

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
учреждения высшего образования
«Пермский национальный исследовательский политехнический университет»

УТВЕРЖДАЮ

Доцент с исп. обязанностей
зав.кафедрой ТД

 Т.О. Сошина

« 01 » 06 2024 г

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля успеваемости и промежуточной
аттестации обучающихся по учебной дисциплине

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Приложение к рабочей программе учебной дисциплины

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 Информационные системы и программирование
(базовая подготовка)

Лысьва, 2024

Фонд оценочных средств разработан на основе:

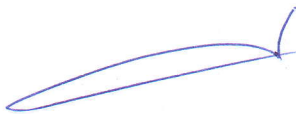
– Федерального государственного образовательного стандарта среднего профессионального образования, утверждённого приказом Министерства образования и науки Российской Федерации «9» декабря 2016 г. № 1547 по специальности 09.02.07 Информационные системы и программирование;

– рабочей программы учебной дисциплины *Основы алгоритмизации и программирования*, утвержденной «01» 06 2024 г.

Разработчик: преподаватель высшей категории Федосеева Е.Л

Фонд оценочных средств рассмотрен и одобрен на заседании предметной (цикловой) комиссии *Естественнонаучных дисциплин* (ПЦК ЕНД) «06» 02 2024 г., протокол № 6.

Председатель ПЦК ЕНД



М.Н. Апталаев

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

В результате освоения учебной дисциплины **Основы алгоритмизации и программирования** обучающийся должен обладать предусмотренными ФГОС по специальности СПО *09.02.07 Информационные системы и программирование* следующими результатами обучения: знаниями и умениями, которые формируют общие и профессиональные компетенции.

Код ОК, ПК, ЛР	Умения	Знания
ОК 01 ОК 02 ОК 04 ОК 05 ОК 09 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 2.4 ПК 2.5 ЛР 5 - 8, 11, 12, 14, 17	<ul style="list-style-type: none"> – разрабатывать алгоритмы для конкретных задач; – использовать программы для графического отображения алгоритмов; – определять сложность работы алгоритмов; – работать в среде программирования; – реализовывать построенные алгоритмы в виде программ на конкретном языке программирования; – оформлять код программы в соответствии со стандартом кодирования; – выполнять проверку, отладку кода программы 	<ul style="list-style-type: none"> – понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции; – эволюцию языков программирования, их классификацию, понятие системы программирования; – основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; – понятие подпрограммы, принципы составления библиотек подпрограмм; – объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения

Перечень общих компетенций элементы, которых формируются в рамках учебной дисциплины:

Код ОК	Наименование ОК
ОК 01	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ОК 04	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста
ОК 09	Пользоваться профессиональной документацией на государственном и иностранном языках

Перечень профессиональных компетенций элементы, которых формируются в рамках учебной дисциплины:

Код ПК	Наименование ПК
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 1.4	Выполнять тестирование программных модулей
ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода.
ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования

После изучения учебной дисциплины обучающийся должен демонстрировать следующие личностные результаты:

Код ЛР	Характеристика ЛР
ЛР 5	демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации
ЛР 6	демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм
ЛР 7	демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности

<i>ЛР 8</i>	пользоваться профессиональной документацией на государственном и иностранном языках
<i>ЛР 11</i>	планировать и реализовывать собственное профессиональное и личностное развитие в условиях развития информационных технологий, применяемых в различных отраслях народного хозяйства
<i>ЛР 12</i>	активно применяющий полученные знания на практике
<i>ЛР 14</i>	работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
<i>ЛР 17</i>	проявлять доброжелательность к окружающим, деликатность, чувство такта и готовность оказать услугу каждому кто в ней нуждается

1 МЕТОДЫ И ФОРМЫ КОНТРОЛЯ ОЦЕНИВАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

1 Для текущего и рубежного контроля освоения дисциплинарных компетенций используются следующие методы:

- *устный опрос,*
- *тестирование,*
- *наблюдение и оценка результатов практических и лабораторных занятий;*
- *экспертная оценка результатов самостоятельной работы;*
- *экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины.*

2 Формой промежуточной аттестации по учебной дисциплине является: **дифференцированный зачёт (2 семестр), экзамен (3 семестр),** которые проводятся в сроки, установленные учебным планом и определяемые календарным учебным графиком образовательного процесса.

Таблица 1 – Методы и формы контроля и оценивания элементов учебной дисциплины

Элемент учебной дисциплины	Методы и формы контроля и оценивания		
	Текущий контроль	Рубежный контроль	Промежуточная аттестация
Раздел 1 Введение в программирование			
Тема 1.1 Языки программирования	Устный опрос Наблюдение и оценка результатов практических занятий	Тестирование	Дифференцированный зачет
Тема 1.2 Типы и переменные	Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины		
Раздел 2 Основные конструкции языков программирования			
Тема 2.1 Основные элементы языка программирования. Линейные алгоритмы	Устный опрос Наблюдение и оценка результатов практических занятий Экспертная оценка результатов самостоятельной		
Тема 2.2			

