

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И. Носова»



АЮ:  
Института  
Лукьянов  
2017 г.

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

*ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ*

Направление подготовки  
09.03.01 Информатика и вычислительная техника

Профиль программы  
Программное обеспечение средств вычислительной техники и  
автоматизированных систем

Уровень высшего образования – бакалавриат

Программа подготовки – академический бакалавриат

Форма обучения  
Очная

Институт	<i>энергетики и автоматизированных систем</i>
Кафедра	<i>вычислительной техники и программирования</i>
Курс	2
Семестр	4

Магнитогорск  
2017 г.

Рабочая программа составлена на основе ФГОС ВО по направлению подготовки (специальности) 09.03.01 Информатика и вычислительная техника, утвержденного приказом МО и Н РФ от 12.01.2016 № 5.

Рабочая программа рассмотрена и одобрена на заседании кафедры вычислительной техники и программирования «26» сентября 2017 г., протокол № 2.

Зав. кафедрой  / О.С. Логунова/

Рабочая программа одобрена методической комиссией института энергетики и автоматизированных систем «27» сентября 2017 г., протокол № 2.

Председатель  / С.И. Лукьянов/

Рабочая программа составлена:

доцентом каф. ВтиП

 / М.В. Зарецким/  
(подпись) (И.О. Фамилия)

Рецензент:

начальник отдела инновационных разработок ЗАО «КонсОмСКБ», канд. техн. наук

 / А.Н. Панов/

## Лист актуализации рабочей программы


---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2017-2018 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от 26 09 2017 г. № 2  
Зав. кафедрой  О.С. Логунова

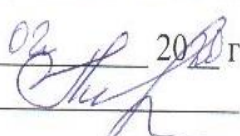
---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2018 - 2019 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от 5 09 2018 г. № 1  
Зав. кафедрой  О.С. Логунова

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2019 - 2020 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от 19 09 2019 г. № 5  
Зав. кафедрой  О.С. Логунова

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2020 - 2021 учебном году на заседании кафедры Вычислительной техники и программирования

Протокол от 19 09 2020 г. № 5  
Зав. кафедрой  О.С. Логунова

## **1 Цели освоения дисциплины (модуля)**

Целями освоения дисциплины «Объектно-ориентированное программное обеспечение» являются:

формирование у студентов понятия об объектно-ориентированной парадигме моделирования бизнес-процессов и ее современных реализациях;

освоение методологии адаптации и применения объектно-ориентированного программного обеспечения для моделирования бизнес-процессов;

выработка компетенций, позволяющих определять применимость данного объектно-ориентированного программного обеспечения в конкретных условиях;

выработка компетенций, позволяющих создавать комплексные решения, в которых эффективно используется объектно-ориентированное программное обеспечение.

Для достижения поставленных целей в курсе «Объектно-ориентированное программное обеспечение» решаются задачи:

- освоение методов объектно-ориентированного анализа предметной области;
- освоение методов объектно-ориентированного моделирования;
- изучение современных применений объектно-ориентированной парадигмы программирования;
- изучение современных объектных систем моделирования бизнес-процессов.

## **2 Место дисциплины (модуля) в структуре образовательной программы подготовки бакалавра (магистра, специалиста)**

Дисциплина «Объектно-ориентированное программирование» входит в вариативную часть блока 1 образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения следующих дисциплин:

- математики (базовая часть блока 1 образовательной программы). Знания, умения и владения, полученные при изучении данной дисциплины, позволят обучающимся осмысленно применять понятие функции, грамотно строить суперпозиции функций;
- информатики (базовая часть блока 1 образовательной программы). Знания, умения и владения, полученные при изучении данной дисциплины, являются основой для освоения средств обработки информации в соответствии с функциональной парадигмой;
- прикладного программирования (базовая часть блока 1 образовательной программы). Знания, умения и владения, полученные при изучении данной дисциплины, являются основой для освоения методологии разработки программ в функциональной парадигме.

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения следующих дисциплин:

- систем автоматизированного проектирования (базовая часть блока 1 образовательной программы). Знания, умения и владения, полученные при изучении дисциплины «Объектно-ориентированное программирование» позволят обучающимся осмысленно применять общепринятые в современных системах автоматизированного проектирования методы представления информации о проектируемом объекте в виде сложных вложенных списков;
- методы управления знаниями (базовая часть блока 1 образовательной программы). Знания, умения и владения, полученные при изучении дисциплины «Объектно-ориентированное программирование» позволят обучающимся осмысленно применять современные программные средства инженерии знаний, основанные на функциональной парадигме;
- методы анализа информации (базовая часть блока 1 образовательной программы). Знания, умения и владения, полученные при изучении дисциплины «Объектно-ориентированное программирование» позволят обучающимся осмысленно применять современные программные средства компьютерного анализа неструктурированной информации, основанные на объектной парадигме;

### 3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Объектно-ориентированное программное обеспечение» обучающийся должен обладать следующими компетенциями:

Структурный элемент компетенции	Планируемые результаты обучения
ПК-2. Обладает способностью разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.	
Знать	<ul style="list-style-type: none"><li>– основные принципы построения объектно-ориентированного программного обеспечения; основы методологии его использования, принципы агентной парадигмы моделирования;</li><li>– методологию канонической декомпозиции предметной области, моделирования в виде системы взаимодействующих агентов;</li><li>– методологию построения иерархических объектных моделей предметной области, выбора между дискретным и непрерывным представлением, выбора между детерминированным и стохастическим представлением модели.</li></ul>
Уметь	<ul style="list-style-type: none"><li>– разрабатывать агентные модели;</li><li>– разрабатывать дискретно-событийные модели;</li><li>– разрабатывать модели системной динамики;</li><li>– разрабатывать многоподходные модели.</li></ul>
Владеть	<ul style="list-style-type: none"><li>– методами применения современных объектно-ориентированных программных средств;</li><li>– методами применения средств поддержки мультипарадигменного моделирования;</li><li>– методами расширения функциональности средств мультипарадигменного моделирования.</li></ul>

#### 4 Структура и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 4 зачетные единицы 144 академических часа, в том числе:

- контактная работа – 54,15 академических часов:
  - аудиторная – 51 академический час;
  - внеаудиторная – 3,15 академических часов
- самостоятельная работа – 54,15 академических часов;
- подготовка к экзамену – 35,7 академических часов

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в академических часах)			Самостоятельная работа (в академических часах)	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код и структурный элемент компетенции
		лекции	лаборат. занятия	практич. занятия				
1. Раздел 1. Объектный подход к моделированию.	4							
1.1. Тема. Понятие о моделях. Виды моделей. Реализация моделей в объектной парадигме.	4	2	2		6	Самостоятельное изучение учебной и научной литературы.	Беседа – обсуждение. Устный опрос.	ПК-2 – зув
1.2. Тема. Инструментальные средства объектного моделирования. Моделирование в системе Anylogic.	4	2	2		6	Самостоятельное изучение учебной и научной литературы. Подготовка к лабораторному занятию. Выполнение лабораторной работы.	Беседа – обсуждение. Анализ программного кода. Устный опрос.	ПК-2 – зув
<b>Итого по разделу</b>	<b>4</b>	<b>4</b>	<b>4</b>		<b>12</b>		<b>Проверка индивидуальных заданий</b>	
2. Раздел 2. Агентное моделирование в объектной парадигме.	4							
2.1. Тема. Понятие об агенте и мультиагентной системе и их объектной	4	2	4/2И		6	Самостоятельное изучение учебной и научной литературы.	Беседа – обсуждение. Анализ программного кода.	ПК-2 – зув

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в акад. часах)			Самостоятельная работа (в акад. часах)	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код и структурный элемент компетенции
		лекции	лаборат. занятия	практич. занятия				
реализации.						Подготовка к лабораторному занятию. Выполнение лабораторной работы.	Устный опрос.	
2.2. Тема. Объектные средства агентного моделирования детерминированных и недетерминированных объектов.	4	2	4/2И		6	Самостоятельное изучение учебной и научной литературы. Подготовка к лабораторному занятию. Выполнение лабораторной работы.	Беседа – обсуждение. Анализ программного кода. Устный опрос.	ПК-2 – зув
<b>Итого по разделу</b>	<b>4</b>	<b>4</b>	<b>8/4И</b>		<b>12</b>		<b>Проверка индивидуальных заданий</b>	
3. Раздел. Дискретно-событийное моделирование. Системная динамика. Объектная реализация.								
3.1. Тема. . Объектные средства дискретно-событийного моделирования детерминированных и недетерминированных объектов	4	2	6		6	Самостоятельное изучение учебной и научной литературы. Подготовка к лабораторному занятию. Выполнение лабораторной работы.	Беседа – обсуждение. Анализ программного кода. Устный опрос.	ПК-2 – зув
3.2. Тема. Объектные средства моделирования детерминированных и недетерминированных объектов в парадигме системной динамики.	4	2	6/2И		6	Самостоятельное изучение учебной и научной литературы. Подготовка к лабораторному занятию.	Беседа – обсуждение. Анализ программного кода. Устный опрос.	ПК-2 – зув

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в акад. часах)			Самостоятельная работа (в акад. часах)	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код и структурный элемент компетенции
		лекции	лаборат. занятия	практич. занятия				
						Выполнение лабораторной работы.		
<b>Итого по разделу</b>	<b>4</b>	<b>4</b>	<b>12/2И</b>		<b>12</b>		<b>Проверка индивидуальных заданий</b>	
4. Раздел. Объектные средства многоподходного моделирования.	4					Самостоятельное изучение учебной и научной литературы. Подготовка к лабораторному занятию. Выполнение лабораторной работы.	Беседа – обсуждение. Анализ программного кода. Устный опрос.	
4.1. Тема. Объектные средства моделирования детерминированных и недетерминированных объектов в многоподходной парадигме.	8	5	10/10И		18,15	Самостоятельное изучение учебной и научной литературы. Подготовка к лабораторному занятию. Выполнение лабораторной работы.	Беседа – обсуждение. Анализ программного кода. Устный опрос.	
<b>Итого по разделу</b>		<b>5</b>	<b>10/8И</b>		<b>18,15</b>		<b>Проверка индивидуальных заданий</b>	
<b>Итого за семестр</b>		<b>17</b>	<b>34/14И</b>		<b>54,15</b>		<b>Экзамен</b>	
<b>Итого по дисциплине</b>		<b>17</b>	<b>34/14И</b>		<b>54,15</b>			



## 5 Образовательные и информационные технологии

1. **Традиционные образовательные технологии** ориентируются на организацию образовательного процесса, предполагающую прямую трансляцию знаний от преподавателя к студенту (преимущественно на основе объяснительно-иллюстративных методов обучения). Учебная деятельность студента носит в таких условиях, как правило, репродуктивный характер.

