self.light}")
        print(f"Disease: {self.disease}")
        for plant in self.plants:
            print(f"{plant.name}: Age {plant.age}, Health {plant.health}")

```

simulate_day(self) – симуляция изменения погоды и роста растений (разный рост в зависимости от погоды); weed(self) – прополка; watering(self) – полив сада; apply_fertilizer(self) – удобрение



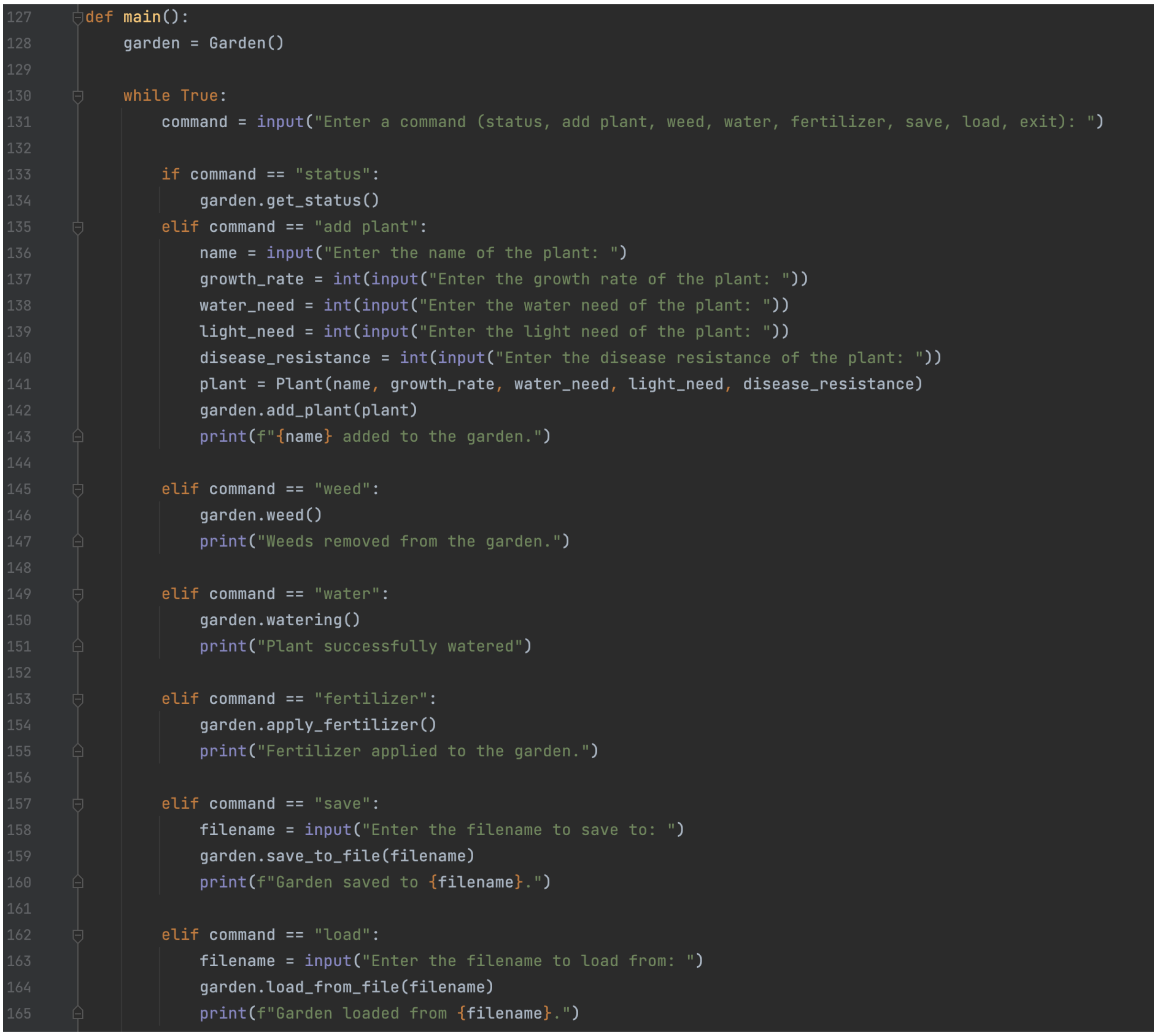
```python
    def simulate_day(self):
        self.isDead()
        if self.weather == "sunny":
            self.light += 10
            self.water -= 10
            self.disease += 1
        elif self.weather == "rainy":
            self.light -= 20
            self.water += 25
            self.disease += 5
        elif self.weather == "drought":
            self.light += 40
            self.water -= 30
            self.disease += 2
        elif self.weather == "snowy":
            self.light -= 30
            self.water += 10
            self.disease -= 3
        else:
            self.light -= 5
            self.water -= 20
            self.disease += 2