<p>Операторы выбора</p> <p>Тема 2.3 Операторы цикла и перехода</p> <p>Тема 2.4 Массивы</p> <p>Тема 2.5 Строки и множества</p>	<p>работы</p> <p>Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины</p>		
Раздел 3 Структурное программирование			
<p>Тема 3.1 Подпрограммы: процедуры и функции</p> <p>Тема 3.2 Файлы</p> <p>Тема 3.3 Структуризация в программировании</p> <p>Тема 3.4 Модульное программирование</p> <p>Тема 3.5 Указатели</p>	<p>Устный опрос</p> <p>Наблюдение и оценка результатов практических занятий</p> <p>Наблюдение и оценка результатов лабораторных занятий</p> <p>Экспертная оценка результатов самостоятельной работы</p> <p>Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины</p>	Тестирование	Дифференцированный зачет
Раздел 4 Введение в объектно-ориентированное программирование			
<p>Тема 4.1 Основные принципы ООП</p> <p>Тема 4.2 Интегрированная среда разработчика</p> <p>Тема 4.3 Визуальное событийно-управляемое программирование</p> <p>Тема 4.4 Разработка оконного приложения</p> <p>Тема 4.5 Этапы</p>	<p>Устный опрос</p> <p>Наблюдение и оценка результатов практических занятий</p> <p>Наблюдение и оценка результатов лабораторных занятий</p> <p>Экспертная оценка результатов самостоятельной работы</p> <p>Экспертная оценка по результатам</p>	Тестирование	

<p>разработки приложений</p> <p>Тема 4.6 Иерархия классов</p> <p>Тема 4.7 Сериализация объектов</p> <p>Тема 4.8 Взаимодействие объектов</p>	<p>наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины</p>		
<p>Форма контроля</p>			<p>Дифференцированный зачет Экзамен</p>

Текущий контроль усвоения материала

Текущий контроль усвоения материала проводится в форме устного опроса обучающихся по темам учебной дисциплины.

Наблюдение и оценка результатов практических и лабораторных занятий

Типовые темы практических и лабораторных занятий приведены в РПД. Комплект заданий на практические занятия приведены в МУ по ПЗ по учебной дисциплине. Комплект заданий на лабораторные занятия приведены в МУ по ЛЗ по учебной дисциплине.

Защита отчетов по практическим и лабораторным занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Экспертная оценка результатов самостоятельной работы

Задания для самостоятельной работы приведены в МУ по СРС по учебной дисциплине.

Качественная оценка определения научного кругозора, степенью овладения методами теоретического исследования и развития самостоятельности мышления студента.

Способом проверки качества организации самостоятельной работы студентов является контроль:

- корректирующий (может осуществляться во время индивидуальных консультаций по поводу выполнения формы самостоятельной работы);
- констатирующий (по результатам выполнения специальных форм самостоятельной работы);
- самоконтроль (осуществляется самим студентом);
- текущий (в ходе выполнения различных форм самостоятельной работы, установленных рабочей программой);

— промежуточный (оценка результата обучения как итога выполнения студентом всех форм самостоятельной работы).

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины

Осуществляется как наблюдение за процессом деятельности обучающегося в режиме реального времени. Является качественной оценкой освоения учебной дисциплины, учитываемой при промежуточной аттестации.

Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, освоенных умений проводится в форме тестирования (после изучения разделов учебной дисциплины).

2 РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ ПРИ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В результате промежуточной аттестации по учебной дисциплине осуществляется комплексная проверка следующих умений и знаний:

Результаты обучения (освоенные умения, усвоенные знания)	Показатели оценки результатов
Уметь:	
– разрабатывать алгоритмы для конкретных задач	умеет разрабатывать алгоритмы для конкретных задач
– использовать программы для графического отображения алгоритмов	умеет использовать программы для графического отображения алгоритмов
– определять сложность работы алгоритмов	умеет определять сложность алгоритмов задачи
– работать в среде программирования	умеет работать в среде программирования
– реализовывать построенные алгоритмы в виде программ на конкретном языке программирования	умеет реализовывать построенные алгоритмы в виде программ на конкретном языке программирования
– оформлять код программы в соответствии со стандартом кодирования	умеет оформлять код программы в соответствии со стандартом кодирования
– выполнять проверку, отладку кода программы	умеет выполнять проверку, отладку кода программы и исправлять ошибки, если они появляются
Знать:	
– понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции	знает понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции
– эволюцию языков программирования, их классификацию, понятие системы программирования	знает эволюцию языков программирования, их классификацию и понятие системы программирования
– основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти	знает основные элементы языка программирования и его структуру, основные операции и управляющие структуры языка программирования
– понятие подпрограммы, принципы составления библиотек подпрограмм	знает понятия подпрограмм и принципы их использования
– объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения	знает объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения

3 КРИТЕРИИ ОЦЕНКИ

Критерии устного ответа

Критерии оценки	Оценка
Дан полный, развернутый ответ на поставленный вопрос, показана совокупность осознанных знаний по дисциплине, доказательно раскрыты основные положения вопросов; в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений. Знания по предмету демонстрируются на фоне понимания его в системе данной науки и междисциплинарных связей. Ответ изложен литературным языком с использованием современной технической терминологии. Могут быть допущены недочеты в определении понятий, исправленные студентом самостоятельно в процессе ответа.	Отлично
Дан полный, развернутый ответ на поставленный вопрос, показано умение выделить существенные и несущественные признаки, причинно-следственные связи. Ответ четко структурирован, логичен, изложен литературным языком с использованием современной технической терминологии. Могут быть допущены некоторые неточности или незначительные ошибки, исправленные студентом с помощью преподавателя.	Хорошо
Дан недостаточно полный и недостаточно развернутый ответ. Логика и последовательность изложения имеют нарушения. Допущены ошибки в раскрытии понятий, употреблении терминов. Студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи. В ответе отсутствуют выводы. Умение раскрыть значение обобщенных знаний не показано. Речевое оформление требует поправок, коррекции.	Удовлетворительно
1) Ответ представляет собой разрозненные знания с существенными ошибками по вопросу. Присутствуют фрагментарность, нелогичность изложения. Студент не осознает связь обсуждаемого вопроса с другими объектами дисциплины. Отсутствуют выводы, конкретизация и доказательность изложения. Речь неграмотная, техническая терминология не используется. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента. 2) Ответ на вопрос полностью отсутствует. 3) Отказ от ответа.	Неудовлетворительно

Критерии оценки практических и лабораторных занятий

1 активность работы на практическом и лабораторном занятиях (выполнение всех заданий, предложенных преподавателем);

2 правильность ответов на вопросы (верное, четкое и достаточно глубокое изложение понятий, идей, и т.д.);

3 полнота и одновременно лаконичность ответа (ответ должен отражать основные теории и концепции по раскрываемому вопросу, содержать их критический анализ и сопоставление);

4 умение формулировать собственную точку зрения, грамотно аргументировать свою позицию по раскрываемому вопросу;

5 культура речи (материал должен быть изложен хорошим профессиональным языком, с грамотным использованием соответствующей системы понятий и терминов).

