

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
"Казанский (Приволжский) федеральный университет"
Набережночелнинский институт (филиал)
Экономическое отделение



Утверждаю

Заместитель директора
по образовательной деятельности
НЧИ КФУ Н.Д.Ахметов



« _____ » _____ 20__ г.

подписано электронно-цифровой подписью

Программа дисциплины

Web-программирование

Направление подготовки: 09.03.03 - Прикладная информатика

Профиль подготовки: Прикладная информатика в экономике

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2020

Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО
2. Место дисциплины (модуля) в структуре ОПОП ВО
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий
 - 4.1. Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)
 - 4.2. Содержание дисциплины (модуля)
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
6. Фонд оценочных средств по дисциплине (модулю)
 - 6.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы и форм контроля их освоения
 - 6.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания
 - 6.3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы
- 6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций
7. Перечень литературы, необходимой для освоения дисциплины (модуля)
8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)
12. Средства адаптации преподавания дисциплины (модуля) к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья
13. Приложение №1. Фонд оценочных средств
14. Приложение №2. Перечень литературы, необходимой для освоения дисциплины (модуля)
15. Приложение №3. Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Программу дисциплины разработал(а)(и) доцент, к.н. (доцент) Лысанов Д.М. (Кафедра бизнес-информатики и математических методов в экономике, Экономическое отделение), DMLysanov@kpfu.ru

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО

Обучающийся, освоивший дисциплину (модуль), должен обладать следующими компетенциями:

Шифр компетенции	Расшифровка приобретаемой компетенции
ПК-2	Способен разрабатывать и адаптировать прикладное программное обеспечение

Обучающийся, освоивший дисциплину (модуль):

Должен знать:

ПК-2: знать способы и методы разработки и адаптации прикладного программного обеспечения.

Должен уметь:

ПК-2: уметь разрабатывать и адаптировать прикладное программное обеспечение.

Должен владеть:

ПК-2: владеть способностью разрабатывать и адаптировать прикладное программное обеспечение

Должен демонстрировать способность и готовность:

ПК-2: демонстрировать способность и готовность разрабатывать и адаптировать прикладное программное обеспечение

2. Место дисциплины (модуля) в структуре ОПОП ВО

Данная дисциплина (модуль) включена в раздел "Б1.В.04 Дисциплины (модули)" основной профессиональной образовательной программы 09.03.03 "Прикладная информатика (Прикладная информатика в экономике)" и относится к вариативной части.

Осваивается на 4 курсе в 7, 8 семестрах.

3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 9 зачетных(ые) единиц(ы) на 324 часа(ов).

Контактная работа - 124 часа(ов), в том числе лекции - 38 часа(ов), практические занятия - 0 часа(ов), лабораторные работы - 86 часа(ов), контроль самостоятельной работы - 0 часа(ов).

Самостоятельная работа - 137 часа(ов).

Контроль (зачёт / экзамен) - 63 часа(ов).

Форма промежуточного контроля дисциплины: экзамен в 7 семестре; экзамен в 8 семестре.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Переменные, константы, типы данных	7	5	0	9	14
2.	Тема 2. Выражения и операции, Работа с данными формы	7	5	0	9	14

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
3.	Тема 3. Конструкции языка	7	4	0	9	13
4.	Тема 4. Ассоциативные массивы	7	4	0	9	13
5.	Тема 5. Функции и области видимости	8	4	0	10	17
6.	Тема 6. Объекты и классы	8	4	0	10	17
7.	Тема 7. Наследование	8	4	0	10	17
8.	Тема 8. Интерфейсы и трейты, Пространство имен	8	4	0	10	16
9.	Тема 9. Обработка ошибок и исключения	8	4	0	10	16
	Итого		38	0	86	137

4.2 Содержание дисциплины (модуля)

Тема 1. Переменные, константы, типы данных

Переменные, Копирование переменных, Типы переменных, Действия с переменными, Определение типа переменной, Установка типа переменной, Оператор присваивания, Ссылочные переменные, Жесткие ссылки, Символические ссылки, Ссылки на объекты, Некоторые условные обозначения, Константы, Предопределенные константы, Определение констант, Проверка существования константы, Константы с динамическими именами, Отладочные функции

Тема 2. Выражения и операции, Работа с данными формы

Выражения, Логические выражения, Строковые выражения, Операции, Арифметические операции, Строковые операции, Операции присваивания, Операции инкремента и декремента, Битовые операции, Операции сравнения, Логические операции, Операция отключения предупреждений, Условные операции, Передача данных командной строки, Формы, Трансляция полей формы, Трансляция переменных окружения, Трансляция cookies, Обработка списков, Обработка массивов, Диагностика, Порядок трансляции переменных, Особенности флажков checkbox

Тема 3. Конструкции языка

Инструкция if-else, Использование альтернативного синтаксиса, Цикл с предусловием while, Цикл с постусловием do-while, Универсальный цикл for, Инструкции break и continue, Нетрадиционное использование do-while и break, Цикл foreach, Конструкция switch-case, Инструкции goto, Инструкции require и include, Инструкции однократного включения