### **Формы учебных занятий с использованием традиционных технологий:**

Информационная лекция – последовательное изложение материала в дисциплинарной логике, осуществляемое преимущественно вербальными средствами (монолог преподавателя).

Семинар – беседа преподавателя и студентов, обсуждение заранее подготовленных сообщений по каждому вопросу плана занятия с единым для всех перечнем рекомендуемой обязательной и дополнительной литературы.

Практическое занятие, посвященное освоению конкретных умений и навыков по предложенному алгоритму.

Лабораторная работа – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.

2. **Технологии проблемного обучения** – организация образовательного процесса, которая предполагает постановку проблемных вопросов, создание учебных проблемных ситуаций для стимулирования активной познавательной деятельности студентов.

3. **Интерактивные технологии** – организация образовательного процесса, которая предполагает активное и нелинейное взаимодействие всех участников, достижение на этой основе лично значимого для них образовательного результата. Наряду со специализированными технологиями такого рода принцип интерактивности прослеживается в большинстве современных образовательных технологий. Интерактивность подразумевает субъект - субъектные отношения в ходе образовательного процесса и, как следствие, формирование саморазвивающейся информационно-ресурсной среды.

### **Формы учебных занятий с использованием специализированных интерактивных технологий:**

Лекция «обратной связи» – лекция–провокация (изложение материала с заранее запланированными ошибками), лекция-беседа, лекция-дискуссия, лекция–пресс-конференция.

4. **Информационно-коммуникационные образовательные технологии** – организация образовательного процесса, основанная на применении специализированных программных сред и технических средств работы с информацией.

## 6 Учебно-методическое обеспечение самостоятельной работы обучающихся

### **Задание к лабораторной работе по теме:**

#### **Понятие о моделях. Виды моделей. Реализация моделей в объектной парадигме.**

Создать и реализовать объектные предложенные модели средствами C++/C#/Python.

1. Объект - трансформер движется по синусоиде. Когда ордината центра объекта неотрицательна, он имеет форму круга, в противном случае — форму квадрата. Радиус круга и сторона квадрата задаются при запуске программы.
2. Два объекта, имеющих форму круга, двигаются по синусоиде и косинусоиде. Движение они начинают одновременно. При встрече они приветствуют друг друга звуковым сигналом.
3. Объект, имеющий форму круга, движется по косинусоиде. При пересечении центром объекта оси абсцисс, цвет заливки меняется. Возможные цвета заливки задаются при запуске программы. Очередной цвет задается случайным образом.
4. Объект движется по окружности с центром в начале координат. Когда центр объекта находится в первой координатной четверти, объект имеет форму квадрата,

когда объект находится во второй координатной четверти, он имеет форму треугольника, когда объект находится в третьей координатной четверти, он имеет форму трапеции, когда объект находится в четвертой координатной четверти, он имеет форму треугольника. Параметры геометрических фигур задаются при запуске программы.

5. Геометрический объект, имеющий форму круга, случайным образом перемещается по квадратному полю. После каждого перемещения он меняет цвет заливки. Возможные цвета заливки задаются при запуске программы. Очередной цвет задается случайным образом.
6. Два геометрических объекта случайным образом перемещаются внутри квадратного поля. При встрече они приветствуют друг друга звуковым сигналом.
7. Геометрический объект имеет форму стрелы с наконечником. Начальная точка стрелы находится в начале координат. Стрела совершает повороты относительно начальной точки на случайно заданный угол. После каждого поворота цвет стрелы меняется. Возможные цвета заливки задаются при запуске программы. Очередной цвет задается случайным образом.
8. Геометрический объект имеет форму стрелы с наконечником. Стрела совершает повороты относительно начальной точки на случайно заданный угол. Если конечная точка стрелы находится в первой координатной четверти, цвет наконечника синий, если конечная точка стрелы находится во второй координатной четверти, цвет наконечника зеленый, если конечная точка стрелы находится в третьей координатной четверти, цвет наконечника малиновый, конечная точка стрелы находится в четвертой координатной четверти, цвет наконечника желтый. Цвет самой стрелы всегда голубой.
9. Геометрический объект имеет форму квадрата с центром в начале координат. Он подвергается преобразованиям растяжения/сжатия по обеим координатным осям. После каждого преобразования цвет объекта меняется. Возможные цвета заливки задаются при запуске программы. Очередной цвет задается случайным образом.
10. Два круглых геометрических объекта движутся по концентрическим окружностям с центром в начале координат. Скорость движения у объектов разная. Когда их центры оказываются на одном луче, исходящем из начала координат, они приветствуют друг друга звуковым сигналом.

#### **Задание к лабораторной работе по теме:**

#### **Инструментальные средства объектного моделирование. Моделирование в системе Anylogic**

Реализовать модели из предыдущей лабораторной работы в системе Anylogic.

#### **Задание к лабораторной работе по теме:**

#### **Понятие об агенте и мультиагентной системе и их объектной реализации**

Реализовать иерархию наследования средствами Anylogic:

1. Транспортное средство (абстрактное). Наследники: транспортное средство, использующее живую силу, механическое транспортное средство. Наследники транспортного средства, использующего живую силу: животное для верховой езды, упряжка.
2. Плоды (абстрактные). Наследники: плоды съедобные, плоды несъедобные. Наследники съедобных плодов: яблоки, огурцы.
3. Аудиоаппаратура (абстрактная). Наследники: проигрыватели, магнитофоны. Наследники проигрывателей: проигрыватели цифровых записей, проигрыватели аналоговых записей.
4. Животные (абстрактные). Наследники: хищные животные, травоядные животные. Наследники травоядных животных: лошади, овцы.
5. Верхняя одежда абстрактная. Наследники: пальто, куртки. Наследники курток: куртки меховые, ветровки.

6. Обувь абстрактная. Наследники: ботинки, туфли. Наследники туфель: кроссовки, балетки.
7. Птицы абстрактные. Наследники: птицы летающие, птицы нелетающие. Наследники птиц летающих: воробьи, синицы.
8. Пища абстрактная. Наследники: блюда холодные, блюда горячие. Наследники блюд горячих: борщ, куллама.
9. Литературный текст абстрактный. Наследники: прозаический текст, поэтический текст. Наследники поэтического текста: стихи, написанные ямбом; стихи, написанные амфибрахием.
10. Средства связи абстрактные. Наследники: средства связи стационарные, средства связи мобильные. Наследники средств связи мобильных: автономные приемно-передающие устройства; средства сотовой связи.

**Задание к лабораторной работе по теме:**

**Объектные средства агентного моделирования детерминированных и недетерминированных объектов.**

Реализовать систему агентов средствами Anylogic:

1. На верхнем уровне классы: «средство передвижения», «загрязнитель окружающей среды». Создать класс «автомобиль» — наследник обоих классов верхнего уровня.
2. На верхнем уровне классы: «средство мобильной связи», «показатель статуса». Создать класс «смартфон для претенциозной личности» — наследник обоих классов верхнего уровня.
3. На верхнем уровне классы «средство обработки информации», «источник опасности для зрения». Создать класс «персональный компьютер» — наследник обоих классов верхнего уровня.
4. На верхнем уровне классы «устройство для приготовления пищи», «источник пожарной опасности». Создать класс «электрическая плита» — наследник обоих классов верхнего уровня.
5. На верхнем уровне классы: «транспортное средство», «налогооблагаемое имущество». Создать класс «автомобиль» — наследник обоих классов верхнего уровня.
6. На верхнем уровне классы «аудиоаппаратура», «источник шумового загрязнения окружающей среды». Создать класс «аудиосистема» — наследник обоих классов верхнего уровня.
7. На верхнем уровне классы «электронное устройство», «бытовая техника». Создать класс «ноутбук» — наследник обоих классов верхнего уровня.
8. На верхнем уровне классы «средство связи», «рекламируемый объект». Создать класс «смартфон» — наследник обоих классов верхнего уровня.
9. На верхнем уровне классы «средство регистрации изображений», «товар, на который дается гарантия». Создать класс «фотоаппарат» — наследник обоих классов верхнего уровня.
10. На верхнем уровне классы «растение», «объект для украшения интерьера». Создать класс «комнатное растение» — наследник обоих классов верхнего уровня.