Критерии оценки практического задания

Критерии оценки	Оценка
<ul style="list-style-type: none"> – практическое задание выполнено в установленный срок с использованием рекомендаций преподавателя – показан высокий уровень знания изученного материала по заданной теме – проявлен творческий подход – умение глубоко анализировать проблему и делать обобщающие практико-ориентированные выводы – работа выполнена без ошибок и недочетов или допущено не более одного недочета 	Отлично
<ul style="list-style-type: none"> – практическое задание выполнено в установленный срок с использованием рекомендаций преподавателя – показан хороший уровень владения изученным материалом по заданной теме – работа выполнена полностью, но допущено в ней: <ul style="list-style-type: none"> а) не более одной негрубой ошибки и одного недочета б) или не более двух недочетов 	Хорошо
<ul style="list-style-type: none"> – практическое задание выполнено в установленный срок с частичным использованием рекомендаций преподавателя – продемонстрированы минимальные знания по основным темам изученного материала – выполнено не менее половины работы или допущены в ней: <ul style="list-style-type: none"> а) не более двух грубых ошибок; б) не более одной грубой ошибки и одного недочета; в) не более двух-трех негрубых ошибок; г) одна негрубая ошибка и три недочета; д) при отсутствии ошибок, 4-5 недочетов 	Удовлетворительно
<ul style="list-style-type: none"> – число ошибок и недочетов превосходит норму, при которой может быть выставлена оценка «удовлетворительно» или если правильно выполнено менее половины задания – если обучающийся не приступал к выполнению задания или правильно выполнил не более 10 процентов всех заданий 	Неудовлетворительн о

Критерии оценки лабораторного задания

Критерии оценки	Оценка
<ul style="list-style-type: none"> – работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов и измерений; все опыты проведены в условиях и режимах, обеспечивающих получение правильных результатов и выводов; соблюдены требования правил безопасности труда; в отчете правильно и аккуратно выполнены все записи, таблицы, рисунки, чертежи, графики, вычисления; правильно 	Отлично

выполнен анализ погрешностей	
– работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов и измерений; все опыты проведены в условиях и режимах, обеспечивающих получение правильных результатов и выводов; соблюдены требования правил безопасности труда; в отчете правильно и аккуратно выполнены все записи, таблицы, рисунки, чертежи, графики, вычисления; правильно выполнен анализ погрешностей Допущено два - три недочета или не более одной негрубой ошибки и одного недочёта	Хорошо
– работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы, если в ходе проведения опыта и измерений были допущены ошибки.	Удовлетворительно
Работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов, если опыты, измерения, вычисления, наблюдения производились неправильно	Неудовлетворительно

Критерии оценивания тестов

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
86 - 100	70 - 85	51 69	68 и менее

Критерии оценивания результатов самостоятельной работы

При экспертной оценке результатов самостоятельной работы учитываются такие критерии:

- Глубина освоения знаний
- Источники информации
- Качество выполнения работы
- Самостоятельность изложения
- Творчество и личный вклад
- Соблюдение правил оформления

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины

Интегральная качественная оценка освоения учебной дисциплины, учитываемая при промежуточной аттестации.

Критерии оценки промежуточной аттестации

Изучение учебной дисциплины осуществляется в течение двух семестров.

Промежуточные аттестации проводятся в формах: **дифференцированного зачета (2 семестр), экзамена (3 семестр).**

К сдаче дифференцированного зачета допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации оценки не ниже «удовлетворительно».

Дифференцированный зачет проводится по завершению курса изучения учебной дисциплины в форме выполнения практического задания с последующим собеседованием с преподавателем с учетом результатов текущего контроля.

Дифференцированный зачет оценивается по четырехбалльной шкале: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Экзамен проводится по завершению курса изучения учебной дисциплины по билетам, содержащим два теоретических вопроса и практическое задание.

К сдаче экзамена допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации оценки не ниже «удовлетворительно».

Экзамен оценивается по четырехбалльной шкале: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Основой для определения оценки на промежуточной аттестации служит объем и уровень усвоения обучающимися материала, предусмотренного рабочей программой учебной дисциплины «Основы алгоритмизации и программирования».

Критерии оценки дифференцированного зачета

Критерии оценки	Оценка
Всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполненные все предусмотренные программой задания, глубоко усвоенные основная и дополнительная литература, рекомендованная программой, активная работа на практических занятиях. Обучающийся разбирается в основных научных концепциях по изучаемой учебной дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала. Ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично	Отлично
Достаточно полное знание учебно-программного материала. Обучающийся не допускает в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических занятиях, показавший систематический характер знаний по учебной	Хорошо

дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению	
Обучающийся показал знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по специальности, не отличавшийся активностью на практических занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на дифференцированном зачете, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей	Удовлетворительно
обучающийся обнаруживает пробелы в знаниях или отсутствие знаний по значительной части основного учебно - программного материала, не выполнивший самостоятельно предусмотренные программой основные задания, допустивший принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавший основные практические занятия, допускающий существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей учебной дисциплине	Неудовлетворительно