Тема 4. Ассоциативные массивы

Создание массива "на лету". Автомассивы, Конструкция list(), Списки и ассоциативные массивы, Конструкция array() и многомерные массивы, Массивы-константы, Операции над массивами, Доступ по ключу, Функция count(), Слияние массивов, Косвенный перебор элементов массива, Прямой перебор массива, Списки и строки, Сериализация, Упаковка, Распаковка

Тема 5. Функции и области видимости

Общий синтаксис определения функции, Инструкция return, Объявление и вызов функции, Параметры по умолчанию, Передача параметров по ссылке, Переменное число параметров, Типы аргументов и возвращаемого значения, Локальные переменные, Глобальные переменные, Массив \$GLOBALS, Как работает инструкция global, Статические переменные, Рекурсия, Факториал, Пример функции: dumper(), Вложенные функции, Условно определяемые функции, Эмуляция функции virtual(), Передача функций по ссылке, Использование call_user_func(), Использование call_user_func_array(), Анонимные функции, Замыкания, Возврат функцией ссылки, Технология отложенного копирования

Тема 6. Объекты и классы

Класс как тип данных, Создание нового класса, Работа с классами, Создание объекта некоторого класса, Доступ к свойствам объекта, Доступ к методам, Создание нескольких объектов, Перегрузка преобразования в строку, Инициализация и разрушение, Конструктор, Деструктор, Алгоритм сбора мусора, Права доступа к членам класса, Модификаторы доступа, Неявное объявление свойств, Общие рекомендации, Класс - self, объект - \$this, счетчик объектов, кэш ресурсов, Константы класса, Перехват обращений к членам класса, Клонирование объектов, Переопределение операции клонирования, Запрет клонирования, Перехват сериализации, Сериализация объектов, Упаковка и распаковка объектов, Методы sleep() и wakeup()

Тема 7. Наследование

Расширение класса, Метод включения, Недостатки метода, Несовместимость типов, Наследование, Переопределение методов, Модификаторы доступа при переопределении, Доступ к методам базового класса, Финальные методы, Запрет наследования, Константы `__CLASS__` и `__METHOD__`, Позднее статическое связывание, Анонимные классы, Полиморфизм, Абстрагирование, Виртуальные методы, Расширение иерархии, Абстрактные классы и методы, Совместимость родственных типов, Уточнение типа в функциях, Оператор `instanceof`, Обратное преобразование типа

Тема 8. Интерфейсы и трейты, Пространство имен

Сложности реализации множественного наследования, Интерфейсы, Наследование интерфейсов, Интерфейсы и абстрактные классы, Трейты, Трейты и наследование, Проблема именования, Объявление пространства имен, Иерархия пространства имен, Импортирование, Автозагрузка классов, Функция `__autoload()`, Функция `spl_autoload_register()`

Тема 9. Обработка ошибок и исключения

Что такое ошибка?, Роли ошибок, Виды ошибок, Контроль ошибок, Директивы контроля ошибок, Установка режима вывода ошибок, Оператор отключения ошибок, Перехват ошибок, Генерация ошибок, Стек вызовов функций, Исключения, Базовый синтаксис, Инструкция `throw`, Раскрутка стека, Исключения и деструкторы, Исключения и `set error handler()`, Классификация и наследование, Блоки-финализаторы, Трансформация ошибок, Серьезность "несерьезных" ошибок, Преобразование ошибок в исключения

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства образования и науки Российской Федерации от 5 апреля 2017 года №301)

Письмо Министерства образования Российской Федерации №14-55-996ин/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений"

Устав федерального государственного автономного образовательного учреждения "Казанский (Приволжский) федеральный университет"

Правила внутреннего распорядка федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет"

Локальные нормативные акты Казанского (Приволжского) федерального университета

6. Фонд оценочных средств по дисциплине (модулю)

6.1 Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы и форм контроля их освоения

Этап	Форма контроля	Оцениваемые компетенции	Темы (разделы) дисциплины
Семестр 7			
	<i>Текущий контроль</i>		
1	Лабораторные работы	ПК-2	1. Переменные, константы, типы данных 2. Выражения и операции, Работа с данными формы 3. Конструкции языка 4. Ассоциативные массивы
2	Устный опрос	ПК-2	1. Переменные, константы, типы данных 2. Выражения и операции, Работа с данными формы 3. Конструкции языка 4. Ассоциативные массивы
3	Тестирование	ПК-2	1. Переменные, константы, типы данных 2. Выражения и операции, Работа с данными формы 3. Конструкции языка 4. Ассоциативные массивы
	<i>Экзамен</i>	ПК-2	
Семестр 8			

Этап	Форма контроля	Оцениваемые компетенции	Темы (разделы) дисциплины
	Текущий контроль		
1	Лабораторные работы	ПК-2	5. Функции и области видимости 6. Объекты и классы 7. Наследование 8. Интерфейсы и трейты, Пространство имен 9. Обработка ошибок и исключения
2	Устный опрос	ПК-2	5. Функции и области видимости 6. Объекты и классы 7. Наследование 8. Интерфейсы и трейты, Пространство имен 9. Обработка ошибок и исключения
3	Тестирование	ПК-2	5. Функции и области видимости 6. Объекты и классы 7. Наследование 8. Интерфейсы и трейты, Пространство имен 9. Обработка ошибок и исключения
	Экзамен		