**Задание к лабораторной работе по теме:**

**Объектные средства дискретно-событийного моделирования детерминированных и недетерминированных объектов**

Выполнить агентное моделирование неформально поставленной задачи:

1. Младенец умеет: есть, пить, плакать, когда голоден, когда хочет пить, плакать, когда надо сменить пеленки, Кроме того, он может плакать из солидарности с другим младенцем (когда слышит плач более 3 минут), плакать просто так. Если младенец плачет более 5 минут, пеленки обязательно будут мокрые. Младенец начинает плакать в случайный момент времени. Мать умеет: кормить младенца, поить младенца, менять ему пеленки, безошибочно определять причину плача,

- успокаивать одновременно несколько младенцев. Метода стирки у нее нет, поэтому при нехватке пеленок ей придется использовать в этом качестве простыни, наволочки и все, что под руку попадется. Отец умеет: стирать и гладить пеленки. Делать он это может только 1 раз в сутки.
2. Кошка умеет: охотиться на мышей (вероятность поимки мыши —  $p$ ), воровать у хозяина мясо (вероятность успеха —  $q$ ), воровать у хозяина молоко (вероятность успеха —  $r$ ). Мышь умеет: прятаться от кошки, прятаться от хозяина, воровать у хозяина муку (вероятность успеха —  $s$ ), воровать у хозяина крупу (вероятность успеха —  $t$ ). Хозяин умеет: отгонять кошку от еды, ловить мышь в капкан, покупать все перечисленные ранее продукты. В произвольные моменты времени хозяин отлучается из дому.
  3. Крестьянин выращивает в поле зерно. Птицы поедают зерно ( $m$  граммов в день). Кроме того, они поедают жучков ( $n$  жучков в день). Жучки поедают зерно ( $s$  граммов в день). Если крестьянин уничтожит или прогонит всех птиц, жучки, не имея естественных врагов, будут беспрепятственно съедать зерно. Вероятность для крестьянина прогнать птицу —  $p$ , вероятность для птицы съесть жучка —  $q$ .
  4. Медицинская страховая компания работает с  $N$  клиентами. За каждого клиента она получает ежемесячно  $w_i$  ( $i = 1, \dots, N$ ) рублей. Вероятность заболеть в течение года для каждого клиента равна  $p_i$  ( $i = 1, \dots, N$ ). Стоимость его излечения является равномерно распределенной случайной величиной в диапазоне от 1000 до 100000 рублей.
  5. Процессинговый центр обслуживает расчеты по дебетовым карточкам. Существует вероятность  $p$  попытки получения суммы  $S$  неправомерным путем. С вероятностью  $q$  эта попытка будет пресечена, а злоумышленник наказан, в противном случае процессинговый центр заплатит банку штраф в размере  $2S$ .
  6. Системный администратор компьютерной сети умеет определять пользователям права доступа, изымать вредные записи, брать плату за работу в сети, лишать нарушителей прав доступа, брать штрафы за нарушения. Добросовестный пользователь может вносить записи в отведенный ему раздел, стирать свои записи, платить за пользование, извлекать доходы или нести убытки из-за не востребо-ванности информации или деятельности хакеров. За 1 килобайт пользователь платит  $K$  рублей, использование 1 килобайта информации приносит  $N$  рублей дохода, если информация окажется востребованной (вероятность —  $p$ ). Хакер может все то, что и добросовестный пользователь. Кроме того, он может с вероятностью  $q$  определять чужие пароли и переадресовывать счета за работу в сети другим пользователям (как добросовестным, так и хакерам). В случае обнаружения недобросовестного поведения, он может быть подвергнут штрафу в размере десятикратно превышающем доход от недобросовестной деятельности.
  7. Спамер рассылает по сети Internet недобросовестную рекламу товара (услуги) ценой в  $G$  рублей. С вероятностью  $p_i$   $i$ -й пользователь купит рекламируемый товар (или услугу),  $q_i$  — не отреагирует на спам, с вероятностью  $r_i$  заставит спамера заплатить штраф, величина которого указывается далее. За каждую покупку спамер получает 1% комиссионных, при изобличении платит штраф, в 10 раз превышающий размер потенциальных комиссионных (то есть комиссионных, которые бы он получил при условии, что все адресаты купят рекламируемый товар).
  8. Дистрибьютор высококачественной компьютерной техники работает с сетью дилеров. Он покупает у производителей продукцию  $n$  наименований по цене  $k_i$  за единицу и поставляет ее дилерам с наценкой в  $p\%$ . Добросовестные дилеры торгуют только фирменным товаром с оговоренной в соглашении с дистрибьютором

наценкой в  $q\%$ . Недобросовестные дилеры могут поступать так же, как и добросовестные. Кроме того, они могут под видом фирменных продавать компьютеры низкого качества с наценкой  $r\% > q\%$ . В случае обнаружения подлога недобросовестный дилер возмещает обманутым клиентам разницу в цене и платит штраф в размере утроенной неправомерно полученной суммы. Покупатель не имеет возможности отличить добросовестного дилера от недобросовестного. Количество денег у покупателя задается случайным образом. Решение о покупке принимается случайным образом при наличии у покупателя достаточной суммы свободных денег

9. В библиотеке имеются книги  $N$  наименований. Количество экземпляров каждой из этих книг  $k_i (i = 1, \dots, N)$ . В библиотеку записано  $M$  читателей. Каждый из читателей может одновременно держать у себя не более  $L$  книг. Книги выдаются на срок до 30 дней. Считаем, что читатель с одинаковой вероятностью может запросить любую из книг. Если запрошенной книги нет в наличии, читателя ставят в очередь на ее получение. Когда читатель возвращает книгу, она достается первому в очереди. Если читатель возвращает книгу после указанного срока, он лишается права получить книги в течение 60 дней. Длительность нахождения книги у читателя является случайной величиной.
10. Фирма распространяет лицензионное программное обеспечение и занимается его сопровождением. За каждую инсталляцию программного обеспечения фирма получает  $k$  рублей прибыли. В случае возникновения проблем с использованием установленного программного обеспечения фирма должна в течение 2 часов с момента получения заявки направить специалиста для решения проблемы. Вероятность возникновения в течение суток проблемы, требующей вмешательства специалиста, равна  $p$ . Специалист устраняет проблему за время, являющееся случайной величиной, равномерно распределенной в интервале  $[0, T]$ . В том случае, когда специалист опаздывает, фирма платит штраф в размере  $r$  рублей за каждый час опоздания. Каждый специалист получает  $s$  рублей в месяц

#### **Задание к лабораторной работе по теме:**

#### **Объектные средства моделирования детерминированных и недетерминированных объектов в парадигме системной динамики.**

Выполнить дискретно-событийное моделирование следующих задач.

1. Снежинка имеет форму правильного многоугольника. Падает вниз. Под воздействием ветра может лететь вверх или в сторону. Может слипаться с другими снежинками. Комок снежинок имеет форму шара. Снежинка может таять. Изобразить снегопад, метель оттепель.
2. Схематично изображенный самолет может стоять на стоянке, вырывать по рулежным дорожкам на взлетно-посадочную полосу, разогнаться и взлетать, лететь по прямой траектории, снижаться, совершать посадку и уходить по рулежным дорожкам с взлетно-посадочной полосы на стоянку.
3. Схематично изображенный вертолет может стоять на стоянке, вертикально взлетать, лететь по прямой траектории, к нему может быть прицеплен груз на внешней подвеске. Вертолет может перевозить груз на внешней подвеске, устанавливать груз на указанное место, приземляться.
4. Заготовка проходит между двумя вращающимися валками, при этом уменьшается ее высота (обжатие), увеличивается ширина (уширение) и длина. Прошедшую между валками заготовку поворачивают на  $90^\circ$  (кантовка) и направляют в обратном направлении. Процесс продолжается до тех пор, пока не будут достигнуты требуемые параметры заготовки.
5. Схематично изображенный грузовик может привозить груз к подъемному крану и отвозить груз от подъемного крана. Схематично изображенный подъемный

- кран разгружает и загружает грузовики.
6. Схематично изображенный надувной шар может быть сплюснутым, его можно надувать (его радиус увеличивается), при излишнем надувании он может лопнуть, он может лопнуть, если его проткнуть иголкой. Шары можно соединять в гирлянду, отделять от гирлянды.
  7. Схематично изображенная ветряная мельница вращает лопасти. Если ветра нет, лопасти неподвижны. При очень сильном ветре лопасти отваливаются.
  8. Схематично изображенный катер может стоять у пристани, в катер может быть помещен груз. Катер может отчаливать, двигаться по фарватеру, причаливать, проводить выгрузку.
  9. Схематично изображенный трамвай может стоять в депо, ехать от остановки до остановки по прямой и с поворотами, останавливаться на остановках, разворачиваться на конечных остановках.
  10. Схематично изображенный автомобиль может стоять у тротуара, двигаться по прямой, перестраиваться из ряда в ряд, поворачивать. При перестроениях и поворотах у автомобиля должен быть включен соответствующий световой указатель. При нарушении правил автомобиль может быть удален из транспортного потока. Светофор поочередно включает в каждом направлении красный, желтый и зеленый свет

#### **Задание к лабораторной работе по теме:**

#### **Объектные средства моделирования детерминированных и недетерминированных объектов в многоподходной парадигме**

Выполнить моделирование в многоподходной парадигме следующих задач:

1. Банк может: привлекать вклады (под  $p\%$  годовых), давать кредиты (под  $q\%$  годовых). Существует вероятность  $s$  того, что банк не сможет расплатиться с вкладчиком. Вкладчик умеет: зарабатывать деньги, вносить вклады, получать по вкладам проценты. Должник умеет зарабатывать деньги, обслуживать долг (выплачивать проценты), возвращать долг. Существует вероятность  $r$  того, что должник окажется неплатежеспособным.
2. Экстрасенс умеет: морочить голову пациентам, вымогать у них деньги. Врач умеет: вылечивать пациентов (с вероятностью  $p$ ). Пациент умеет: зарабатывать деньги на лечение у врача или экстрасенса, принимать решение, у кого лечиться — врача или экстрасенса, погибать от несвоевременного лечения. Зарубок пациента задается случайным образом. Случайным образом задается решение о том, у кого лечиться
3. Производитель программного обеспечения (ПО) позиционирует свою продукцию в классе ShareWare. Затраты на выпуск ПО составляют  $\$U$ . Получив его продукцию, пользователь заплатит  $\$t$  с вероятностью  $p$ .
4. Провайдер Интернет предлагает пользователям  $k$  тарифных планов. Для привлечения новых пользователей он использует звонки потенциальным покупателям услуги по телефону и обход их жилищ. Вероятность приобретения тарифного плана после звонка по телефону равна  $p_i, i = 1, \dots, k$ , вероятность приобретения тарифного плана после посещения жилища равна  $q_i, i = 1, \dots, k$ . Себестоимость привлечения одного клиента с использованием телефона равна  $v$ , с использованием обхода жилищ —  $w$ . Прибыль от одного клиента, пользующегося услугами провайдера равна  $t_i, i = 1, \dots, k$ .
5. Продовольственный магазин торгует товарами  $N$  наименований. У каждого товара есть оптовая цена за единицу  $b_i$ , розничная цена за единицу  $c_i$ , минимальный объем оптовой закупки  $k_i$ , срок реализации  $t_i$  (езде  $i = 1, \dots, N$ ). Если товар не реализован своевременно, он подлежит списанию. Ежедневно магазин посе-

- щает  $M$  покупателей. Каждый из них с вероятностью  $p_i, i = 1, \dots, N$  покупает один из товаров, причем величина покупки также является случайной. Покупка может быть совершена только при наличии товара в магазине. В магазине учитываются покупки, потери от списания непроданного товара, а также недополученная прибыль от неудовлетворенного спроса.
6. Открывается компьютерная школа. В ней желающие будут изучать Windows и Microsoft Office. За изучение Windows ученик будет платить  $K$  рублей, за изучение Microsoft Office —  $L$  рублей. В штате состоят два преподавателя, для проведения экзаменов привлекается независимый эксперт. Имеется  $K$  людей, желающих получить компьютерное образование. Вероятность того, что они пойдут учиться в эту школу равна  $p$ . Преподаватели получают за работу по  $m$  рублей (независимо от количества учеников). Эксперту платят по  $n$  рублей с каждого экзаменуемого.
  7. Компьютерный хитрец добывает в Интернете рефераты для ленивых студентов. Нахождение одного реферата обходится хитрецу в  $l$  рублей. Студент платит за один реферат  $m$  рублей. Если реферат не будет зачтен, студент получает от хитреца  $k$  рублей ( $k < m$ ). Вероятность сдать реферат равна  $p$ . Имеется  $R$  ленивых студентов и  $S$  компьютерных хитрецов. Каждому ленивому студенту может потребоваться от 0 до 10 рефератов. Количество рефератов — равномерно распределенная случайная величина.
  8. Сетевой торговец ходит с мешком товара по квартирам. Цена единицы товара —  $s$  рублей. С вероятностью  $p$  ему не откроют дверь. Если ему дверь откроют, с вероятностью  $q$  он сможет продать одну единицу товара. Всего торговцев —  $N$ , потенциальных покупателей —  $K$ . Если покупатель что-либо уже купил у одного торговца, то никому из них он в течение 10 дней не откроет дверь. Для продолжения работы торговец» должен продавать в день товара на сумму не менее  $t$  рублей. Если он в течение 10 дней не продает товара на эту сумму, ему приходится прекращать свою торговую деятельность.
  9. Инвестиционный строительный фонд пирамидальной конструкции покупает своим старым пайщикам жилье за счет части взносов новых пайщиков. Так продолжается до тех пор, пока имеется достаточное количество новых пайщиков. Когда их становится недостаточно, фонд объявляет о банкротстве. Цена жилища —  $S$  рублей, суммарный взнос пайщика —  $R$  рублей, на покупку жилья фонд направляет  $f\%$  денежных поступлений. В городе имеется  $K$  жителей, нуждающихся в жилье и имеющих достаточно средств. Вероятность привлечь в фонд одного жителя —  $p$ . Каждый человек вступает только один раз только в один фонд. Пусть количество фондов —  $N$ .
  10. Провайдер кабельного телевидения предлагает пользователям  $k$  пакетов. Для привлечения новых пользователей он использует рекламу в социальных сетях, реклама в эфирных каналах телевидения. Вероятность приобретения пакета в результате воздействия рекламы в социальных сетях равна  $p_i, i = 1, \dots, k$ , вероятность приобретения тарифного плана после воздействия рекламы в эфирных каналах телевидения равна  $q_i, i = 1, \dots, k$ . Себестоимость привлечения одного клиента с использованием рекламы в социальных сетях  $v$ , с использованием рекламы в эфирных каналах телевидения —  $w$ . Прибыль от одного клиента, пользующегося услугами провайдера равна  $t_i, i = 1, \dots, k$ .

### Индивидуальные задания к разделу 1.

В предложенных фрагментах программ средствами языка Python описаны системы взаимодействующих объектов. Создайте аналог средствами Anylogic.

```

1.
class Borg:
# __shared_state ={"1":"2"}
  my__shared_state ={"1":"2"}
  def __init__(self):
    self.x=7
    self.__dict__=self.my__shared_state
    pass
  def get_shared(self):
    return self.my__shared_state

b1,b2=Borg(),Borg()
print(b1.get_shared())
print(b2.get_shared())
b1.x=4
b2.y=9
p1 = b1 is b2
print("Borg object 'b1': ",b1)
print("Borg object 'b2': ",b2)
print("Object state 'b1' :",b1.__dict__)
print("Object state 'b2' :",b2.__dict__)
print(b1.get_shared())
print(b2.get_shared())
print(p1);

2.
from abc import ABCMeta, abstractmethod

class PizzaFactory(metaclass = ABCMeta):
  @abstractmethod
  def createVegPizza(self):
    pass

  @abstractmethod
  def createNonVegPizza(self):
    pass;

3.
class IndianPizzaFactory(PizzaFactory):

  def createVegPizza(self):
    return DeluxVeggiePizza()

  def createNonVegPizza(self):
    return ChickenPizza()

4.
class USPizzaFactory(PizzaFactory):

  def createVegPizza(self):
    return MexicanVegPizza()

```



```
def createNonVegPizza(self):
    return HamPizza()
```

5.

```
class VegPizza(metaclass = ABCMeta):
    @abstractmethod
    def prepare(self, VegPizza):
        pass
```

6.

```
class NonVegPizza(metaclass = ABCMeta):
    @abstractmethod
    def serve(self, VegPizza):
        pass
```

7.

```
class DeluxVeggiePizza(VegPizza):
    def prepare(self):
        print("Prepare ",type(self).__name__)
```

8.

```
class ChickenPizza(NonVegPizza):
    def serve(self, VegPizza):
        print(type(self).__name__," is served with Chicken on ",
              type(VegPizza).__name__)
```

9.

```
class MexicanVegPizza(VegPizza):
    def prepare(self):
        print("Prepare ",type(self).__name__)
```

10.

```
def makePizzas(self):
    for factory in [IndianPizzaFactory(),USPizzaFactory()]:
        self.factory = factory
        self.NonVegPizza=self.factory.createNonVegPizza()
        self.VegPizza=self.factory.createVegPizza()
        self.VegPizza.prepare()
        self.NonVegPizza.serve(self.VegPizza)
```

### **Индивидуальные задания к разделу 2.**

В предложенных фрагментах программ средствами языка Python описаны системы взаимодействующих объектов — экземпляров классов. Создайте аналог средствами Anylogic. Класс aPerson должен быть рассмотрен всеми студентами. Недостающие классы следует создать самостоятельно.

```
class aPerson:
```

```

def __init__(self,first,last,age):
    self.__firstname,self.__lastname,self.__age = first,last,age

def __str__(self):
    return self.__firstname + '+' + self.__lastname + ' ' + str(self.__age)

```

1.

```

from aPerson import *
class anEmployee(aPerson):
    def __init__(self,first,last,age,staffnum):
        super().__init__(first,last,age)
        self.__staffnumber = staffnum

    def __str__(self):
        return super().__str__() + ', ' + self.__staffnumber

```

2.

```

class A:
    def __init__(self):
        print('A')

class B(A):
    def __init__(self):
        print('B')
        super().__init__()

class C(A):
    def __init__(self):
        print('C')
        super().__init__()

class D(B,C):
    def __init__(self):
        print('D')
        super().__init__()

```

d=D()

3.

```

class aPerson:
    def __init__(self,first,last,age):
        self.__firstname,self.__lastname,self.__age = first,last,age

    def __str__(self):
        return self.__firstname + '+' + self.__lastname + ' ' + str(self.__age)

from aPerson import *
def Start():
    St1 = aPerson('Oleg','Kotov',20)
    print(St1)

```

4.

```

class Length:
    __metric = {'mm': 0.001, 'cm': 0.01, 'm': 1, 'km': 1000,
                'in': 0.0254, 'ft': 0.3048, 'yd': 0.9144, 'mi': 1609.344}

    def __init__(self, value, unit = 'm'):
        self.value = value
        self.unit = unit

```

5.

```

def Converse2Metres(self):
    return self.value * Length.__metric[self.unit]

def __add__(self, other):
    if type(other) == int or type(other) == float:
        l = self.Converse2Metres() + other
    else:
        l = self.Converse2Metres() + other.Converse2Metres()
    return Length(l /
                  Length.__metric[self.unit], self.unit)