Критерии оценивания экзамена

Критерии оценки	Оценка
<p>Всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполненные все предусмотренные программой задания, глубоко усвоенные основная и дополнительная литература, рекомендованная программой, активная работа на практических и лабораторных занятиях</p> <p>Обучающийся разбирается в основных научных концепциях по изучаемой учебной дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала.</p> <p>Ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично</p>	Отлично
<p>Достаточно полное знание учебно-программного материала.</p> <p>Обучающийся не допускает в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических и лабораторных занятиях, показавший систематический характер знаний по учебной дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению</p>	Хорошо
<p>Обучающийся показал знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по специальности, не отличавшийся активностью на практических и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на дифференцированном зачете, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей</p>	Удовлетворительно
<p>обучающийся обнаруживает пробелы в знаниях или отсутствие</p>	Неудовлетворительно

<p>знаний по значительной части основного учебно - программного материала, не выполнивший самостоятельно предусмотренные программой основные задания, допустивший принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавший основные практические и лабораторные занятия, допускающий существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей учебной дисциплине</p>	
---	--

4 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ ЗНАНИЙ И УМЕНИЯ

Типовые задания для оценки освоения

Раздела 1 Введение в программирование

Обучающийся должен

знать:

- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;
- эволюцию языков программирования, их классификацию, понятие системы программирования.

Типовые вопросы для устного опроса

1. Перечислите области применения языков программирования.
2. Назовите языки программирования высокого уровня.
3. Перечислите основные этапы решения задач на компьютере.
4. Что такое компилятор?
5. Что такое интерпретатор?
6. Назовите основные типы переменных языка C#
7. Опишите неявное преобразование типа переменных, приведите примеры.
8. Опишите явное преобразование типа переменных, приведите примеры.

Типовые задания для оценки освоения

Раздела 2 Основные конструкции языков программирования

Обучающийся должен

знать:

- понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;
- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;

уметь:

- разрабатывать алгоритмы для конкретных задач;
- использовать программы для графического отображения алгоритмов;

- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- оформлять код программы в соответствии со стандартом кодирования;
- выполнять проверку, отладку кода программы.

Типовые вопросы для устного опроса

1. Как описываются константы на языке C#
2. Перечислите основные операторы языка C#
3. Что такое составной оператор?
4. Что такое оператор выбора?
5. Как записываем вложенные условные операторы?
6. Какие виды циклов существуют?
7. Опишите цикл с постусловием
8. Опишите цикл с предусловием
9. Опишите цикл с параметром
10. Перечислите основные операторы перехода.
11. Что такое вложенные циклы?
12. Что такое массив?
13. Как объявляется массив на языке программирования C#?
14. Перечислите операции над множествами.

Типовой тест

Раздел «Введение в программирование», «Основные конструкции языков программирования»

Условия выполнения задания

- *тест выполняется в аудитории во время практических занятий;*
- *для выполнения теста необходимо следующее оборудование: бланки ответов, ручки, карточки с тестами (для выполнения электронного варианта теста: компьютерный класс, тестировщик).*

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа, ответы занесите в бланк ответов.

1. Установите соответствие

bool		'g'
int		"Hello"
double		45
char		3.14
string		False

2. Установите соответствие

табуляция		/n
удаление предыдущего символа		/r
перевод каретки на следующую строку		/b
возврат каретки		/t

3. Строго определенная последовательность действий, необходимых для решения поставленной задачи – это ...

- A) метод решения Б) алгоритм В) блок-схема

4. В технологической цепочке решения задач на ЭВМ
 постановка задачи — математическая формализация — построение алгоритма — перевод алгоритма на язык программирования — ... — анализ полученных результатов
 отсутствует пункт ...

- 1) отладка и тестирование программы 2) определение данных и требуемых результатов
 3) графическое описание процесса 4) ввод и редактирование программы

5. При разработке программного продукта формализация постановки задачи выполняется на этапе ...

- 1) исследования эффективности алгоритма 2) тестирования и отладки
 3) кодирования алгоритма 4) построения математической модели

6. Какое значение примет переменная C в результате выполнения программы:

```

алг V1;
цел A, B, C
нач
  A:=2;
  B:=A*3 + 1;
  если B>A+5 то A := 4;
  если B<A+5 то A := 6;
  если B=A+5 то A := B - A;

```

```

C := A*B + A - B;
вывод ('C=', C);

```

кон.

- 1) 33 2) 25 3) 5 4) 41

7. Укажите, какие операции относятся к логическим операциям:

- 1) IF, OR, AND; 3) OR, MOD, AND;
 2) AND, OR, NOT; 4) NOT, AND, THEN.

8. Тело цикла - это...

- 1) группа команд, не входящих в циклическую структуру;
 2) произвольный текст;

