Министерство науки и высшего образования

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

«Уфимский университет науки и технологий»

Кафедра АСУ

Отчет по расчетно-графической работе

Дисциплина: «Объектно-ориентированное моделирование и программирование»

Выполнил: ст. гр. ПИ-215Бз

Швецов. А.Е.

Проверил: канд. техн. наук, доцент

Кромина Л.А.

Уфа – 2023

**Основы объектно-ориентированного подхода   
в языке программирования «Python»**

**Цель:** Освоить основы объектно-ориентированного подхода в языке программирования «Python»

**Задание №1**

Напишите программу на языке программирования Python, содержащую набор характеристик (тактовая частота центрального процессора, количество ядер центрального процессора, объем оперативной памяти компьютера, объем памяти видеокарты, размер свободного пространства на жестком диске, версия Windows), представленных в виде строковых переменных для описания данных о трех компьютерах. В результате работы программы должна выводиться вся информация о каждом компьютере.

**Текст программы**

class Computer:

    def \_\_init\_\_(

*self*,

*cpu\_clock\_speed*,

*cpu\_cores\_numbers*,

*ram\_amount*,

*gpu\_vram\_amount*,

*hdd\_amount\_free\_space*,

*os\_version*,

    ):

*self*.cpu\_clock\_speed = *cpu\_clock\_speed*  *# Тактовая частота ЦП*

*self*.cpu\_cores\_numbers = *cpu\_cores\_numbers*  *# Количество ядер ЦП*

*self*.ram\_amount = *ram\_amount*  *# Объём ОЗУ*

*self*.gpu\_vram\_amount = *gpu\_vram\_amount*  *# Объём VRAM*

*self*.hdd\_amount\_free\_space = *(*

*hdd\_amount\_free\_space*  *# Размер свободного места на HDD*

*)*

*self*.os\_version = *os\_version*  *# Версия OS*

*# Создание метода для вывода всей информации о компьютере*

    def conclusion(*self*):

        print(

            "\n",

            "Тактовая частота ЦП =",

*self*.cpu\_clock\_speed,

            "\n",

            "Количество ядер ЦП =",

*self*.cpu\_cores\_numbers,

            "\n",

            "Объём ОЗУ =",

*self*.ram\_amount,

            "\n",

            "Объём VRAM =",

*self*.gpu\_vram\_amount,

            "\n",

            "Размер свободного места на HDD =",

*self*.hdd\_amount\_free\_space,

            "\n",

            "Версия OS =",

*self*.os\_version,

        )

*# Информация о компьютерах*

Computer1 = Computer(

*cpu\_clock\_speed*="2.4 GHz",

*cpu\_cores\_numbers*="4 cores",

*ram\_amount*="8 GB",

*gpu\_vram\_amount*="2 GB",

*hdd\_amount\_free\_space*="500 GB",

*os\_version*="Windows 10",

)

Computer2 = Computer(

*cpu\_clock\_speed*="4.7 GHz",

*cpu\_cores\_numbers*="6 cores",

*ram\_amount*="16 GB",

*gpu\_vram\_amount*="8 GB",

*hdd\_amount\_free\_space*="1 TB",

*os\_version*="Windows 10 Pro",

)

Computer3 = Computer(

*cpu\_clock\_speed*="2.8 GHz",

*cpu\_cores\_numbers*="6 cores",

*ram\_amount*="12 GB",

*gpu\_vram\_amount*="4 GB",

*hdd\_amount\_free\_space*="2 TB",

*os\_version*="Windows 7",

)

*# Вывод всей информации о всех компьютерах*

Computer1.conclusion()

Computer2.conclusion()

Computer3.conclusion()

**Результат работы программы**

lexot@DESKTOP-EOIUDK6 MINGW64 /d/Github/uust-oomip-distance-learning (master)

$ C:/Users/lexot/AppData/Local/Programs/Python/Python310/python.exe

d:/Github/uust-oomip-distance-learning/rgr.py

 Тактовая частота ЦП = 2.4 GHz

 Количество ядер ЦП = 4 cores

 Объём ОЗУ = 8 GB

 Объём VRAM = 2 GB

 Размер свободного места на HDD = 500 GB

 Версия OS = Windows 10

 Тактовая частота ЦП = 4.7 GHz

 Количество ядер ЦП = 6 cores

 Объём ОЗУ = 16 GB

 Объём VRAM = 8 GB

 Размер свободного места на HDD = 1 TB

 Версия OS = Windows 10 Pro

 Тактовая частота ЦП = 2.8 GHz

 Количество ядер ЦП = 6 cores

 Объём ОЗУ = 12 GB

 Объём VRAM = 4 GB

 Размер свободного места на HDD = 2 TB

 Версия OS = Windows 7

**Задание №2**

На языке программирования Python создайте класс Computer. Укажите атрибуты, рассмотренные в задании 1. Добавьте метод под названием loading\_the\_operating\_system («загрузка операционной системы»). Создайте объект класса Computer, установите атрибуты, вызовите метод loading\_the\_operating\_system. В результате работы программы должна выводиться строка, содержащая значение любого из описанных атрибутов, а также сообщение о том, что операционная система загружена.