```

6.

```

def __str__(self):
    return str(self.Converse2Metres())

def __repr__(self):
    return "Length("+str(self.value)+", "+self.unit+""")

def __iadd__(self, other):
    if type(other) == int or type(other) == float:
        l = self.Converse2Metres() + other
    else:
        l = self.Converse2Metres() + other.Converse2Metres()
    self.value = l / Length.__metric[self.unit]
    return self

```

7.

```

from person import *
class Employee(Person):
    def __init__(self, first, last, age, staffnum):
        super().__init__(self, first, last, age)
        self.__staffnumber = staffnum
    def __str__(self):
        return super().__str__() + ' '+self.__staffnumber

```

8.

```

class S(object):
    def __init__(self, v):
        self.val = v

```

```
x=S(42)
print(x.val)
x.new = 77
print(x.new)
print(x.__dir__())
```

9.

```
def dict_01():
    """Dictionary 1"""
    city_population = {'Bratislava': 410000, "Brno" : 430000,
    'Warszawa': 1500000, "Praha" : 1100000, "Kyiv" : 220000}
    N_Kyiv = city_population['Kyiv']
    print(N_Kyiv)
    city_population["Kharkiv"]=1400000
    print(city_population)
    N_Warszawa = city_population["Warszawa"]
    print(N_Warszawa)
    print(city_population)
```

10.

```
def dict_02():
    """Dictionary 2"""
    city = {}
    tc = type(city)
    print(city,tc,sep=' &&& ')
    city['Ufa'],tc = 1100000, type(city)
    print(city,tc,sep=' @@@ ')
```

### Индивидуальные задания к разделу 3.

В предложенных фрагментах программ средствами языка Python описаны системы взаимодействующих объектов, моделирующих функционирование агентной системы. Создайте аналог средствами Anylogic.

1.

```
class EventManager(object):
    def __init__(self):
        print("Event Manager:: Let me talk to the folks\n")

    def arrange(self):
        self.hotelier = Hotelier()
        self.hotelier.bookHotel()

        self.florist = Florist()
        self.florist.setFlowerRequirements()

        self.caterer = Caterer()
        self.caterer.setCuisine()

        self.musician = Musician()
        self.musician.setMusicType()
```

2.

```
class Hotelier(object):
```

```

def __init__(self):
    print("Arranging the Hotel for Marriage? ")

def __isAvailable(self):
    print("Is the Hotel free for the event on given day?")
    return True

def bookHotel(self):
    if self.__isAvailable():
        print("Registered the Booking\n\n")

```

3.

```

class Florist(object):
    def __init__(self):
        print("Flower Decorations for the Event? ")

    def setFlowerRequirements(self):
        print("""Carnations, Roses and Lillies would be used for
Decorations\n\n""")

```

4.

```

class Caterer(object):
    def __init__(self):
        print("Food Aragements for the Event ")

    def setCuisine(self):
        print("Chinese & Continental Cuisine to be served\n\n")

```

5.

```

class Musician(object):
    def __init__(self):
        print("Musical Arrangement for Marriage ")

    def setMusicType(self):
        print("Jazz and Classical will be played\n\n")

```

6.

```

class You(object):
    def __init__(self):
        print("You:: Whoa! Marriage Arragements??!!!")

    def askEventManager(self):
        print("Let's Contact the Event Manager\n\n")
        em = EventManager()
        em.arrange()

    def __del__(self):
        print("You:: Thanks to Event Manager, all preparations done!")

```

7.

```

class Actor(object):
    def __init__(self):
        self.isBusy = False

    def occupied(self):
        self.isBusy = True
        print(type(self).__name__,"is occupied with current movie")

    def available(self):
        self.isBusy = False
        print(type(self).__name__,"is free for the movie")

    def getStatus(self):
        return self.isBusy

```

8.

```

class Agent(object):
    def __init__(self):
        self.principal = None

    def work(self):
        self.actor = Actor()
        if self.actor.getStatus():
            self.actor.occupied
        else:
            self.actor.available()

```

9.

```

class Payment(metaclass = ABCMeta):
    @abstractmethod
    def do_pay(self):
        pass

class Bank(Payment):
    def __init__(self):
        self.card,self.account = None,None

    def __getAccount(self):
        self.account = self.card
        return self.account

    def __hasFunds(self):
        print("Bank:: Cheking if Account",self.__getAccount(),
              "has enough funds")
        return True

    def setCard(self,card):
        self.card = card

    def do_pay(self):
        if self.__hasFunds():
            print("Bank:: Paying the merchant")
            return True

```

```

else:
    print("Bank:: Sorry, not enough funds!")
    return False

class DebitCard(Payment):
    def __init__(self):
        self.bank = Bank()

    def do_pay(self):
        card = input("Proxy:: Punch in Card Number: ")
        self.bank.setCard(card)
        return self.bank.do_pay()

```

10.

```

class You:
    def __init__(self):
        print("You:: Let's buy the Denim shirt")
        self.debitcard = DebitCard()
        self.isPurchased = None

    def make_payment(self):
        self.isPurchased = self.debitcard.do_pay()

    def __del__(self):
        if self.isPurchased:
            print("You:: Wow! Denim shirt is Mine: ")
        else:
            print("You:: I should earn more :(")

```

#### **Индивидуальные задания к разделу 4.**

Создайте для предложенных заданий многоподходные модели средствами Anylogic.

##### **Вариант 1. Информационная система ВУЗа**

Студенты, организованные в группы, учатся на одном из факультетов ВУЗа.

В учебном процессе участвуют преподаватели кафедр, административно относящихся к одному из факультетов. Преподаватели подразделяются на следующие категории: ассистенты, преподаватели, старшие преподаватели, доценты, профессора.

Учебный процесс регламентируется учебным планом. Учебный план содержит перечень учебных дисциплин с указанием курса и семестра для студентов каждого года набора, количества часов на каждый вид занятий по дисциплине (виды занятий: лекции, семинары, лабораторные работы, консультации, курсовые работы, ИР и т.д.) и формы контроля (зачет, экзамен). Перед началом учебного семестра деканаты раздают на кафедры учебные поручения, в которых указывается, какие кафедры (не обязательно относящиеся к данному факультету) какие дисциплины и для каких групп должны

вести в очередном семестре. Преподаватель может вести занятия по одной или нескольким дисциплинам для студентов как своего, так и других факультетов. Сведения о проведенных экзаменах и зачетах собираются деканатом.

Обязательные функции информационной системы:

1. Получить перечень студентов указанных групп либо указанного курса факультета (по половому признаку, году рождения, возрасту, признаку наличия детей, по признаку получения и размеру стипендии).
2. Получить список преподавателей указанных кафедр либо указанного факультета полностью либо указанных категорий (ассистенты, доценты, профессора и т.д.) по половому признаку, году рождения, возрасту, признаку наличия и количеству детей.
3. Получить перечень кафедр, проводящих занятия в указанной группе либо на указанном курсе указанного факультета в указанном семестре, либо за указанный период.
4. Получить список и общее число преподавателей, проводивших (проводящих) занятия по указанной дисциплине в указанной группе либо на указанном курсе указанного факультета.
5. Получить перечень преподавателей, проводивших (проводящих) лекционные, семинарские и другие виды занятий в указанной группе либо на указанном курсе указанного факультета в указанном семестре (за указанный период).
6. Получить список и общее число студентов указанных групп, сдавших зачет либо экзамен по указанной дисциплине с указанной оценкой.
7. Получить список и общее число студентов указанных групп или указанного курса указанного факультета, сдавших указанную сессию на отлично, без троек, без двоек.
8. Получить перечень преподавателей, принимающих (принимавших) экзамены в указанных группах, по указанным дисциплинам, в указанном семестре.

**Вариант 2. Информационная система проектной организации**

Проектная организация представлена следующими категориями сотрудников: конструкторы, инженеры, техники, лаборанты, прочий обслуживающий персонал.

Сотрудники разделены на отделы, руководимые начальником. Каждый сотрудник числится только в одном отделе.

Проектная организация заключает договоры с заказчиками на выполнение проектов.

По одному договору может выполняться более одного проекта, и один проект может выполняться для нескольких договоров. Суммарная стоимость договора определяется стоимостью всех проектных работ, выполняемых для этого договора. Каждый договор и проект имеет руководителя и группу сотрудников, выполняющих этот договор или проект, причем это могут быть сотрудники не только одного отдела.

Ведется учет кадров, учет выполнения договоров и проектов, стоимостной учет всех выполненных работ.

Обязательные функции информационной системы:

1. Получить данные о составе указанного отдела или всей организации (по указанной категории сотрудников, по возрастному составу).
2. Получить перечень руководителей отделов.
3. Получить перечень договоров или проектов, выполняемых в данный момент или в указанный период времени.
4. Получить информацию о том, какие проекты выполняются (выполнялись) в рамках указанного договора и какие договоры поддерживаются указанными проектами.



5. Получить данные о стоимости выполненных договоров (проектов) в течение указанного периода времени.
6. Получить сведения об участии указанного сотрудника или категории сотрудников в проектах (договорах) за определенный период времени.
7. Получить данные о численности и составе сотрудников, участвующих в указанном проекте (в целом и по отдельным категориям).
8. Получить сведения об эффективности договоров (стоимость договоров, соотношенная с затраченным временем или стоимость с учетом привлеченных людских ресурсов).
9. Получить сведения об эффективности проектов (стоимость проектов, соотношенная с затраченным временем или стоимость с учетом привлеченных людских ресурсов).