- 3) группа команд, повторяющихся некоторое число раз;
произвольная группа команд?
9. Что называется массивом?
- 1) Под *массивом* понимается совокупность конечного числа данных различных типов.
 - 2) Под *массивом* понимается совокупность конечного числа данных одного типа.
 - 3) Под *массивом* понимается совокупность бесконечного числа данных одного типа
10. Верным является следующее объявление и инициализация переменных ...
- A) double n, k=3; Б) float n, k, int z=-1; В) char a, b='No'; Г) float *t=4;
11. После выполнения фрагмента кода программы
- ```
int x, y; float z=1;
for (x = 0, y = 16; x + y >= 8; x ++)
```
- ```
{
  z += float(x - y)/2;
  y -= 3;
}
```
- переменные x, y, z будут иметь значения ...
- A) x = 4, y = 4, z = -19 В) x = 5, y = 5, z = -18
- Б) x = 8, y = 0, z = -20 Г) x = 5, y = 1, z = -19
12. Какие циклы существуют в языке C#:
- A) for, while Б) for, while, do while, foreach В) for, while, do while
13. Перевод программы с языка программирования на язык машинных кодов называется ...
- Ответ: _____
14. Процесс исправления ошибок в программе, при котором цель исправить все ошибки на ставится, называется ...
- Ответ: _____
15. Массивом называют упорядоченную совокупность элементов ... типа
- Ответ: _____
16. Какие бывают массивы ...
- A) Одномерные и многомерные Б) Резиновые и статичные В) Сложные и простые
17. Чему будет равен c, если int a = 0; int c = --a;
- A) Null Б) 1 В) -1
18. Как найти квадратный корень из числа x ...
- A) Sqrt(x) Б) Math.Sqrt(x) В) Arifmetic.sqrt(x)
19. Что будет содержать строка name после выполнения данного кода?
- ```
1 | string name = "Alexey";
2 | name[1] = ".";
```
- A) "A.еxey"                      Б) "Alexey"                      В) ".lexey"                      Г) Возникнет ошибка
20. Верно ли, что тип string хранит набор символов Unicode?
- A) Нет                      Б) Да
21. Что будет выведено на консоль?

1 | `Console.WriteLine("Text" + " test");`

- A) String.Empty
- Б) "Text test"
- В) "Text"
- Г) Возникнет ошибка
- Д) " test"

22. Сумму элементов каждого столбца двумерного массива позволяет найти фрагмент кода ...

- A) 

```
int j, i, s[5], num[5][6];
s[0]=s[1]=s[2]=s[3]=s[4]=0;
for (i = 0; i < 5; ++i)
{
 for (j = 0; j < 6; ++j)
 s[i] += num[i][j];
}
```
- Б) 

```
int j, i, s[6], num[5][6];
for (i = 0; i < 5; ++i)
{
 s[i]=0;
 for (j = 0; j < 6; ++j)
 s[i] += num[i][j];
}
```
- В) 

```
int j, i, s[6], num[5][6];
for (j = 0; j < 6; ++j)
{
 s[j]=0;
 for (i = 0; i < 5; i++)
 s[j] += num[i][j];
}
```
- Г) 

```
int j, i, s[5], num[5][6];
for (j = 0; j < 6; ++j)
{
 s[j]=0;
 for (i = 0; i < 5; ++i)
 s[i] = num[i][j];
}
```

23. Чему будет равен с, если `int a = 10; int b = 4; bool c = (a == 10 && b == 4);`

- A) True
- Б) Null
- В) False

24. Тип данных определяет...

- A) множество значений, которые могут принимать объекты программы
- Б) множество значений, допустимых для операций программы
- В) множество значений, которые могут принимать объекты программы, а также совокупность операций, допустимых над этими значениями
- Г) совокупность операций, допустимых над этими значениями
- Д) совокупность операций, допустимых над объектами программы

25. Система программирования - это ...

- A) устройство для создания компьютерных программ
- Б) специальная программа, предназначенная для создания компьютерных программ
- В) программное обеспечение компьютера, предназначенное для разработки, отладки и исполнения программ, записанных на определённом языке программирования
- Г) совокупность программ на определённом языке программирования

26. Установите соответствия

|                            |  |                                                                            |
|----------------------------|--|----------------------------------------------------------------------------|
| система команд исполнителя |  | Алгоритм, записанный на специальном языке, понятном компьютеру, - на языке |
| программа                  |  |                                                                            |

|                    |  |                                                                                         |
|--------------------|--|-----------------------------------------------------------------------------------------|
|                    |  | программирования,<br>называется...                                                      |
| блок-схема         |  | Графическое описание<br>алгоритмов как<br>последовательности<br>действий называется ... |
| компьютерная среда |  |                                                                                         |

27. Данный алгоритм

```
For (i=1;i<=10;i++)
```

```
{
```

```
 A[i]:=10 - i + 1;
```

```
} формирует массив A из 10 элементов вида ...
```

|                                    |                                    |
|------------------------------------|------------------------------------|
| А) (10, 9, 8, 7, 6, 5, 4, 3, 2, 1) | В) (1, 2, 3, 4, 5, 6, 7, 8, 9, 10) |
| Б) (9, 8, 7, 6, 5, 4, 3, 2, 1, 0)  | Г) (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)  |

28. Ниже перечислены основные свойства алгоритма. Некоторые из этих понятий не относятся к основным свойствам алгоритма. Укажите, какие именно.

А) дискретность

Г) результативность

Е) строгость

Б) определенность

Д) массовость

Ж) секретность

В) актуальность

29. Элементы массивов в языке C# нумеруются с ...

|      |                                               |      |
|------|-----------------------------------------------|------|
| А) 0 | Б) с заданного при объявлении массива индекса | В) 1 |
|------|-----------------------------------------------|------|

30. Нужно объявить массив из 11 элементов типа int. Какое из объявлений верно?

|                             |                             |
|-----------------------------|-----------------------------|
| А) int a [] = new int [11]; | В) int [] a = new int [11]; |
| Б) int [] a = new int [10]; | Г) int a [] = new int [10]; |

31. Что покажет приведенный ниже код?

```
text='beegeek';
```

```
text.upper();
```

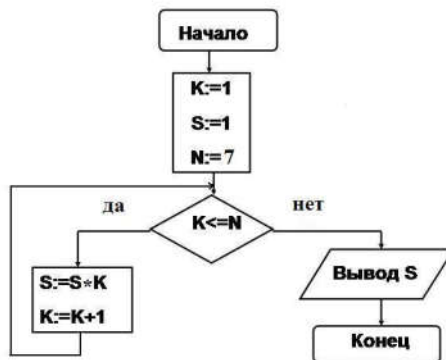
```
Concole.Write(text);
```

|            |            |                      |
|------------|------------|----------------------|
| А) beegeek | В) BEEGEEK | Д) произойдет ошибка |
| Б) Beegeek | Г) beegeeK |                      |