6.2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Форма контроля	Критерии оценивания				Этап
	Отлично	Хорошо	Удовл.	Неуд.	
Семестр 7					
Текущий контроль					
Лабораторные работы	Оборудование и методы использованы правильно. Проявлена превосходная теоретическая подготовка. Необходимые навыки и умения полностью освоены. Результат лабораторной работы полностью соответствует её целям.	Оборудование и методы использованы в основном правильно. Проявлена хорошая теоретическая подготовка. Необходимые навыки и умения в основном освоены. Результат лабораторной работы в основном соответствует её целям.	Оборудование и методы частично использованы правильно. Проявлена удовлетворительная теоретическая подготовка. Необходимые навыки и умения частично освоены. Результат лабораторной работы частично соответствует её целям.	Оборудование и методы использованы неправильно. Проявлена неудовлетворительная теоретическая подготовка. Необходимые навыки и умения не освоены. Результат лабораторной работы не соответствует её целям.	1
Устный опрос	В ответе качественно раскрыто содержание темы. Ответ хорошо структурирован. Прекрасно освоен понятийный аппарат. Продемонстрирован высокий уровень понимания материала. Превосходное умение формулировать свои мысли, обсуждать дискуссионные положения.	Основные вопросы темы раскрыты. Структура ответа в целом адекватна теме. Хорошо освоен понятийный аппарат. Продемонстрирован хороший уровень понимания материала. Хорошее умение формулировать свои мысли, обсуждать дискуссионные положения.	Тема частично раскрыта. Ответ слабо структурирован. Понятийный аппарат освоен частично. Понимание отдельных положений из материала по теме. Удовлетворительное умение формулировать свои мысли, обсуждать дискуссионные положения.	Тема не раскрыта. Понятийный аппарат освоен неудовлетворительно. Понимание материала фрагментарное или отсутствует. Неумение формулировать свои мысли, обсуждать дискуссионные положения.	2
Тестирование	86% правильных ответов и более.	От 71% до 85 % правильных ответов.	От 56% до 70% правильных ответов.	55% правильных ответов и менее.	3

Форма контроля	Критерии оценивания				Этап
	Отлично	Хорошо	Удовл.	Неуд.	
Экзамен	Обучающийся обнаружил всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоил основную литературу и знаком с дополнительной литературой, рекомендованной программой дисциплины, усвоил взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявил творческие способности в понимании, изложении и использовании учебно-программного материала.	Обучающийся обнаружил полное знание учебно-программного материала, успешно выполнил предусмотренные программой задания, усвоил основную литературу, рекомендованную программой дисциплины, показал систематический характер знаний по дисциплине и способен к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.	Обучающийся обнаружил знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справился с выполнением заданий, предусмотренных программой, знаком с основной литературой, рекомендованной программой дисциплины, допустил погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладает необходимыми знаниями для их устранения под руководством преподавателя.	Обучающийся обнаружил значительные пробелы в знаниях основного учебно-программного материала, допустил принципиальные ошибки в выполнении предусмотренных программой заданий и не способен продолжить обучение или приступить по окончании университета к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.	
Семестр 8					
Текущий контроль					
Лабораторные работы	Оборудование и методы использованы правильно. Проявлена превосходная теоретическая подготовка. Необходимые навыки и умения полностью освоены. Результат лабораторной работы полностью соответствует её целям.	Оборудование и методы использованы в основном правильно. Проявлена хорошая теоретическая подготовка. Необходимые навыки и умения в основном освоены. Результат лабораторной работы в основном соответствует её целям.	Оборудование и методы частично использованы правильно. Проявлена удовлетворительная теоретическая подготовка. Необходимые навыки и умения частично освоены. Результат лабораторной работы частично соответствует её целям.	Оборудование и методы использованы неправильно. Проявлена неудовлетворительная теоретическая подготовка. Необходимые навыки и умения не освоены. Результат лабораторной работы не соответствует её целям.	1
Устный опрос	В ответе качественно раскрыто содержание темы. Ответ хорошо структурирован. Прекрасно освоен понятийный аппарат. Продемонстрирован высокий уровень понимания материала. Превосходное умение формулировать свои мысли, обсуждать дискуссионные положения.	Основные вопросы темы раскрыты. Структура ответа в целом адекватна теме. Хорошо освоен понятийный аппарат. Продемонстрирован хороший уровень понимания материала. Хорошее умение формулировать свои мысли, обсуждать дискуссионные положения.	Тема частично раскрыта. Ответ слабо структурирован. Понятийный аппарат освоен частично. Понимание отдельных положений из материала по теме. Удовлетворительное умение формулировать свои мысли, обсуждать дискуссионные положения.	Тема не раскрыта. Понятийный аппарат освоен неудовлетворительно. Понимание материала фрагментарное или отсутствует. Неумение формулировать свои мысли, обсуждать дискуссионные положения.	2
Тестирование	86% правильных ответов и более.	От 71% до 85 % правильных ответов.	От 56% до 70% правильных ответов.	55% правильных ответов и менее.	3