### **Вариант 3. Информационная система авиастроительного предприятия**

Структурно предприятие разбито на цеха, которые в свою очередь подразделяются на участки. Выпускаемые изделия предприятия - самолеты (гражданские, транспортные, военные), планеры, вертолеты, дельтапланы, ракеты (артиллерийские, авиационные, военно-морские), прочие изделия. По каждой категории изделий может собираться несколько видов изделий.

Сотрудники предприятия - инженерно-технический персонал (инженеры, технологи, техники) и рабочие (сборщики, токари, слесари, сварщики и пр.).

Рабочие объединяются в бригады, которыми руководят бригадиры. Бригадиры выбираются из числа рабочих, мастера, начальники участков и цехов назначаются из числа инженерно-технического персонала.

Каждое изделие собирается в своем цехе (в цехе может собираться несколько видов изделий) и в процессе изготовления проходит определенный цикл работ, перемещаясь с одного участка на другой. Все работы по сборке конкретного изделия на определенном участке выполняет одна бригада рабочих, при этом на участке может работать несколько бригад. Возглавляет работу на участке начальник участка, в подчинении которого находится несколько мастеров. Различные изделия могут проходить одни и те же циклы работ на одних и тех же участках цеха.

#### Обязательные функции информационной системы:

1. Получить перечень видов изделий, собираемых указанным цехом (в целом и отдельной категории).
2. Получить число и перечень изделий (отдельной категории и в целом), собранных указанным цехом, участком, предприятием за определенный отрезок времени.
3. Получить данные о кадровом составе цеха, предприятия в целом и по указанным категориям инженерно-технического персонала и рабочих.
4. Получить перечень участков и их начальников указанного цеха или предприятия в целом.
5. Получить перечень работ, которые проходит указанное изделие.
6. Получить состав бригад указанного участка, цеха.
7. Получить список мастеров указанного участка, цеха.
8. Получить перечень изделий отдельной категории и в целом, собираемых в настоящий момент указанным участком, цехом, предприятием.
9. Получить состав бригад, участвующих в сборке указанного изделия.
10. Получить число и перечень изделий отдельной категории и в целом, собираемых указанным цехом, участком, предприятием в целом в настоящее время.

### **Вариант 4. Информационная система библиотечного фонда города**

Библиотечный фонд города составляют библиотеки, расположенные на территории города. Каждая библиотека включает в себя абонементы и читальные залы.

Каждый читатель, будучи зарегистрированным в одной из библиотек, имеет доступ ко всему библиотечному фонду города.

Библиотечный фонд (книги, журналы, газеты, сборники статей, сборники стихов, диссертации, рефераты, сборники докладов и тезисов докладов и пр.) размещен в залах-хранилищах различных библиотек на определенных местах хранения (номер зала, стеллажа, полки) и идентифицируется номенклатурными номерами. При этом существуют различные правила относительно тех или иных изданий: какие-то подлежат только чтению в читальных залах библиотек; для тех, что выдаются, может быть установлен различный срок выдачи и т.д. С одной стороны, библиотечный фонд может пополняться, с другой, - с течением времени происходит его списание.

Произведения авторов, составляющие библиотечный фонд, также можно разделить на различные категории: учебники, повести, романы, статьи, стихи, диссертации, рефераты, тезисы докладов и т.д.

Сотрудники библиотеки, работающие в различных залах различных библиотек, ведут учет читателей, а также учет размещения и выдачи литературы.

#### Обязательные функции информационной системы:

1. Получить список читателей, на руках у которых находится указанное издание (книга, журнал и т.д).
2. Получить список литературы, которая в настоящий момент выдана с определенной полки некоторой библиотеки.
3. Получить список читателей с просроченным сроком литературы.
4. Получить перечень указанной литературы, которая поступила (была списана) в течение некоторого периода.
5. Выдать список библиотекарей, работающих в указанном читальном зале некоторой библиотеки.
6. Получить перечень читателей, которые в течение указанного промежутка времени получали издание с некоторым произведением, и название этого издания.
7. Получить список читателей, не посещавших библиотеку в течение указанного времени.
8. Выдать из библиотечного фонда указанной библиотеки список инвентарных номеров и названий произведений указанного автора.
9. Получить список самых популярных произведений.

#### **Вариант 5. Информационная система спортивных организаций города**

Спортивная инфраструктура города представлена спортивными сооружениями различного типа: спортивные залы, манежи, стадионы, корты и т.д.

Спортсмены под руководством тренеров занимаются отдельными видами спорта, при этом один и тот же спортсмен может заниматься несколькими видами спорта, и в рамках одного и того же вида спорта может тренироваться у нескольких тренеров. Все спортсмены объединяются в спортивные клубы, при этом каждый из них может выступать только за один клуб.

Организаторы соревнований проводят состязания по отдельным видам спорта на спортивных сооружениях города. По результатам участия спортсменов в соревнованиях производится награждение.

#### Обязательные функции информационной системы:

1. Получить перечень спортивных сооружений указанного типа в целом или удовлетворяющих заданным характеристикам (например, стадионы, вмещающие не менее указанного числа зрителей).

2. Получить список спортсменов, занимающихся указанным видом спорта.
3. Получить список спортсменов, тренирующихся у заданного тренера.
4. Получить список спортсменов, занимающихся более чем одним видом спорта с указанием этих видов спорта.
5. Получить список тренеров указанного спортсмена.
6. Получить перечень соревнований, проведенных в течение заданного периода времени в целом либо указанным организатором.
7. Получить список призеров указанного соревнования.
8. Получить перечень спортивных клубов и число спортсменов этих клубов, участвовавших в спортивных соревнованиях в течение заданного интервала времени.
9. Получить список тренеров по определенному виду спорта.
10. Получить список спортсменов, не участвовавших ни в каких соревнованиях в течение определенного периода времени.
11. Получить перечень спортивных сооружений и даты проведения на них соревнований в течение определенного периода времени.

### **Вариант 6. Информационная система гостиничного комплекса**

Гостиничный комплекс состоит из нескольких зданий-гостиниц (корпусов). Каждый корпус имеет ряд характеристик, таких, как класс отеля (двух-, пятизвездочные), количество этажей в здании, общее количество комнат, комнат на этаже, местность номеров (одно-, двух-, трехместные и т.д.), наличие служб быта: ежедневная уборка номера, прачечная, химчистка, питание (рестораны, бары) и развлечения (бассейн, сауна, бильярд и пр.). От типа корпуса и местности номера зависит сумма оплаты за него. Химчистка, стирка, дополнительное питание, все развлечения производятся за отдельную плату.

С крупными организациями (туристические фирмы, организации, занимающиеся проведением международных симпозиумов, конгрессов, семинаров, карнавалов и т.д.) заключаются договора, позволяющие организациям бронировать номера с большими скидками на определенное время вперед не для одного человека, а для группы людей. В брони указывается класс отеля, этаж, количество комнат. Броня может быть отменена за неделю до заселения.

Новые жильцы пополняют перечень клиентов гостиницы. Ведется учет свободных номеров, дополнительных затрат постояльцев гостиницы. Ведется учет долгов постояльца гостинице за все дополнительные услуги.

#### Обязательные функции информационной системы:

1. Получить перечень фирм, забронировавших места в объеме, не менее указанного, за весь период сотрудничества, либо за некоторый период.
2. Получить перечень постояльцев, заселявшихся в номера с указанными характеристиками за некоторый период.
3. Получить общее количество свободных номеров на данный момент.
4. Получить сведения о количестве свободных номеров с указанными характеристиками.
5. Получить список занятых номеров, которые освобождаются к указанному сроку.
6. Получить данные об объеме бронирования номеров данной фирмой за указанный период.
7. Получить сведения о фирмах, с которыми заключены договора о брони на указанный период.
8. Получить сведения о наиболее часто посещающих гостиницу постояльцах по всем корпусам гостиниц, по определенному зданию.

9. Получить сведения о новых клиентах за указанный период.
10. Получить сведения о конкретном человеке, сколько раз он посещал гостиницу, в каких номерах и в какой период останавливался, какие счета оплачивал.
11. Получить процентное отношение всех номеров к номерам, бронируемым партнерами.

### **Вариант 7. Информационная система библиотеки вуза**

Библиотека включает в себя абонементы, читальные залы и справочную систему каталогов и картотек. Читателями библиотеки вуза имеют право быть: студенты всех форм обучения, профессорско-преподавательский состав, аспиранты, ассистенты и другие сотрудники подразделений вуза, слушатели подготовительного отделения (ПО), факультета повышения квалификации (ФПК), стажеры, абитуриенты. Слушатели ФПК, абитуриенты, стажеры - разовые читатели - имеют право пользоваться только читальными залами.

Читатели библиотеки имеют право получать книги и другие источники информации на всех пунктах выдачи библиотеки (абонементах и читальных залах), а также получать необходимые издания по межбиблиотечному абонементу, сделав предварительный заказ.

За нарушение правил пользования библиотекой читатели лишаются права пользования всеми пунктами обслуживания библиотеки на установленные администрацией сроки (от 1 до 6 месяцев). В случае невозвращения в библиотеку книг в установленный срок читатель обязан заплатить штраф.

При поступлении новых изданий в библиотеку они должны быть внесены в картотеку с указанием их количества для каждого абонемента и читального зала. Выдача книг, сроки, штрафы и т.п. собираются и обрабатываются администрацией.