32. Дан массив  $\text{int } L[3,3] = \{ \{ 2, 3, 4 \}, \{ 3, 4, 8 \}, \{ 1, 0, 9 \} \}$ . Чему будет равно значение элемента этого массива  $L[1,2]$

|      |      |      |      |
|------|------|------|------|
| А) 2 | Б) 3 | В) 4 | Г) 8 |
|------|------|------|------|

33. Алгоритмическая конструкция какого типа изображена на фрагменте блок-схемы, представленном на рисунке



Ответ: \_\_\_\_\_

34. Команда, которая позволяет сравнивать две строки ...

|               |             |
|---------------|-------------|
| A) Contains() | B) Split()  |
| Б) Compare()  | Г) Concat() |

35. Что выведет на экран фрагмент программы

```

string text = "хороший день";

text = text.Replace("хороший", "плохой");
Console.WriteLine(text);

```

Ответ: \_\_\_\_\_

36. С помощью какой команды можно изменить порядок элементов на обратный в массиве ...

|             |              |
|-------------|--------------|
| A) Clear()  | B) IndexOf() |
| Б) Length() | Г) Reverse() |

**Ответы:**

| 1          | 2         | 3           | 4         | 5           | 6         | 7         | 8         | 9         | 10        |
|------------|-----------|-------------|-----------|-------------|-----------|-----------|-----------|-----------|-----------|
| 1Д2Б3Г4А5Б | 1Г2В3А4Б  | 2           | 1         | 4           | 1         | 2         | 3         | 3         | А         |
| <b>11</b>  | <b>12</b> | <b>13</b>   | <b>14</b> | <b>15</b>   | <b>16</b> | <b>17</b> | <b>18</b> | <b>19</b> | <b>20</b> |
| Г          | Б         | трансляция  | отладка   | одного      | А         | В         | Б         | А         | Б         |
| <b>21</b>  | <b>22</b> | <b>23</b>   | <b>24</b> | <b>25</b>   | <b>26</b> | <b>27</b> | <b>28</b> | <b>29</b> | <b>30</b> |
| Б          | В         | А           | В         | В           | 1А2А3Б4Б  | А         | Ж         | А         | В         |
| <b>31</b>  | <b>32</b> | <b>33</b>   | <b>34</b> | <b>35</b>   | <b>36</b> |           |           |           |           |
| В          | Г         | циклический | Б         | Плохой день | Г         |           |           |           |           |

## Типовые задания для оценки освоения Раздела 3 Структурное программирование

Обучающийся должен

**знать:**

- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;
- понятие подпрограммы, принципы составления библиотек подпрограмм;

**уметь:**

- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- оформлять код программы в соответствии со стандартом кодирования;
- выполнять проверку, отладку кода программы.

### Типовые вопросы для устного опроса

1. Что такое подпрограмма?
2. Для чего нужны подпрограммы?
3. В чем заключается различие между процедурами и функциями?
4. Что такое рекурсивная функция?
5. Какие основные операторы используется при работе с файлами
6. Какие существуют методы структурного программирования
7. Назовите стандартные модули в C#
8. Что такое указатели?

### Типовой тест

**Условия выполнения задания**

- тест выполняется в аудитории во время практических занятий;
- для выполнения теста необходимо следующее оборудование: бланки ответов, ручки, карточки с тестами (для выполнения электронного варианта теста: компьютерный класс, тестировщик).

**Инструкция:** на выполнение теста отводится 15 минут, внимательно прочитайте вопрос, выберите один вариант ответа, ответы занесите в бланк ответов.



1. Процедуры и функции — это...
  - а) операторы
  - б) подпрограммы
  - в) имена
  - г) переменные
2. Файл — это...
  - а) база данных
  - б) поименованный участок на внешних носителях памяти
  - в) список
  - г) раздел на жестком диске
3. Рекурсия – это ...
  - а) повторение выполнения функции или процедуры внутри себя
  - б) оператор
  - в) цикл
  - г) метод определения функции или процедуры
4. Глобальные переменные действуют
  - а) во всех процедурах
  - б) во всех функциях
  - в) во всех модулях
  - г) во всей программе
5. Что такое структурное программирование?
  - А) Технология разработки программного обеспечения, основой которой служит представление программы в виде математических формул и символов естественного языка
  - Б) Технология разработки программного обеспечения, основой которой служит представление программы в виде иерархической конструкции логических целостных фрагментов
  - В) Технология разработки программного обеспечения, основой которой служит представление программы в виде логарифмических уравнений
6. Как называется алгоритм, который на каком-либо шаге обращается сам к себе?  
 Ответ: \_\_\_\_\_

**Ответы:**

| 1 | 2 | 3 | 4 | 5 | 6               |
|---|---|---|---|---|-----------------|
| Б | Б | А | Г | Б | вспомогательный |

## Типовые задания для оценки освоения

### Раздела 4 Введение в объектно-ориентированное программирование

Обучающийся должен

**знать:**

– объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения;

**уметь:**

- определять сложность работы алгоритмов;
- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- оформлять код программы в соответствии со стандартом кодирования;
- выполнять проверку, отладку кода программы.

### Типовые вопросы для устного опроса

1. Что такое объект?
2. Что такое объектно-ориентированное программирование?
3. Что такое инкапсуляция?
4. Что такое наследование?
5. Что такое полиморфизм?
6. Что такое тестирование приложения?
7. Что такое перегрузка методов?
8. Что такое сериализация?

### Типовой тест

**Условия выполнения задания**

- *тест выполняется в аудитории во время практических занятий;*
- *для выполнения теста необходимо следующее оборудование: бланки ответов, ручки, карточки с тестами (для выполнения электронного варианта теста: компьютерный класс, тестировщик).*

**Инструкция:** на выполнение теста отводится 15 минут, внимательно прочитайте вопрос, выберите один вариант ответа, ответы занесите в бланк ответов.