Форма контроля	Критерии оценивания				Этап
	Отлично	Хорошо	Удовл.	Неуд.	
Экзамен	Обучающийся обнаружил всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоил основную литературу и знаком с дополнительной литературой, рекомендованной программой дисциплины, усвоил взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявил творческие способности в понимании, изложении и использовании учебно-программного материала.	Обучающийся обнаружил полное знание учебно-программного материала, успешно выполнил предусмотренные программой задания, усвоил основную литературу, рекомендованную программой дисциплины, показал систематический характер знаний по дисциплине и способен к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.	Обучающийся обнаружил знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справился с выполнением заданий, предусмотренных программой, знаком с основной литературой, рекомендованной программой дисциплины, допустил погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладает необходимыми знаниями для их устранения под руководством преподавателя.	Обучающийся обнаружил значительные пробелы в знаниях основного учебно-программного материала, допустил принципиальные ошибки в выполнении предусмотренных программой заданий и не способен продолжить обучение или приступить по окончании университета к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.	

6.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Приложение. Развёрнутое содержание оценочных средств - в прикреплённом файле [F1763575090/B1.V.04. Web_programmirovanie.pdf](#)

Семестр 7

Текущий контроль

1. Лабораторные работы

Темы 1, 2, 3, 4

Тема 1. Переменные, константы, типы данных.

Инкремент и декремент логической переменной, Жесткая ссылка на несуществующий элемент, Ссылки на объекты, Константа с динамическим именем, Использование `var_export()`.

Тема 2. Выражения и операции, Работа с данными формы.

Упаковка значений в битовое представление, Упаковка пяти значений в целое число, Расшифровка закодированного примитива, Логические переменные, Операторы равенства и эквивалентности, Сравнение объектов, Навязчивые предупреждения, Отключение навязчивого предупреждения, Средний параметр в условной конструкции не обязателен, Вывод параметров командной строки, Страница с формой, Использование данных формы, Усовершенствованный скрипт блокировки сервера, Вывод IP-адреса и браузера пользователя, Демонстрация работы с массивом `$_COOKIES`, Все глобальные переменные, Гарантированный прием значений от флажков.

Тема 3. Конструкции языка.

Альтернативный синтаксис `if-else`, Вывод всех степеней двойки до 2^{31} включительно, Демонстрация цикла `for`, Модель сценария для обработки формы, Вывод всех переменных окружения, Использование оператора `goto`, Шапка, Тело скрипта, Подвал, Библиотека для работы с Excel, Библиотека для работы с Word, Ошибка в скрипте.

Тема 4. Ассоциативные массивы.

Демонстрация работы со списками, Слияние списков при помощи оператора `+`, Перебор списка, Перебор ассоциативного массива, Перебор копии массива вместо оригинала, Изменение элементов при переборе.

https://shelly.kpfu.ru/pls/student/docs/F1763575090/B1.V.04._Web_programmirovanie.pdf

2. Устный опрос

Темы 1, 2, 3, 4

Темы 1, 2, 3, 4

1. Переменные (Копирование переменных, Типы переменных, Действия с переменными, Определение типа переменной, Установка типа переменной)
2. Оператор присваивания
3. Ссылочные переменные (Жесткие ссылки, Символические ссылки, Ссылки на объекты)
4. Константы (Предопределенные константы, Определение констант, Проверка существования константы, Константы с динамическими именами)
5. Отладочные функции
6. Выражения (Логические выражения, Строковые выражения)
7. Операции (Арифметические операции, Строковые операции, Операции присваивания, Операции инкремента и декремента, Битовые операции, Операции сравнения, Логические операции, Операция отключения предупреждений, Условные операции)
8. Передача данных командной строки
9. Формы (Трансляция полей формы, Трансляция переменных окружения, Трансляция cookies)
10. Обработка списков
11. Обработка массивов
12. Диагностика
13. Порядок трансляции переменных
14. Инструкция if, else
15. Цикл с предусловием while
16. Цикл с постусловием do, while
17. Универсальный цикл for
18. Инструкции break и continue
19. Цикл foreach
20. Конструкция switch?case
21. Инструкции goto
22. Инструкции require и include
23. Инструкции однократного включения
24. Создание массива. Автомассивы
25. Конструкция list()
26. Списки и ассоциативные массивы
27. Конструкция array() и многомерные массивы
28. Массивы, константы
29. Операции над массивами (Доступ по ключу, Функция count(), Слияние массивов, Косвенный перебор элементов массива, Прямой перебор массива)
30. Списки и строки
31. Сериализация (Упаковка, Распаковка)

3. Тестирование

Темы 1, 2, 3, 4

1. Какие парадигмы программирования поддерживает PHP логическую процедурную объектно-ориентированную

2. Написать программу, которая выводит ?жирными? буквами (тег) строку ?Добро пожаловать!? с использованием языка PHP.

```
<?php echo
```

```
"<b> Добро пожаловать!</b> ";
```

```
?>
```

```
<?php <b>
```

```
echo "Добро пожаловать!" </b>
```

```
?>
```

```
<?>
```

```
echo "<bold>Добро пожаловать! </bold>"
```

```
?>
```

3. Известно, что настройки PHP можно сохранять/изменять не только в `php.ini`, но и в самих PHP скриптах, в файлах `.htaccess` и в файле настроек сервера `httpd.conf`. Где можно установить значение опции `session.auto_start`?