#### Обязательные функции информационной системы:

1. Получить перечень и общее число читателей для данного читального зала или абонента, либо по всей библиотеке.
2. Получить список и общее число всех читателей-задолжников, задолжников со сроком более 10 дней на данном абонементе либо по всей библиотеке, по категориям читателей.
3. Получить перечень и общее число книг, поступивших и утерянных за последний год, для данного читального зала, абонемента или по всей библиотеке, по указанному автору, году выпуска, году поступления в библиотеку.
4. Получить перечень и общее число книг, заказанных на межбиблиотечном абонементе за последний месяц, год.
5. Получить количество экземпляров книги для данного читального зала или абонемента, во всей библиотеке, всех изданий.
6. Получить перечень и общее число читателей, лишенных права пользования библиотекой сроком более двух месяцев, во всей библиотеке, по категориям читателей.
7. Получить перечень и общее число книг, заказанных данным читателем за последний месяц, семестр, год, список книг, которые у него на руках.
8. Определить, есть ли данная книга в наличии на абонементах, и в каком количестве.
9. Получить перечень читателей, у которых на руках некоторая книга и читателя, который раньше всех ее должен сдать.

### **Вариант 8. Информационная система городской телефонной сети**

ГТС представляет собой разветвленную сеть локальных АТС. АТС подразделяются на городские, ведомственные и учрежденческие. У каждой АТС есть свои абоненты. У абонента может стоять телефон одного из трех типов: основной, параллельный или спаренный. За каждым абонентом (у него есть фамилия, имя, отчество, пол, возраст и т.д.) закреплен свой номер телефона. Каждому номеру телефона соответствует адрес (индекс, район, улица, дом, квартира), причем параллельные или спаренные телефоны обязательно должны находиться в одном доме.

Все телефоны городской АТС имеют выход на междугороднюю связь, но для конкретного абонента он может быть либо открыт, либо закрыт по какой-либо причине (отключен по желанию абонента, за неуплату и т.п.). Сведения о междугородних переговорах собираются и анализируются на ГТС.

Абоненты обязаны платить абонентскую плату. Плата должна вноситься каждый месяц до 20-го числа. При неуплате после письменного уведомления в течение двух суток абонент отключается. При задолженности за междугородние разговоры и неоплате после письменного уведомления производится отключение только возможности выхода на междугороднюю связь. Включение того и (или) другого производится при оплате стоимости включения, абонентской платы и пени.

В городе также существуют общественные телефоны и таксофоны, расположенные по определенным адресам.

#### Обязательные функции информационной системы:

1. Получить перечень и общее число абонентов указанной АТС.
2. Получить перечень и общее число должников на указанной АТС, по всей ГТС, по данному району, абонентов, которые имеют задолженность уже больше недели (месяца), по признаку задолженности за междугороднюю связь и (или) по абонентской плате, по размеру долга.
3. Определить АТС (любого или конкретного типа), на которой самое большое (маленькое) число должников, самая большая сумма задолженности.
4. Получить перечень и общее число общественных телефонов и таксофонов во всем городе, принадлежащих указанной АТС, по признаку расположения в данном районе.
5. Получить перечень и общее число абонентов указанной АТС, по всей ГТС, по данному району, по типам АТС, имеющих параллельные телефоны.
6. Определить, есть ли по данному адресу телефон, общее количество телефонов и (или) количество телефонов с выходом на междугороднюю связь, с открытым выходом на междугороднюю связь в данном доме, на конкретной улице.
7. Определить город, с которым происходит большее количество междугородних переговоров.
8. Получить перечень и общее число должников на указанной АТС, по всей ГТС, по данному району, которым следует послать письменное уведомление, отключить телефон и (или) выход на междугороднюю связь.

#### **Вариант 9. Информационная система театра**

Работников театра можно разделить на актеров, музыкантов, постановщиков и служащих. Актеры, музыканты и постановщики, работающие в театре, могут уезжать на гастроли. Актеры театра могут иметь звания заслуженных и народных артистов, могут быть лауреатами конкурсов. Также актерами театра могут быть и студенты театральных училищ. Каждый актер имеет свои вокальные и внешние данные (пол, возраст, голос, рост и т.п.), которые могут подходить для каких-то ролей, а для каких-то нет (не всегда женщина может сыграть мужчину и наоборот).

Для постановки любого спектакля необходимо подобрать актеров на роли и дублеров на каждую главную роль. Естественно, что один и тот же актер не может играть более одной роли в спектакле, но может играть несколько ролей в различных спектаклях. У спектакля также имеется режисер-постановщик, художник-постановщик, дирижер-постановщик, автор. Спектакли можно подразделить по жанрам: музыкальная комедия, трагедия, оперетта и пр. В репертуаре театра указывается какие спектакли, в какие дни и в какое время будут проходить, а также даты премьер. В кассах театра можно заранее приобрести билеты на любые спектакли. Административной функцией театра фиксируется количество проданных билетов на каждый спектакль.

#### Обязательные функции информационной системы:

1. Получить список и общее число все работников театра, актеров, музыкантов, по стажу работы в театре, по половому признаку, году рождения, возрасту, признаку наличия и количества детей, размеру заработной платы.
2. Получить перечень и общее число спектаклей, указанных в репертуаре на данный сезон, уже сыгранных спектаклей, спектаклей указанного жанра, когда-либо сыгранных в этом театре, за указанный период.
3. Получить перечень и общее число всех поставленных спектаклей, спектаклей указанного жанра, когда-либо поставленных в этом театре, поставленных за указанный период.
4. Получить список авторов поставленных спектаклей, авторов, живших в указанном веке, авторов указанной страны, авторов спектаклей указанного жанра когда-либо поставленных в этом театре, поставленных за указанный период времени.
5. Получить список актеров, подходящих по своим данным на указанную роль.
6. Получить общее число и список актеров театра, имеющих звания.
7. Получить список для указанного спектакля: актеров, их дублеров, имена режисера-постановщика, художника-постановщика, дирижера-постановщика, авторов, дату премьеры.
8. Получить перечень и общее число ролей, сыгранных указанным актером всего, за некоторый период времени, в спектаклях определенного жанра, в спектаклях указанного режисера-постановщика.
9. Получить сведения о числе проданных билетов на все спектакли, на конкретный спектакль, на премьеры, за указанный период.
10. Получить общую сумму вырученных денег за указанный спектакль, за некоторый период времени.
11. Получить перечень и общее число свободных мест на все спектакли, на конкретный спектакль, на премьеры.

#### **Вариант 10. Информационная система аэропорта**

Работников аэропорта можно подразделить на пилотов, диспетчеров, техников, кассиров, работников службы безопасности, справочной службы и других, которые административно относятся каждый к своему отделу. В отделах существует разбиение работников на бригады. Отделы возглавляются начальниками, которые представляют собой администрацию аэропорта.

За каждым самолетом закрепляется бригада пилотов, техников и обслуживающего персонала. Пилоты обязаны проходить каждый год медосмотр, не прошедших медосмотр необходимо перевести на другую работу. Самолет должен своевременно осматриваться техниками и при необходимости ремонтироваться. Подготовка к рейсу включает в себя техническую часть (техосмотр, заправка необходимого количества топлива) и обслуживающую часть (уборка салона, запас продуктов питания и т.п.).

В расписании указывается тип самолета, рейс, дни вылета, время вылета и прилета, маршрут (начальный и конечный пункты назначения, пункт пересадки), стоимость билета. Билеты на авиарейсы можно приобрести заранее или забронировать в авиакассах. До отправления рейса, если в этом есть необходимость, билет можно вернуть. Авиарейсы могут быть отменены, если не продано меньше установленного минимума билетов.

Авиарейсы можно разделить на следующие категории: внутренние, международные, чартерные, грузоперевозки, специальные рейсы. Пассажир при посадке в самолет должен предъявить билет, паспорт, а для международного рейса обязан также предъявить заграничный паспорт и пройти таможенный досмотр. Пассажиры могут сдавать свои вещи в багажное отделение. На рейсы грузоперевозок и специальные рейсы билеты не продаются. Для спец. рейсов не существует расписания. Билеты на чартерные рейсы распространяет то агентство, которое его организовало.

#### Обязательные функции информационной системы:

1. Получить список и общее число всех работников аэропорта, начальников отделов, работников указанного отдела, по стажу работы в аэропорту, половому признаку, возрасту, признаку наличия и количеству детей.
2. Получить перечень и общее число работников в бригаде, по всем отделам, в указанном отделе, обслуживающих конкретный рейс.
3. Получить перечень и общее число пилотов, прошедших медосмотр либо не прошедших его в указанный год.
4. Получить перечень и общее число самолетов, приписанных к аэропорту.
5. Получить перечень и общее число самолетов, прошедших техосмотр за определенный период времени, отправленных в ремонт в указанное время, отремонтированных заданное число раз, по количеству совершенных рейсов до ремонта, по возрасту самолета.
6. Получить перечень и общее число рейсов по указанному маршруту, по длительности перелета.
7. Получить перечень и общее число отмененных рейсов полностью, в указанном направлении, по указанному маршруту, по количеству невостребованных мест.
8. Получить перечень и общее число рейсов, по которым летают самолеты заданного типа и среднее количество проданных билетов на определенные маршруты.
9. Получить перечень и общее число авиарейсов указанной категории, в определенном направлении, с указанным типом самолета.
10. Получить перечень и общее число пассажиров на данном рейсе, улетевших в указанный день, по признаку сдачи вещей в багажное отделение, по половому признаку, по возрасту.
11. Получить перечень и общее число свободных и забронированных мест на указанном рейсе, на определенный день, по указанному маршруту, по цене, по времени вылета.

