

ПРИЛОЖЕНИЕ А
ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ «Автоматное программирование»

1. Перечень оценочных средств для компетенций, формируемых в результате освоения дисциплины

Код контролируемой компетенции	Способ оценивания	Оценочное средство
ПК-5: Владение навыками использования различных технологий разработки программного обеспечения	Зачет	Комплект контролирующих материалов для зачета

2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Оцениваемые компетенции представлены в разделе «Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций» рабочей программы дисциплины «Автоматное программирование».

При оценивании сформированности компетенций по дисциплине «Автоматное программирование» используется 100-балльная шкала.

Критерий	Оценка по 100-балльной шкале	Оценка по традиционной шкале
Студент освоил изучаемый материал, выполняет задания в соответствии с индикаторами достижения компетенций, может допускать отдельные ошибки.	25-100	<i>Зачтено</i>
Студент не освоил основное содержание изученного материала, задания в соответствии с индикаторами достижения компетенций не выполнены или выполнены неверно.	0-24	<i>Не зачтено</i>

3. Типовые контрольные задания или иные материалы, необходимые для оценки уровня достижения компетенций в соответствии с индикаторами

1.Примеры задач

Компетенция	Индикатор достижения компетенции
ПК-5 Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1 Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
	ПК-5.2 Использует современные технологии разработки программного обеспечения для решения прикладных задач

В результате освоения дисциплины “Автоматное программирование” у обучающихся должны быть сформированы компетенции с соответствующими индикаторами их достижения:

Компетенция	Содержание компетенции	Индикатор	Содержание индикатора
ПК-5	Владение навыками использования различных технологий разработки программного	ПК-5.1	Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
		ПК-5.2	Использует современные технологии разработки программного обеспечения для решения прикладных задач

Кейс 1

Имеется система “Банкомат”. Банкомат позволяет считывать информацию с карты пользователя, просматривать счет, снимать деньги, переводить их между своими счетами, оплачивать услуги, например, за телефон.

Вам предлагается сделать программную модель такой системы.

Для этого необходимо:

1. Обосновать и выбрать ключевую технологию разработки ПО для решения данной задачи. Технология должна позволять выполнить следующие возможности:
 - Работать с режимами работы как с логически выделенными состояниями.
 - Осуществлять переключения между режимами как дискретные переходы между состояниями, используя данные текущего состояния и реакцию на действия пользователя.
 - Обеспечивать объектно-ориентированную абстракцию, то есть не содержать комбинацию последовательного кода по переключению режимов, обработки действий пользователя и реакции на эти действия в каждом из режимов.
2. Использовать современные программные технологии разработки для:
 - Первоначального графического моделирования работы системы (UML диаграммы и редакторы, в том числе, встроенные в среду разработки и позволяющие генерировать код).
 - Реализации спроектированных состояний и переходов (реализовать свою систему или использовать готовую вроде Workflow Foundation).
 - Осуществления взаимодействия с пользователем (одна из систем реализации интерфейса, графического (WPF, Winforms, Qt, web) или интерфейса командной строки).
3. Оставить возможности расширения решения.

В результате должна получиться предварительно спроектированная, работоспособная, расширяемая программная система.

Кейс 2

Имеется система “Лифт”. Лифт может перемещаться между этажами, реагируя на вызовы, исходящие от кнопок на этажах и внутри лифта. Лифт не едет без пассажиров и с открытыми дверями.

Вам предлагается сделать программную модель такой системы с визуализацией перемещения лифта между этажами.

Для этого необходимо:

1. Обосновать и выбрать ключевую технологию разработки ПО для решения данной задачи. Технология должна позволять выполнить следующие возможности:
 - Работать с режимами работы кабины как с логически выделенными состояниями.
 - Осуществлять переключения между режимами как дискретные переходы между состояниями, используя данные текущего состояния (наличие/отсутствие движения,

- состояния дверей, состояния кнопок, наличие пассажира и реакцию на действия пользователя).
- Обеспечивать объектно-ориентированную абстракцию, то есть не содержать комбинацию последовательного кода по переключению режимов, обработки действий пользователя и реакции на эти действия в каждом из режимов.
2. Использовать современные программные технологии разработки для:
 - Первоначального графического моделирования работы системы (UML диаграммы и редакторы, в том числе, встроенные в среду разработки и позволяющие генерировать код).
 - Реализации спроектированных состояний и переходов (реализовать свою систему или использовать готовую вроде Workflow Foundation).
 - Осуществления взаимодействия с пользователем (одна из систем реализации интерфейса, WPF, Winforms, Qt, web).
 3. Оставить возможности расширения решения.

В результате должна получиться предварительно спроектированная, работоспособная, расширяемая программная система.

Кейс 3

Имеется система “Набор лифтов”. Лифт может перемещаться между этажами, реагируя на вызовы, исходящие от кнопок на этажах и внутри лифта. В помещении имеется два лифта, которые управляются теми же кнопками на этажах и действуют асинхронно.

Вам предлагается сделать программную модель такой системы с визуализацией перемещения лифтов между этажами.

Для этого необходимо:

1. Обосновать и выбрать ключевую технологию разработки ПО для решения данной задачи. Технология должна позволять выполнить следующие возможности:
 - Работать с режимами работы кабины как с логически выделенными состояниями.
 - Осуществлять переключения между режимами как дискретные переходы между состояниями, используя данные текущего состояния (наличие/отсутствие движения, состояния дверей, состояния кнопок, реакцию на действия пользователя).
 - Позволять моделировать асинхронные перемещения кабин и их синхронизацию.
 - Обеспечивать объектно-ориентированную абстракцию, то есть не содержать комбинацию последовательного кода по переключению режимов, обработки действий пользователя и реакции на эти действия в каждом из режимов.
2. Использовать современные программные технологии разработки для:
 - Первоначального графического моделирования работы системы (UML диаграммы и редакторы, в том числе, встроенные в среду разработки и позволяющие генерировать код).
 - Реализации спроектированных состояний и переходов (реализовать свою систему или использовать готовую вроде Workflow Foundation).
 - Осуществления взаимодействия с пользователем (одна из систем реализации интерфейса, WPF, Winforms, Qt, web).
3. Оставить возможности расширения решения.

В результате должна получиться предварительно спроектированная, работоспособная, расширяемая программная система.

4. Файл и/или БТЗ с полным комплектом оценочных материалов прилагается.