в `php.ini` или `httpd.conf`
в скрипте пользователя
в файле `.htaccess`

4. Какой оператор обозначает равенство значений в языке PHP?

`:=`
`=`
`=:`
`==`

5. Как можно задать массив в языке PHP?

```
$arr[?a?,?b?,?c?] = ?q?;  
$arr (?0"=> ?a?);  
$arr = array(?a?,?b?,?c?);  
$arr[0] = ?a?;
```

6. Какие из утверждений относительно оператора `require` верны?

`require` выполняет код указанного в нем файла только один раз
`require` используется для включения в программу содержимого другого файла
при использовании внутри условных блоков `require` не нужно заключать в фигурные скобки

7. Каким будет результат выполнения следующей программы?

```
<?php  
for ($j = 1;  
{  
  $i = round (9/$j);  
  switch ($i)  
  {  
    case 5: echo ?+?;  
    break;  
  
    case 9: echo ?-?;  
    break;  
  
    case 3: echo ?!?;  
    break 2;  
    default: echo $i;  
    break;  
  }  
  $j++;  
}  
?>
```

`+!`
ошибка синтаксиса
бесконечный цикл
`-+!`

8. Дана команда: `if(!$var) echo ?Hello?;` В каком случае на экран будет выведено слово `?Hello??`

если `$var == true`
если `$var = 0`
если `$var` преобразуется к логическому `false`

9. Какие из операторов `switch` записаны правильно с точки зрения синтаксиса?

```
switch ($par){ "1": echo "1"; "2": echo 2;  
}  
switch ($par){  
  case "1": echo "1"; case 2: echo 2; default: echo 3;
```

```
}  
switch ($par):  
case "1": echo "1"; break;  
case 2: echo 2; break; endswitch;
```

Файл vars.php: <?php

```
$a = 1; ?>
```

Файл index.php: <?php

```
for ($l=0; $l , 3; $l++)
```

```
{  
Include(?vars.php?);  
echo $a++;  
} ?>
```

10. Что мы получим в результате обработки интерпретатором файла index.php?

```
12  
1234  
123  
111
```

11. В каком случае выполняется блок действий цикла

```
for (expr1; expr2; expr3) {  
// блок действий  
}
```

если второе выражение (expr2) вычисляется как true
если третье выражение (expr3) вычисляется как true
если первое выражение (expr1) вычисляется как true

12. Какое из утверждений относительно оператора include верно?

файл, включаемый с помощью include, должен быть правильным HTML-файлом
include используется для включения в программу содержимого другого файла
файл, включаемый с помощью include, может быть любым файлом

13. Как можно узнать виртуальный путь до выполняющегося в данный момент скрипта?

```
getenv(?SCRIPT_NAME?)  
$_SERVER[?SCRIPT_NAME?]  
$_SERVER[?PHP_SELF?]  
с помощью константы SCRIPT_NAME
```

14. Чем отличается клиент от сервера?

клиент посылает запросы, а сервер обрабатывает их
клиент отображает данные на экране компьютера пользователя, а сервер предоставляет данные
сервер выполняет запросы клиента, специализируясь на эффективном решении задач определенного класса
сервер есть часть компьютерной архитектуры сервер-сервер, а клиент ? архитектуры клиент-клиент
сервер создает соединение, а клиент, пользуясь им, передает свой запрос

15. Как можно получить адрес страницы, с которой был послан запрос?

```
$_SERVER[?HTTP_REFERER?]  
getenv(?REMOTE_HOST?)  
getenv(?HTTP_REFERER?)  
с помощью константы HTTP_REFERER
```

16. В чем состоят функции сервера?

обрабатывать запросы клиента
запускать процессы, запрошенные клиентом, и возвращать клиенту результаты
отображать данные на экране компьютера пользователя
инициировать соединение с клиентом

Дана следующая html форма:

```
<form name="my_form" method="get">
```

```
Field 1 <input type=text name="f1"><br>
Select field
<select name="f2">
<option value="1">first
<option value="2">second
</select><br>
<input type=submit>
</form>
```

17. Как можно получить переданные клиентом значения из этой формы?

```
$f1 и $f2, если register_globals=off
$_POST[?f1?] и $_POST[?f2?]
$_GET[?f1?] и $_GET[?f2?]
$POST[?f1?] и $POST[?f2?]
$_REQUEST[?f1?] и $_REQUEST[?f2?]
```

18. Создать класс A, расширяющий класс B с помощью метода add() и переменной \$a.

```
class B{
var $b;
function B(){
}
}
class B extends A {
var $a;
function add($b){
$this->a += $b;
}
}
class B{
var $b;
function B(){
}
}
class A extends B {
var $a;
function add($b){ $this->a += $b;
}
}
class A extends B {
var $a;
function add($b){
$this-> += $b;
}
}
```

