

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ ВЛАДИМИРА ДАЛЯ»

Факультет компьютерных систем и информационных технологий

Кафедра информатики и программной инженерии

УТВЕРЖДАЮ

Декан факультета компьютерных
систем и информационных технологий

Кочевский А. А.

« 19 » 04 2023 г.

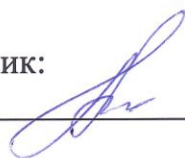
ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по учебной дисциплине

«Встроенные алгоритмические языки»

09.04.03 Прикладная информатика

«Прикладная информатика в экономике»

Разработчик:

доцент  Петрущенко Т. В.

ФОС рассмотрен и одобрен на заседании кафедры информатики и
программной инженерии от 18 апреля 2023 г., протокол № 17

Заведующий кафедрой  Кочевский А.А.

Луганск 2023 г.

**Паспорт
фонда оценочных средств по учебной дисциплине
«Встроенные алгоритмические языки»**

**Перечень компетенций (элементов компетенций),
формируемых в результате освоения учебной дисциплины**

№ п/п	Код контролируемой компетенции	Формулировка контролируемой компетенции	Контролируемые темы учебной дисциплины	Этапы формирования (семестр изучения)
1	ПК-1	Способен применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов и создания информационных систем	Тема 1. PHP. Основные понятия. Тема 2. MySQL. Основные понятия Тема 3. Использование PHP Тема 4. Работа с MySQL Тема 5. Варианты построения сети.	1

**Показатели и критерии оценивания компетенций,
описание шкал оценивания**

№ п/п	Код контролируемой компетенции	Показатель оценивания (знания, умения, навыки)	Контролируемые темы учебной дисциплины	Наименование оценочного средства
1	ПК-1	Знать: современные методы и инструментальные средства прикладной информатики. Уметь: применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач. Владеть: современными методами и инструментальными средствами прикладной информатики для	Тема 1. Тема 2. Тема 3. Тема 4.	Лабораторные работы; контрольные работы; промежуточная аттестация (зачет)

		автоматизации и информатизации решения прикладных задач различных классов и создания ИС.		
--	--	--	--	--

Фонды оценочных средств по дисциплине «Встроенные алгоритмические языки»

Лабораторные работы:

Типовые задания к лабораторным работам

Лабораторная работа 1. Практика программирования скриптов на PHP

Объявить переменную каждого из поддерживаемых PHP типов данных, исследовать поведение PHP при выполнении всех видов арифметических операций со всеми объявленными переменными.

Лабораторная работа 2. Объектно-ориентированное программирование на PHP (часть 1)

Создать класс Users со свойствами name, login, password; создать три объекта от класса Users; задать произвольные значения свойствам для каждого из объектов.

Лабораторная работа 3. Объектно-ориентированное программирование на PHP (часть 2).

В классе Users необходимо описать конструктор; конструктор должен задавать начальные значения свойств name, login, password; создать заново три объекта, экземпляра класса Users;

Лабораторная работа 4. Объектно-ориентированное программирование на PHP (часть 3).

Описать класс SuperUsers унаследованный от класса Users. В классе SuperUsers описать свойство character и создать объект, экземпляр класса SuperUsers. Задать значение свойству character = «admin». Вызвать метод getInfo() для созданного объекта. Отдельно от метода getInfo() вывести значение свойства character;

Лабораторная работа №5. Объектно-ориентированное программирование на PHP (часть 4).

Сделать все параметры конструктора класса Users параметрами по умолчанию со значением «пустая строка» («»).

В конструкторе класса Users сгенерировать исключение, если введены не все данные.

Описать перехват исключения и вывести в браузер сообщение об ошибке.

Создать экземпляр класса Users без какого-либо параметра.

Критерии и шкала оценивания по оценочному средству лабораторная работа

Шкала оценивания (интервал баллов)	Критерий оценивания
5	Лабораторная работа выполнена на высоком уровне (правильность выполнения 90-100%)
4	Лабораторная работа выполнена на среднем уровне (правильность выполнения 75-89%)
3	Лабораторная работа выполнена на низком уровне (правильность выполнения 50-74%)
2	Лабораторная работа выполнена на неудовлетворительном уровне (правильность выполнения менее чем на 50%)

Контрольные работы:

Типовые вопросы к контрольным работам

1. Назначение и область применения языка PHP. Режимы работы PHP-приложений. Консольные приложения, аргументы командной строки. Запуск приложения с правами пользователя `wwwrun`, основные проблемы доступа к каталогам и файлам. Расширенные списки доступа ACL. Пояснить на примерах.
2. Основы синтаксиса языка PHP. Структура простейшей программы. Переменные. Константы. Предопределенные константы. Ссылки в PHP, примеры. Элементы структурного программирования (ветвление, организация циклов, передача управления).
3. Использование массивов в PHP. Способы создания/удаления массива. Индексные массивы (списки), правило вычисления индекса, переиндексация, обращение к элементу массива. Ассоциативные массивы, обращение к элементу массива. Обращение к элементу массива внутри строки. Цикл `foreach`.
4. Функции в PHP. Синтаксис описания функций. Порядок определения и вызова. Способы передачи аргументов, значения по умолчанию. Список аргументов переменной длины. Переменные внутри функции: локальные, глобальные, статические. Стек вызовов функций, функции вывода стека. Пояснить на примерах. Средства PHP для отправки/принятия HTTP-сообщений. Использование HTTP-контекста.
5. Схема работы Web-приложения, архитектура клиент-сервер. Протокол HTTP. Типы HTTP-соединений. Общая структура HTTP-сообщений (запрос, ответ). Выбор метода: GET или POST? Кодирование в URL.
6. Обработка данных HTML-форм. Способы передачи параметров сценарию PHP. Обращение к параметру, переданному из формы. Обработка списка с множественным выбором (`<select>`), параметр-массив. Обработка специальных случаев: якоря (`#`), имена с точкой (`name=picture.gif`), магические кавычки. Обработка `<input type="image">`..., доступ к координатам.

7. Cookie-наборы, назначение, время жизни. Схема обмена заголовками между клиентом и сервером (Cookie - в запросе, Set-Cookie - в ответе). Способы установки cookie в php-скрипте. Установка/изменение/удаления cookie-набора с помощью setcookie (пример). Доступ к значениям cookie-набора в php-скрипте.
8. Сессии (сеансы) PHP, основная идея использования сессий, начало сеанса, завершение сеанса. Идентификатор сессии, способ передачи идентификатора от страницы к странице. Переменные сессии. HTTP-аутентификация, способы, защита одной страницы, защита нескольких страниц.
9. Многофайловые сценарии PHP. Включение файлов, принцип работы include, include_once, require, require_once. Загрузка файлов на сервер (upload). Запуск приложений из PHP. Выделение статичной информации, способы. Область использования .htaccess.
10. Создание класса, синтаксис описания. Создание объекта, представление объекта по ссылке. Клонирование, метод __clone. Область видимости private, protected, public. Конструктор класса, передача параметров в конструктор, вызов родительского конструктора. Деструктор класса, вызов родительского деструктора. Константы класса, определение, обращение внутри класса, обращение вне класса. Пояснить на примерах.

Критерии и шкала оценивания по оценочному средству «контрольная работа»

Шкала оценивания	Критерий оценивания
5	Контрольная работа выполнена на высоком уровне (правильные ответы даны на 90-100% вопросов/задач)
4	Контрольная работа выполнена на среднем уровне (правильные ответы даны на 75-89% вопросов/задач)
3	Контрольная работа выполнена на низком уровне (правильные ответы даны на 50-74% вопросов/задач)
2	Контрольная работа выполнена на неудовлетворительном уровне (правильные ответы даны менее чем на 50%)

Оценочные средства для промежуточной аттестации (зачет)

Типовые вопросы к зачету

1. Назначение и область применения языка PHP.
2. Режимы работы PHP-приложений.
3. Консольные приложения, аргументы командной строки.
4. Запуск приложения с правами пользователя `wwwrun`, основные проблемы доступа к каталогам и файлам.
5. Расширенные списки доступа ACL. Пояснить на примерах.
6. Основы синтаксиса языка PHP. Структура простейшей программы. Переменные. Константы. Предопределенные константы. Ссылки в PHP, примеры. Элементы структурного программирования (ветвление, организация циклов, передача управления).
7. Использование массивов в PHP.
8. Способы создания/удаления массива. Индексные массивы (списки), правило вычисления индекса, переиндексация, обращение к элементу массива. Ассоциативные массивы, обращение к элементу массива. Обращение к элементу массива внутри строки. Цикл `foreach`.
9. Функции в PHP. Синтаксис описания функций.
10. Порядок определения и вызова. Способы передачи аргументов, значения по умолчанию. Список аргументов переменной длины. Переменные внутри функции: локальные, глобальные, статические. Стек вызовов функций, функции вывода стека. Пояснить на примерах. Средства PHP для отправки/принятия HTTP-сообщений. Использование HTTP-контекста.
11. Схема работы Web-приложения, архитектура клиент-сервер.
12. Протокол HTTP. Типы HTTP-соединений. Общая структура HTTP-сообщений (запрос, ответ).
13. Выбор метода: GET или POST? Кодирование в URL.
14. Обработка данных HTML-форм.
15. Способы передачи параметров сценарию PHP. Обращение к параметру, переданному из формы. Обработка списка с множественным выбором (`<select>`), параметр-массив.
16. Обработка специальных случаев: якоря (`#`), имена с точкой (`name=picture.gif`), магические кавычки. Обработка `<input type="image">`..., доступ к координатам.
17. Cookie-наборы, назначение, время жизни.
18. Схема обмена заголовками между клиентом и сервером (Cookie - в запросе, Set-Cookie - в ответе). Способы установки cookie в PHP-скрипте. Установка/изменение/удаления cookie-набора с помощью `setcookie` (пример).
19. Доступ к значениям cookie-набора в PHP-скрипте.
20. Сессии (сеансы) PHP, основная идея использования сессий, начало сеанса, завершение сеанса. Идентификатор сессии, способ передачи