        for plant in self.plants:
            plant.grow(self.water, self.light, self.disease)
        self.weather_changing()

    def weed(self):
        self.disease += 5

    def watering(self):
        self.water += 15

    def apply_fertilizer(self):
        self.water += 10
        self.disease -= 5

```

save_to_file(self, filename) – сохранение состояние сада в файл; load_from_file(self, filename) – загрузка состояния сада из файла

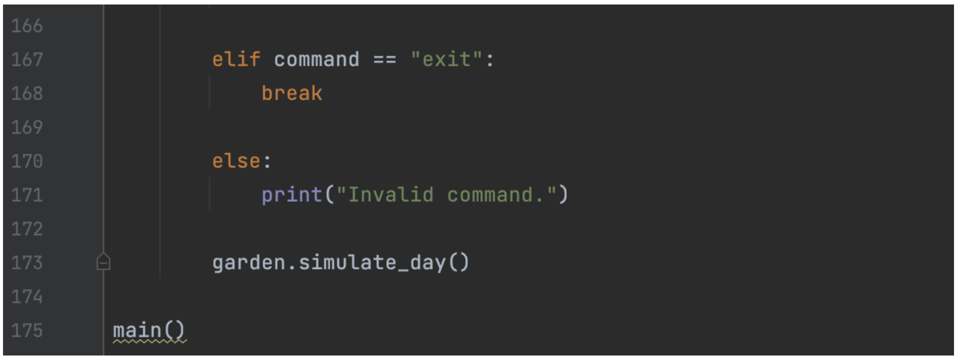


```python
    def save_to_file(self, filename):
        with open(filename, "w") as f:
            f.write(f"{self.weather},{self.water},{self.light},{self.disease}\n")
            for plant in self.plants:
                f.write(
                    f"{plant.name},{plant.growth_rate},{plant.water_need},{plant.light_need},"
                    f"{plant.disease_resistance},{plant.age},{plant.health}\n")

    def load_from_file(self, filename):
        with open(filename, "r") as f:
            data = f.readlines()
            weather, water, light, disease = data[0].strip().split(",")
            self.weather = weather
            self.water = int(water)
            self.light = int(light)
            self.disease = int(disease)
            self.plants = []
            for line in data[1:]:
                name, growth_rate, water_need, light_need, disease_resistance, age, health = line.strip().split(",")
                plant = Plant(name, int(growth_rate), int(water_need), int(light_need), int(disease_resistance))
                plant.age = int(age)
                plant.health = int(health)
                self.plants.append(plant)
```

def main() – комманды ввода, взаимодействие пользователя с системой



```python
def main():
    garden = Garden()

    while True:
        command = input("Enter a command (status, add plant, weed, water, fertilizer, save, load, exit): ")

        if command == "status":
            garden.get_status()
        elif command == "add plant":
            name = input("Enter the name of the plant: ")
            growth_rate = int(input("Enter the growth rate of the plant: "))
            water_need = int(input("Enter the water need of the plant: "))
            light_need = int(input("Enter the light need of the plant: "))
            disease_resistance = int(input("Enter the disease resistance of the plant: "))
            plant = Plant(name, growth_rate, water_need, light_need, disease_resistance)
            garden.add_plant(plant)
            print(f"{name} added to the garden.")

        elif command == "weed":
            garden.weed()
            print("Weeds removed from the garden.")

        elif command == "water":
            garden.watering()
            print("Plant successfully watered")

        elif command == "fertilizer":
            garden.apply_fertilizer()
            print("Fertilizer applied to the garden.")

        elif command == "save":
            filename = input("Enter the filename to save to: ")
            garden.save_to_file(filename)
            print(f"Garden saved to {filename}.")

        elif command == "load":
            filename = input("Enter the filename to load from: ")
            garden.load_from_file(filename)
            print(f"Garden loaded from {filename}.")
```



```python

        elif command == "exit":
            break

        else:
            print("Invalid command.")

        garden.simulate_day()

main()
```