1. Свойство объектов, при котором действие с одинаковыми именами вызывает различное поведение для различных объектов
  - A) Полиморфизм
  - В) Наследование
  - Б) Передача
  - Г) Монорфизм
2. Данные, характеризующие состояние объекта
  - А) Доли объекта
  - Б) Части объекта
  - В) Атрибуты объекта
3. Совокупность объектов, характеризующаяся общностью методов и свойств
  - А) Класс
  - Б) Вид
  - В) Род
4. Действие, которое может выполнить объект
  - А) Метод
  - Б) Событие
  - В) Свойство
5. Совокупность свойств и методов
  - А) Объект;
  - Б) Свойство;
  - В) Событие.
6. Компоненты, которые видны во время работы приложения, с ними напрямую может взаимодействовать пользователь, называются ...
  - А) Абстрактными
  - Б) Видимыми
  - В) Визуальными
7. Какая из перечисленных функций не может быть конструктором
  - А) String (const int a)
  - Б) String (String & s)
  - В) void String ()
8. Свойство Name отвечает за ...
  - А) Название компонента
  - Б) Имя компонента
  - В) Назначение компонента
9. Этот компонент предназначен для вывода текста на поверхность формы.
  - А) Label
  - Б) Edit
  - В) Button
10. Скрытие деталей реализации называется ...
  - А) Инкапсуляцией
  - Б) Наследование
  - В) Полиморфизм
  - Г) Абстракцией

**Ответы:**

| <b>№ вопроса</b> | <b>Ответ</b> | <b>№ вопроса</b> | <b>Ответ</b> |
|------------------|--------------|------------------|--------------|
| <b>1</b>         | <b>A</b>     | <b>6</b>         | <b>B</b>     |
| <b>2</b>         | <b>B</b>     | <b>7</b>         | <b>B</b>     |
| <b>3</b>         | <b>A</b>     | <b>8</b>         | <b>A</b>     |
| <b>4</b>         | <b>A</b>     | <b>9</b>         | <b>A</b>     |
| <b>5</b>         | <b>A</b>     | <b>10</b>        | <b>A</b>     |

## **5 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

Изучение учебной дисциплины осуществляется в течение двух семестров.

Промежуточные аттестации проводятся в формах: **дифференцированного зачета (2 семестр), экзамена (3 семестр).**

Дифференцированный зачет проводится по завершению курса изучения учебной дисциплины в форме выполнения практического задания с последующим собеседованием с преподавателем с учетом результатов текущего контроля.

Экзамен проводится по завершению курса изучения учебной дисциплины по билетам, содержащим два теоретических вопроса и практическое задание.

### **Вопросы для подготовки к дифференцированному зачету**

#### **Перечень вопросов для оценки усвоенных знаний**

- 1 Развитие языков программирования. Обзор языков программирования.
- 2 Области применения языков программирования.
- 3 Стандарты языков программирования.
- 4 Среда проектирования.
- 5 Компиляторы и интерпретаторы.
- 6 Жизненный цикл программы
- 7 Программа. Программный продукт и его характеристики.
- 8 Основные этапы решения задач на компьютере
- 9 Переменные: объявление переменных, именование переменных, пространство имен.
- 10 Система типов языка C#. Встроенные типы. Типы CTS.
- 11 Преобразование типов: неявное преобразование типа, явное преобразование типа, операции преобразования для данных строкового типа.
- 12 Консольный ввод и вывод
- 13 Структурная схема программы на алгоритмическом языке
- 14 Лексика языка C#. Переменные и конспекты. Выражения и операции.
- 15 Операторы языка C#: операторы объявления, операторы выражения.
- 16 Синтаксис операторов C#
- 17 Составной оператор. Вложенные условные операторы. Оператор выбора.
- 18 Применение алгебры логики при создании алгоритмов
- 19 Цикл с постусловием.
- 20 Цикл с предусловием.

- 21 Цикл с параметром
- 22 Вложенные циклы. Операторы перехода: break, continue, goto, return
- 23 Структуры данных. Массивы. Объявление массива. Ввод и вывод одномерных и двумерных массивов.
- 24 Обработка массивов. Стандартные функции для массива целых и вещественных чисел
- 25 Простейшие алгоритмы поиска в массиве.
- 26 Алгоритмы сортировки в массиве
- 27 Строки и множества. Объявление строковых типов данных.
- 28 Операции со строками. Стандартные функции и процедуры работы со строками
- 29 Объявление множества. Операции над множествами

### Перечень заданий для оценивания освоенных умений

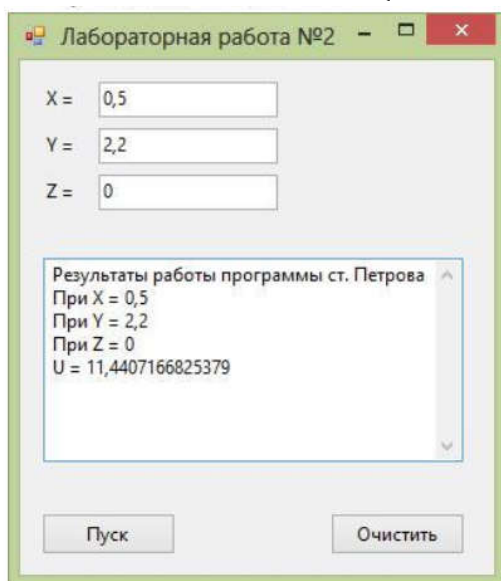
1. Разместите на форме четыре кнопки (Button). Сделайте на кнопках следующие надписи: «красный», «зеленый», «синий», «желтый». Создайте четыре обработчика события нажатия на данные кнопки, которые будут менять цвет формы в соответствии с текстом на кнопках.