19. Как можно получить и вывести на экран список всех методов класса, представителем которого является объект?

```
get_methods(get_class(объект));
get_class_methods(get_class(объект));
get_class_methods(get_class -> объект);
Класс MyClass задан следующим образом:
```

```
<?php
class MyClass{
var $a;
function MyClass() {
$this->a = "hello";
}
function MyFunc($b) {
return $b. " ". $this->a;
}}
?>
```

20. Создать представителя класса MyClass. Получить и вывести значения всех свойств этого класса. Вызвать методы данного класса.

```
$myObj = new MyClass();
echo $myObj->a;
$str = $myObj->MyFunc("people");
Конструктор MyClass() вызывается автоматически при создании представителя класса.
new MyClass();
echo $MyClass->a;
$str = $MyClass->MyFunc("people");
Конструктор MyClass() вызывается автоматически при создании представителя класса.
$myObj = new MyClass("test");
$b = "a";
echo $myObj->$b;
MyFunc($b);
```

21. Имеется объект (\$obj) какого-то класса. Требуется получить список всех свойств класса, которому принадлежит объект.

```
$get_class_vars(get_class->$obj);
get_class_var(get_class($obj));
$vars = get_class_vars(get_class($obj));
```

22. Как можно программно узнать имя класса, представителем которого является объект?

```
$(get_class->объект);
get_class(объект);
class(объект);
```

23. Изменить значения переданных элементов, увеличив их в 10 раз. Использовать функцию array_walk(). Вывести значения элементов до и после применения array_walk

Имеется следующая форма:

```
<form action=task3.php>
Запись номер 1 <input type=checkbox name=id[] value=10> <br>
Запись номер 2 <input type=checkbox name=id[] value=20> <br>
Запись номер 3 <input type=checkbox name=id[] value=30> <br>
<input type=submit value="Отправить">
</form>
```

```
<?
print_r($_GET);
array_walk($_GET,"test");
function test(&$val,$key){
foreach ($val as $k=> $v)
$val[$k] = $v*10;
}
print_r($_GET);
?>
<?
print_r($_GET);
array_walk($_GET,"test");
function test(&$val,$key){
$val = $val*10;
}
print_r($_GET)
?>
<?
print_r($_POST);
array_walk($_POST,"test");
function test(&$val,$key){
foreach ($val as $k=> $v)
$val[$k] = $v*10;
}
```

```
print_r($_POST);
?>
```

24. Дана строка `?<h2>Наука ? двигатель прогресса!</h2>?.` Как можно вывести ее в браузер именно в таком виде, без форматирования согласно стандарту HTML?

```
echo htmlspecialchars("<h2>Наука - <font color=red>двигатель</font> прогресса!</h2>");
echo htmlentities("<h2>Наука ? <font color=red>двигатель</font> прогресса!</h2>");
echo htmlspecialchars("<h2>Наука - <font color=red>двигатель</font> прогресса!</h2>", ENT_COMPAT);
```

25. Дана строка текста: `?PHP ? очень простой язык. Я знаю PHP!!!?.` Заменить в ней все вхождения слова `?PHP?` на слово `?Perl?`.

```
<?php
$str = "PHP ? очень простой язык. Я знаю PHP!!!";
echo str_replace($str, "Perl");
?>
```

```
<?php
$str = "PHP ? очень простой язык. Я знаю PHP!!!";
echo str_replace("PHP", "Perl", $str);
?>
```

```
<?php
$str = "PHP ? очень простой язык. Я знаю PHP!!!";
echo substr_replace("PHP", "Perl", $str);
?>
```

Экзамен

Вопросы к экзамену:

1. Переменные
2. Оператор присваивания
3. Ссылочные переменные
4. Константы
5. Отладочные функции
6. Выражения
7. Операции
8. Передача данных командной строки
9. Формы
10. Обработка списков
11. Обработка массивов
12. Диагностика
13. Порядок трансляции переменных
14. Особенности флажков checkbox
15. Инструкция if?else
16. Цикл с предусловием while
17. Цикл с постусловием do?while
18. Универсальный цикл for
19. Инструкции break и continue
20. Цикл foreach
21. Конструкция switch?case
22. Инструкции goto
23. Инструкции require и include
24. Инструкции однократного включения
25. Создание массива "на лету". Автомассивы
26. Конструкция list()
27. Списки и ассоциативные массивы
28. Конструкция array() и многомерные массивы
29. Массивы?константы
30. Операции над массивами
31. Списки и строки
32. Сериализация

Семестр 8

Текущий контроль

1. Лабораторные работы

Темы 5, 6, 7, 8, 9

Тема 5. Функции и области видимости.

Пример функции и ее использования, Возврат массива, Неявный возврат null, Передача параметров по значению, Передача параметров по ссылке, Переменное число параметров, Использование func_get_args(), Использование ..., Типы аргументов и возвращаемого значения, Строгая типизация, Локальные переменные, Глобальные переменные в функции, Особенности инструкции global, Статические переменные, Функция для вывода содержимого переменной, Пример использования dumper(), Вложенные функции, Эмуляция virtual(), Использование call_user_func_array(), Анонимная функция, Передача анонимной функции в качестве параметра, Замыкания, Возврат ссылки, Сравнение скорости передачи параметров

Тема 6. Объекты и классы.