## 7 Оценочные средства для проведения промежуточной аттестации

### а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации:

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
<b>ПК-2. Обладает способностью разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования.</b>		
Знать	<ul style="list-style-type: none"> <li>– основные принципы построения объектно-ориентированного программного обеспечения; основы методологии его использования, принципы агентной парадигмы моделирования;</li> <li>– методологию канонической декомпозиции предметной области, моделирования в виде системы взаимодействующих агентов;</li> <li>– методологию построения иерархических объектных моделей предметной области, выбора между дискретным и непрерывным представлением, выбора между детерминированным и стохастическим представлением модели.</li> </ul>	<p>Список теоретических вопросов:</p> <ul style="list-style-type: none"> <li>– эмпирические модели и их построение в объектной парадигме;</li> <li>– теоретические модели и их построение в объектной парадигме;</li> <li>– понятие об имитационной модели;</li> <li>– имитационное моделирование дискретных процессов в объектной парадигме;</li> <li>– имитационное моделирование непрерывных процессов в объектной парадигме;</li> <li>– неопределенность и ее учет в модели;</li> <li>– понятие о программном агенте;</li> <li>– объектная модель программного агента;</li> <li>– современные методы разработки программных агентов;</li> <li>– создание программных агентов средствами Anylogic;</li> <li>– создание агентной модели непрерывных объектов в Anylogic;</li> <li>– создание агентной модели дискретных объектов в Anylogic;</li> <li>– понятие о дискретно-событийном моделировании и его реализации объектными методами.</li> <li>– создание дискретно-событийной модели детерминированных объектов средствами Anylogic;</li> <li>– создание дискретно-событийной модели недетерминированных объектов средствами Anylogic;</li> <li>– понятие о моделировании в парадигме системной динамики;</li> <li>– создание системно-динамической модели детерминированных объектов средствами Anylogic;</li> <li>– создание системно-динамической модели недетерминированных объектов средствами Anylogic;</li> <li>– понятие об объектных средствах многоподходного моделирования;</li> </ul>



Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		– создание гибридной модели системы разнородных объектов средствами Anylogic.
Уметь	<ul style="list-style-type: none"> <li>– разрабатывать агентные модели;</li> <li>– разрабатывать дискретно-событийные модели;</li> <li>– разрабатывать модели системной динамики;</li> <li>– разрабатывать многоподходные модели.</li> </ul>	<p>Список практических заданий:</p> <ul style="list-style-type: none"> <li>– система определения добросовестности потенциального заемщика в банке. Клерк рассматривает заявление о предоставлении кредита в размере <math>S</math> рублей, анализирует документы потенциального заемщика (кредитную историю, сведения о доходах, сведения о недвижимом имуществе, сведения о поручителях). Заемщик может быть добросовестным и недобросовестным. Добросовестный заемщик предоставляет достоверные документы, недобросовестный заемщик предоставляет подложные документы. Клерк должен оценить истинность документов. Каждый добросовестному заемщику кредит дает банку 10% прибыли от суммы кредита. Каждый кредит недобросовестному заемщику приносит <math>1,2S</math> рублей убытка. Создать систему, в которой имеется <math>M</math> заемщиков и <math>N</math> клерков (<math>M &gt; N</math>). Тип заемщика задается в системе, клерку он неизвестен. Сформировать систему агентов и провести имитационное моделирование;</li> <li>– Торговое предприятие продвигает свои товары на рынок. Для этого используются Интернет – тролли. Каждый из них размещает в Интернете рекламные тексты о товарах из ассортимента торгового предприятия. Вероятность того, что пользователя Интернета удастся уговорить приобрести товар, равна <math>q</math>. Выполнить моделирование задачи средствами Anylogic.</li> </ul>
Владеть	<ul style="list-style-type: none"> <li>– методами применения современных объектно-ориентированных программных средств;</li> <li>– методами применения средств поддержки мультипарадигменного моделирования;</li> <li>– методами расширения функциональности средств мультипарадигменного моделирования.</li> </ul>	<p>Список комплексных заданий:</p> <ul style="list-style-type: none"> <li>– условия задания 1 из предыдущего раздела пополнить следующими данными: наличие различных условий кредитования — различный процент по кредиту, возможность досрочного погашения. Выполнить моделирование описанной системы средствами Anylogic.</li> <li>– условия задачи 2 из предыдущего раздела пополнить возможностью вести рекламную компанию в электронной и «бумажной». Выполнить моделирование описанной системы средствами Anylogic.</li> </ul>

## **б) Порядок проведения промежуточной аттестации, показатели и критерии оценивания:**

Промежуточная аттестация по дисциплине «Объектно-ориентированное программное обеспечение» включает теоретические вопросы, позволяющие оценить уровень усвоения обучающимися знаний, и практические задания, выявляющие степень сформированности умений и владений, проводится в форме экзамена и в форме выполнения и защиты курсовой работы.

Экзамен по данной дисциплине проводится в устной форме по экзаменационным билетам, каждый из которых включает два теоретических вопроса и одно практическое задание.

### **Показатели и критерии оценивания экзамена:**

– на оценку **«отлично»** (5 баллов) – обучающийся демонстрирует высокий уровень сформированности компетенций, всестороннее, систематическое и глубокое знание учебного материала, свободно выполняет практические задания, свободно оперирует знаниями, умениями, применяет их в ситуациях повышенной сложности.

– на оценку **«хорошо»** (4 балла) – обучающийся демонстрирует средний уровень сформированности компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.

– на оценку **«удовлетворительно»** (3 балла) – обучающийся демонстрирует пороговый уровень сформированности компетенций: в ходе контрольных мероприятий допускаются ошибки, проявляется отсутствие отдельных знаний, умений, навыков, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

– на оценку **«неудовлетворительно»** (2 балла) – обучающийся демонстрирует знания не более 20% теоретического материала, допускает существенные ошибки, не может показать интеллектуальные навыки решения простых задач.

– на оценку **«неудовлетворительно»** (1 балл) – обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать интеллектуальные навыки решения простых задач.

## **8 Учебно-методическое и информационное обеспечение дисциплины (модуля)**

### **а) Основная литература:**

1. Зайцев, М.Г. Объектно-ориентированный анализ и программирование : учебное пособие : [16+] / М.Г. Зайцев ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2017. – 84 с. : ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=576800> (дата обращения: 30.10.2020). – Библиогр. в кн. – ISBN 978-5-7782-3308-9. – Текст : электронный.
2. Корчуганова, М.Р. Объектно-ориентированное программирование на C++: электронное учебное пособие / М.Р. Корчуганова, К.С. Иванов, Л.В. Бондарева ; Кемеровский государственный университет, Кафедра вычислительной математики. – Кемерово : Кемеровский государственный университет, 2015. – 196 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=481559> (дата обращения: 30.10.2020). – Библиогр. в кн. – ISBN 978-5-8353-1832-2. – Текст : электронный.

### **б) Дополнительная литература:**

1. Сорокин, А.А. Объектно-ориентированное программирование: учебное пособие (курс лекций) / А.А. Сорокин ; Северо-Кавказский федеральный университет. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2014. – 174 с. : ил. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=457696> (дата обращения: 30.10.2020). – Текст : электронный.

### **в) Методические указания:**

1. Зарецкий М.В. Методические указания для выполнения самостоятельных работ по курсу «Объектно-ориентированное программирование» для студентов специальности 220400 / М.В. Зарецкий, Ю.Б. Кухта - Магнитогорск. МГТУ, 2005- 17 с.

**г) Программное обеспечение и Интернет-ресурсы:**

*Программное обеспечение:* лицензионное программное обеспечение: операционная система; офисные программы; математические пакет, статистические пакеты, установленные на каждом персональном компьютере вычислительного центра ФГБОУ ВПО «МГТУ».

Перечень лицензионного программного обеспечения по ссылке:

<http://sps.vuz.magtu.ru/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FShared%20Documents%2F%D0%9F%D0%BE%D0%B4%D0%B3%D0%BE%D1%82%D0%BE%D0%B2%D0%BA%D0%B0%20%D0%BA%20%D0%B0%D0%BA%D0%BA%D1%80%D0%B5%D0%B4%D0%B8%D1%82%D0%B0%D1%86%D0%B8%D0%B8%202020%2F%D0%A1%D0%B0%D0%BC%D0%BE%D0%BE%D0%B1%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%202019%D0%B3%2F%D0%9B%D0%B8%D1%86%D0%B5%D0%BD%D0%B7%D0%B8%D0%BE%D0%BD%D0%BD%D0%BE%D0%B5%20%D0%9F%D0%9E&InitialTabId=Ribbon.Document&VisibilityContext=WSSTabPersistence>

Официальные сайты промышленных предприятий и организаций: <http://www.mmk.ru>, <http://www.creditural.ru>, <http://www.magtu.ru>, <http://www.gks.ru> и т.п.; разработчиков программных продуктов: <http://www.statsoft.ru>, <http://www.microsoft.com>, <http://www.ptc.com> и т.п; сайты лабораторий компьютерной графики <http://graphics.cs.msu.ru> , <http://cgm.graphicon.ru>.

**9 Материально-техническое обеспечение дисциплины (модуля)**

Материально-техническое обеспечение дисциплины включает:

Тип и название аудитории	Оснащение аудитории
Лекционная аудитория	Мультимедийные средства хранения, передачи и представления информации
Компьютерный класс	Персональные компьютеры с пакетом Office, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета
Аудитории для самостоятельной работы: компьютерные классы; читальные залы библиотеки	Все классы УИТ и АСУ с персональными компьютерами, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета
Аудиторий для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Ауд. 282 и классы УИТ и АСУ
Помещения для самостоятельной работы обучающихся, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и наличием доступа в электронную информационно-образовательную среду организации	Классы УИТ и АСУ
Помещения для хранения и профилактического обслуживания учебного оборудования	Центр информационных технологий – ауд. 379