2. Составить программу вычисления для заданных значений  $x$ ,  $y$ ,  $z$  арифметического выражения:

$$u = tg^2(x + y) - e^{y-z} \sqrt{\cos x^2 + \sin z^2}$$

3. Создать форму, которая выполняет вычисления для функции

$$u = \begin{cases} y \sin(x) + z, & z - x = 0 \\ ye^{\sin(x)} - z, & z - x < 0 \\ y \sin(\sin(x)) + z, & z - x > 0 \end{cases}$$



4. Вычислить и вывести на экран таблицу значений функции  $y = a \cdot \ln(x)$  при  $x$ , изменяющемся от  $x_0$  до  $x_k$  с шагом  $dx$ ,  $a$  – константа

5. В массиве из 20 целых чисел найти наибольший элемент и поменять его местами с первым элементом

6. Дана матрица  $A(3,3)$ . Вычислить сумму второй строки и произведение первого столбца. Вывести исходную матрицу и результаты вычислений

7. Написать программу подсчета числа слов в произвольной строке. В качестве разделителя может быть любое число пробелов. Для ввода строк использовать ListVox. Строки вводятся на этапе проектирования формы, используя окно свойств. Вывод результата организовать в метку Label.

### **Вопросы для подготовки к экзамену**

#### **Перечень вопросов для оценки усвоенных знаний**

- 1 Понятие подпрограммы. Процедуры и функции, их сущность, назначение и различие.
- 2 Организация процедур, стандартные процедуры.
- 3 Процедуры, определенные пользователем: синтаксис, передача аргументов.
- 4 Формальные и фактические параметры.
- 5 Процедуры с параметрами, описание процедур.
- 6 Рекурсивные функции.
- 7 Типы файлов. Организация доступа к файлам.
- 8 Файлы последовательного доступа. Открытие и закрытие файла последовательного доступа.
- 9 Файлы произвольного доступа. Порядок работы с файлами произвольного доступа
- 10 Основы структурного программирования. Методы структурного программирования.
- 11 Стили структурного программирования. Разработка программы
- 12 Понятие модуля. Структура модуля.
- 13 Компиляция и компоновка программы.
- 14 Стандартные модули
- 15 Описание указателей. Основные понятия и применение динамически распределяемой памяти.
- 16 Создание и удаление динамических переменных. Структуры данных на основе указателей.
- 17 Задача о стеке
- 18 История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс

- 19 Основные принципы ООП: инкапсуляция, наследование, полиморфизм.
- 20 Понятие наследования. Особенности базового и производственного классов.
- 21 Доступ к элементам класса и наследование
- 22 Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Понятие полиморфизма.
- 23 Ссылка на объект базового класса и объекты производственных классов. Использование ссылки на объект базового класса
- 24 Виртуальные методы. Динамическое связывание.
- 25 Основные этапы реализации полиморфизма.
- 26 Приведение типов объектов. Абстрактные методы
- 27 Классы объектов. Компоненты и их свойства.
- 28 Событийно-управляемая модель программирования.
- 29 Компонентно-ориентированный подход
- 30 Требования к аппаратным и программным средствам интегрированной среды разработчика
- 31 Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты.
- 32 Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта
- 33 Состав и характеристика проекта.
- 34 Выполнение проекта. Настройка среды и параметров проекта
- 35 Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение.
- 36 Дополнительные элементы управления. Свойства компонентов. Виды свойств. Синтаксис определения свойств. Назначения свойств и их влияние на результат
- 37 Управление объектом через свойства. События компонентов (элементов управления), их сущность и назначение
- 38 Создание процедур на основе событий
- 39 Диалоги. Класс диалогов. Формы. Хеширование
- 40 Разработка функционального интерфейса приложения.
- 41 Создание интерфейса приложения
- 42 Разработка функциональной схемы работы приложения.
- 43 Разработка игрового приложения
- 44 Разработка приложения. Проектирование объектно-ориентированного приложения.
- 45 Создание интерфейса пользователя.



- 46 Тестирование, отладка приложения
- 47 Классы ООП: виды, назначение, свойства, методы, события.
- 48 Перегрузка методов.
- 49 Создание многоуровневой иерархии. Абстрактные классы
- 50 Понятие сериализации. Сериализация объектов в бинарный файл.
- 51 Десериализация объектов из бинарного файла
- 52 Организация взаимодействия объектов. Реализация взаимодействия объектов

### **Перечень заданий для оценивания освоенных умений**

1. Составить программу с тремя функциями, выводящих на экран символы #, @, \$
2. Составить программу для вычисления площади кольца по значениям внутреннего и внешнего радиусов, используя функцию вычисления площади круга. Функции передается один параметр – радиус и возвращается результат - вычисленная площадь.
3. Напишите рекурсивную функцию, которая переворачивает элементы массива и обратном порядке.
4. На основе базового класса «Автомобиль», создать произвольный класс «Общественный транспорт».
5. Создать функции, определяющие максимальное и минимальное значение переменных каждого объекта. Используйте конструктор.
6. Разработать приложение «Калькулятор».
7. Расположить на форме четыре панели для отображения текущего времени, даты, названия дня недели и панель с секундомером. Подобрать цвета, стиль и расположение панелей на форме.
8. Разработать приложение «Конвертор величин измерения» с использованием компонента ввода строки Edit. Создать программный код необходимых событий.

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

| <b>№<br/>п.п.</b> | <b>Содержание изменения</b> | <b>Дата, номер<br/>протокола заседания ПЦК.<br/>Подпись председателя ПЦК</b> |
|-------------------|-----------------------------|------------------------------------------------------------------------------|
|                   |                             |                                                                              |