Пример класса, Вызов метода объекта, Класс с методом, Перегрузка интерполяции, Класс с конструктором, Использование конструктора, Старый способ задания конструкторов, Явное освобождение ресурсов, Деструктор, Использование класса с деструктором, Проблемы алгоритма со счетчиком ссылок, Использование статических членов класса, Локальное кэширование ресурса, Использование констант в классах, Проверка существования функции клонирования, Класс cls, Сериализация объекта \$obj класса cls, Восстановление объекта из строки, Сериализация объекта класса user, Использование методов __sleep() и __wakeup(), Восстановление объекта

Тема 7. Наследование.

Базовый класс, "Ручное" наследование, Проверка класса FileLoggerDebug0, Несовместимость типов, Наследование, Финальные методы, self не позволяет переопределить метод, static позволяет переопределить метод, Использование анонимных классов, Вложенные анонимные классы, Базовый класс страницы, Базовый класс для кэшируемых страниц, Статические страницы, Новости, Проверка виртуальных методов, Абстрактный класс страницы, Абстрактный класс кэшированной страницы, Уточнение и совместимость типов, Оператор instanceof

Тема 8. Интерфейсы и трейты. Пространство имен.

Наследование интерфейсов, Интерфейсы и абстрактные классы, Использование трейтов, Порядок перегрузки методов, Разрешение конфликтов, Объявление пространства имен, Использование пространств имен, Несколько пространств имен в одном файле, Относительные ссылки на элементы, Доступ к глобальному пространству имен, Импортрование, Трейт PHP7\Seo, Трейт PHP7\Tag, Класс PHP7\Page, Использование класса PHP7\Page, Использование функции __autoload(), Загрузка классов, Использование анонимной функции

Тема 9. Обработка ошибок и исключения.

Отключение ошибок, Удобство оператора @, Перехват ошибок и предупреждений, Использование функции trigger_error(), Вывод дерева вызовов функции, Простой пример использования исключений, Инструкция try во вложенных функциях, Деструкторы и исключения, Недостатки set_error_handler(), Наследование исключений, Вывод сведений об исключении, Классификация исключений, Классы-исключения, Использование иерархии исключений, Ошибка использования оператора [], Попытка отловить исключение Error, Использование конструкции finally, Перехват всех исключений, Преобразование ошибок в исключения, Класс для преобразования ошибок PHP в исключения, Иерархия ошибок

2. Устный опрос

Темы 5, 6, 7, 8, 9

Темы 5, 6, 7, 8, 9

1. Класс как тип данных
2. Создание нового класса
3. Работа с классами (Создание объекта некоторого класса, Доступ к свойствам объекта, Доступ к методам, Создание нескольких объектов, Перегрузка преобразования в строку)
4. Инициализация и разрушение (Конструктор, Деструктор, Алгоритм сбора мусора)
5. Права доступа к членам класса (Модификаторы доступа, Неявное объявление свойств, Общие рекомендации)
6. Класс self, объект \$this (счетчик объектов, кэш ресурсов)
7. Константы класса
8. Перехват обращений к членам класса
9. Клонирование объектов (Переопределение операции клонирования, Запрет клонирования)
10. Перехват сериализации
11. Сериализация объектов (Упаковка и распаковка объектов, Методы sleep() и wakeup())
12. Расширение класса (Метод включения, Наследование, Переопределение методов, Константы __CLASS__ и __METHOD__, Позднее статическое связывание)
13. Анонимные классы
14. Полиморфизм (Абстрагирование, Виртуальные методы, Расширение иерархии, Абстрактные классы и методы, Совместимость родственных типов)
15. Интерфейсы (Наследование интерфейсов, Интерфейсы и абстрактные классы)
16. Трейты
17. Объявление пространства имен

18. Иерархия пространства имен

19. Импортирование

20. Автозагрузка классов (Функция __autoload(), Функция spl_autoload_register())

21. Что такое ошибка? (Роли ошибок, Виды ошибок)

22. Контроль ошибок (Директивы контроля ошибок, Установка режима вывода ошибок, Оператор отключения ошибок, Перехват ошибок, Генерация ошибок, Стек вызовов функций)

23. Исключения (Базовый синтаксис, Инструкция throw, Раскрутка стека, Исключения и деструкторы, Исключения и set error handler(), Классификация и наследование, Блоки финализаторы)

24. Трансформация ошибок (Серьезность "несерьезных" ошибок, Преобразование ошибок в исключения)

3. Тестирование

Темы 5, 6, 7, 8, 9

1. Дана строка текста: ?PHP ? очень простой язык. Я знаю PHP!!!?. Заменить в ней все вхождения слова ?PHP? на слово ?Perl?.

```
<?php
$str = "PHP ? очень простой язык. Я знаю PHP!!!";
echo str_replace($str, "PHP", "Perl");
?>

<?php
$str = "PHP ? очень простой язык. Я знаю PHP!!!";
echo str_replace("PHP", "Perl", $str);
?>

<?php
$str = "PHP ? очень простой язык. Я знаю PHP!!!";
echo substr_replace("PHP", "Perl", $str);
?>
```

2. Что делает функция move_uploaded_file (временное имя файла, место назначения)?