**Текст программы**

class Computer:

    def \_\_init\_\_(

*self*,

*cpu\_clock\_speed*,

*cpu\_cores\_numbers*,

*ram\_amount*,

*gpu\_vram\_amount*,

*hdd\_amount\_free\_space*,

*os\_version*,

    ):

*self*.cpu\_clock\_speed = *cpu\_clock\_speed*  *# Тактовая частота ЦП*

*self*.cpu\_cores\_numbers = *cpu\_cores\_numbers*  *# Количество ядер ЦП*

*self*.ram\_amount = *ram\_amount*  *# Объём ОЗУ*

*self*.gpu\_vram\_amount = *gpu\_vram\_amount*  *# Объём VRAM*

*self*.hdd\_amount\_free\_space = *(*

*hdd\_amount\_free\_space*  *# Размер свободного места на HDD*

*)*

*self*.os\_version = *os\_version*  *# Версия OS*

*# Создание метода «загрузка операционной системы»*

    def loading\_the\_operating\_system(*self*):

        print("\n", "Операционная система", *self*.os\_version, "загружена")

*# Информация о компьютерах*

Computer1 = Computer(

*cpu\_clock\_speed*="2.4 GHz",

*cpu\_cores\_numbers*="4 cores",

*ram\_amount*="8 GB",

*gpu\_vram\_amount*="2 GB",

*hdd\_amount\_free\_space*="500 GB",

*os\_version*="Windows 10",

)

Computer2 = Computer(

*cpu\_clock\_speed*="4.7 GHz",

*cpu\_cores\_numbers*="6 cores",

*ram\_amount*="16 GB",

*gpu\_vram\_amount*="8 GB",

*hdd\_amount\_free\_space*="1 TB",

*os\_version*="Windows 10 Pro",

)

Computer3 = Computer(

*cpu\_clock\_speed*="2.8 GHz",

*cpu\_cores\_numbers*="6 cores",

*ram\_amount*="12 GB",

*gpu\_vram\_amount*="4 GB",

*hdd\_amount\_free\_space*="2 TB",

*os\_version*="Windows 7",

)

*# Сообщение о том, что ОS загружена*

Computer1.loading\_the\_operating\_system()

Computer2.loading\_the\_operating\_system()

Computer3.loading\_the\_operating\_system()

**Результат работы программы**

lexot@DESKTOP-EOIUDK6 MINGW64 /d/Github/uust-oomip-distance-learning (master)

$ C:/Users/lexot/AppData/Local/Programs/Python/Python310/python.exe

d:/Github/uust-oomip-distance-learning/rgr2.py

 Операционная система Windows 10 загружена

 Операционная система Windows 10 Pro загружена

 Операционная система Windows 7 загружена

**Задание №3**

На языке программирования Python создайте класс Computer. Укажите атрибуты, рассмотренные в задании 1 и в задании 2. Узнайте размер оперативной памяти компьютера через вызов метода get, затем измените данное значение через вызов метода set. В результате работы программы должны выводиться два значения: начальное и конечное.

**Текст программы**

class Computer:

    def \_\_init\_\_(

*self*,

*cpu\_clock\_speed*,

*cpu\_cores\_numbers*,

*ram\_amount*,

*gpu\_vram\_amount*,

*hdd\_amount\_free\_space*,

*os\_version*,

    ):

*self*.cpu\_clock\_speed = *cpu\_clock\_speed*  *# Тактовая частота ЦП*

*self*.cpu\_cores\_numbers = *cpu\_cores\_numbers*  *# Количество ядер ЦП*

*self*.ram\_amount = *ram\_amount*  *# Объём ОЗУ*

*self*.gpu\_vram\_amount = *gpu\_vram\_amount*  *# Объём VRAM*

*self*.hdd\_amount\_free\_space = *(*

*hdd\_amount\_free\_space*  *# Размер свободного места на HDD*

*)*

*self*.os\_version = *os\_version*  *# Версия OS*

*# Создание метода получения значения объёма ОЗУ*

    def get\_ram\_amount(*self*):

*return* *self*.ram\_amount

*# Создание метода изменения значения объёма ОЗУ*

    def set\_ram\_amount(*self*, *newName*):

*self*.ram\_amount = *newName*

*# Информация о компьютерах*

Computer1 = Computer(

*cpu\_clock\_speed*="2.4 GHz",

*cpu\_cores\_numbers*="4 cores",

*ram\_amount*="8 GB",

*gpu\_vram\_amount*="2 GB",

*hdd\_amount\_free\_space*="500 GB",

*os\_version*="Windows 10",

)

Computer2 = Computer(

*cpu\_clock\_speed*="4.7 GHz",

*cpu\_cores\_numbers*="6 cores",

*ram\_amount*="16 GB",

*gpu\_vram\_amount*="8 GB",

*hdd\_amount\_free\_space*="1 TB",

*os\_version*="Windows 10 Pro",

)

Computer3 = Computer(

*cpu\_clock\_speed*="2.8 GHz",

*cpu\_cores\_numbers*="6 cores",

*ram\_amount*="12 GB",

*gpu\_vram\_amount*="4 GB",

*hdd\_amount\_free\_space*="2 TB",

*os\_version*="Windows 7",

)

*# Вывод изначального объёма ОЗУ*

print("\n", "Изначальный объём ОЗУ =", Computer1.get\_ram\_amount())

*# Изменение значения через вызов метода «set\_ram\_amount»*

Computer1.set\_ram\_amount("24")

*# Вывод итогового объёма ОЗУ*

print("\n", "Итоговый объём ОЗУ =", Computer1.get\_ram\_amount())

**Результат работы программы**

lexot@DESKTOP-EOIUDK6 MINGW64 /d/Github/uust-oomip-distance-learning (master)

$ C:/Users/lexot/AppData/Local/Programs/Python/Python310/python.exe

d:/Github/uust-oomip-distance-learning/rgr3.py

 Изначальный объём ОЗУ = 8 GB

 Итоговый объём ОЗУ = 24 GB

**Вывод о проделанной лабораторной работе:** Данная работа помогла освоить и углубиться в основы объектно-ориентированного подхода в языке программирования «Python»