- идентификатора от страницы к странице. Переменные сессии. HTTP-аутентификация, способы, защита одной страницы, защита нескольких страниц.
21. Многофайловые сценарии PHP.
 22. Включение файлов, принцип работы `include`, `include_once`, `require`, `require_once`. Загрузка файлов на сервер (`upload`). Запуск приложений из PHP. Выделение статичной информации, способы. Область использования `.htaccess`.
 23. Создание класса, синтаксис описания.
 24. Создание объекта, представление объекта по ссылке. Клонирование, метод `__clone`. Область видимости `private`, `protected`, `public`. Конструктор класса, передача параметров в конструктор, вызов родительского конструктора.
 25. Деструктор класса, вызов родительского деструктора. Константы класса, определение, обращение внутри класса, обращение вне класса. Пояснить на примерах.
 26. Статические свойства и методы класса.
 27. Наследование, модель одиночного наследования. Синтаксис описания класса-потомка. Перегрузка метода класса, предпочтение локального (`self`) конструктора. Управление наследованием с помощью `final`. Абстрактный метод, абстрактный класс. Интерфейсы, назначение, синтаксис описания, реализация. Пояснить на примерах.
 28. Архитектура MVC, описание, назначение основных компонентов. Преимущества MVC. Что такое Smarty?
 29. Принцип работы Smarty. Назначение переменных и массивов в php (`assign`), обращение к ним в шаблоне. Пример использования встроенной функции `{foreach}`.
 30. Переменная `{$_smarty}`. Работа с конфигурационными файлами Smarty, загрузка скрытых секций. Пояснить на примерах.
 31. Основные принципы организации реляционных БД.
 32. Связь PHP и MySQL в контексте LAMP. Создание БД, права суперпользователя. Создание, удаление, изменение таблиц, получение информации. Работа с СУБД MySQL из PHP. `Mysqli`, процедурный и ООП-синтаксис, SQL-запросы, использование функций в запросах. Кодировки клиента, соединения, результата и базы. Примеры.
 33. Модель обработки ошибок в PHP 5. Типы ошибок. Алгоритм обработки ошибок.
 34. Стандартные обработчик ошибок. Подавление ошибок. Обработка исключений в PHP 5. Особенности создания собственного обработчика. Способы отладки сценариев. Оптимизация php-программ.
 35. Отладка сценариев PHP. Средства языка, используемые для отладки. Трассировка. Журналирование. Использование утверждений (`assert`).
 36. Настройки безопасности Apache, права и расширение возможностей Web-сервера, сокращение PHP.

Критерии и шкала оценивания по оценочному средству промежуточный контроль (экзамен)

Шкала оценивания	Критерий оценивания
зачтено	Студент глубоко и в полном объёме владеет программным материалом. Грамотно, исчерпывающе и логично его излагает в устной или письменной форме. При этом знает рекомендованную литературу, проявляет творческий подход в ответах на вопросы и правильно обосновывает принятые решения, хорошо владеет умениями и навыками при выполнении практических задач.
	Студент знает программный материал, грамотно и по сути излагает его в устной или письменной форме, допуская незначительные неточности в утверждениях, трактовках, определениях и категориях или незначительное количество ошибок. При этом владеет необходимыми умениями и навыками при выполнении практических задач.
	Студент знает только основной программный материал, допускает неточности, недостаточно чёткие формулировки, непоследовательность в ответах, излагаемых в устной или письменной форме. При этом недостаточно владеет умениями и навыками при выполнении практических задач. Допускает до 30% ошибок в излагаемых ответах.
не зачтено	Студент не знает значительной части программного материала. При этом допускает принципиальные ошибки в доказательствах, в трактовке понятий и категорий, проявляет низкую культуру знаний, не владеет основными умениями и навыками при выполнении практических задач. Студент отказывается от ответов на дополнительные вопросы

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)

Экспертное заключение

Представленный фонд оценочных средств (далее - ФОС) по дисциплине *«Встроенные алгоритмические языки»* соответствует требованиям ГОС ВО.

Предлагаемые формы и средства текущего и промежуточного контроля адекватны целям и задачам реализации основной образовательной программы по направлению подготовки 09.04.03 Прикладная информатика.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы обучающегося представлены в полном объеме.

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

Разработанный и представленный для экспертизы фонд оценочных средств рекомендуется к использованию в процессе подготовки бакалавров, по указанному направлению.

Председатель учебно-методической
комиссии факультета компьютерных
систем и информационных
технологий



Ветрова Н. Н.