удаляет загруженный файл

проверяет, загружен ли файл, и если да, то перемещает его из временной директории в указанную директорию для хранения

перемещает файл из временной директории в указанную директорию для хранения

3. Как узнать размер файла, который клиент хочет загрузить на сервер с помощью этой формы.

Имеется следующая html-форма

```
<form enctype="multipart/form-data" action="parse.php" method="post">
<input type="hidden" name="MAX_FILE_SIZE" value="30000" />
Загрузить файл: <input type="file" name="my" /><br>
<input type="submit" value="Отправить файл" />
</form>
```

```
filesize($_FILES[?my?][?tmp_name?])
```

```
filesize($_FILES[?my?][?name?])
```

```
$_FILES[?size?]
```

```
filesize($_FILES[?name?][?my?])
```

4. С помощью какой функции можно проверить, существует ли файл?

```
file_exists()
```

```
is_writable()
```

```
is_readable()
```

5. Какие из перечисленных функций считывают данные из файла, ссылка на который установлена функцией fopen?

```
fgetc()
```

```
fgetss()
```

```
file_read()
```

```
fread()
```

6. Как узнать, что файл, загружаемый на сервер с помощью этой формы, был загружен успешно?

Имеется следующая html-форма

```
<form enctype="multipart/form-data" action="parse.php" method="post">
```



```
<input type="hidden" name="MAX_FILE_SIZE" value="30000" />
Загрузить файл: <input type="file" name="my" /><br>
<input type="submit" value="Отправить файл" />
</form>
```

```
if (isset($_FILES['my'])) echo "файл загружен успешно?";
if ($_FILES['my']['error'] == 0) echo "файл загружен успешно?";
if ($_FILES['my']['error'] == 1) echo "файл загружен успешно?";
```

7. Каков синтаксис оператора удаления таблицы?

```
SELECT * FROM Articles WHERE author="$author", title="$title", abstract="$abstract" ORDER BY author;
SELECT (title, author, abstract) FROM Articles WHERE ORDER BY author;
SELECT title, author, abstract FROM Articles ORDER BY author;
```

8. Оператор update предназначен для обновления значений существующих столбцов таблицы в соответствии с введенными значениями. Каков синтаксис этого оператора?

```
UPDATE имя_таблицы
GET имя_столбца1=выражение1 [, имя_столбца2=выражение2, ...]
UPDATE имя_таблицы SET имя_столбца1=выражение1 [, имя_столбца2=выражение2, ...]
UPDATE [LOW_PRIORITY] [IGNORE] имя_таблицы
SET имя_столбца1=выражение1 [, имя_столбца2=выражение2, ...]
[WHERE where_definition] [LIMIT число]
```

9. Составить запрос к таблице описаний статей (Articles): изменить название статьи (title), автор (author) которой ?Петров? на название ?Второе название?.

```
UPDATE Articles SET title="Второе название" WHERE author="Петров";
UPDATE Articles SET title="Второе название";
UPDATE Articles SET title="Второе название" WHERE title="Первое название";
```

10. Составить запрос к таблице Articles на добавление описания статьи с названием (title) ?Новая статья? и автором (author) ?Сидоров С. С.?.

```
INSERT INTO Articles SET title='Новая статья', author='Сидоров С. С.';
INSERT INTO Articles title='Новая статья', author='Сидоров С. С.';
INSERT INTO Articles (title, author) VALUES('Новая статья', 'Сидоров С. С.');
```

11. Какие значения будут установлены для параметров server, username, password, если они не были заданы при вызове функции mysql_connect?

Функция mysql_connect, устанавливающая соединение с базой данных MySQL, имеет следующий синтаксис:

```
mysql_connect ( server, username, password, new_link, client_flags)
```

```
server = 'localhost:3306'
```

```
username = имя пользователя владельца процесса сервера
```

```
password = пустой пароль
```

```
эти параметры обязательны для функции mysql_connect
```

```
server = 'localhost:8080'
```

```
username = имя пользователя владельца процесса сервера
```

```
password = пароль пользователя владельца процесса сервера
```

12. Каковы синтаксис и семантика функции explode?

массив explode (строка string). Эта функция разбивает строку string на части с помощью разделителя ? ? и возвращает массив полученных строк

эта функция без параметров. Ее семантика неизвестна

массив explode(строка separator, строка string [, int limit]). Эта функция разбивает строку string на части с помощью разделителя separator и возвращает массив полученных строк

13. Параметр session.cookie_lifetime задает длительность жизни cookies в секундах. Какое значение имеет этот параметр по умолчанию?

по умолчанию это ?0?, т.е. данные в cookies считаются правильными до закрытия окна браузера

по умолчанию это ?10?, т.е. данные в cookies считаются правильными ровно 10 секунд

по умолчанию это ?60?

14. Уничтожить текущую сессию целиком можно командой session_destroy(); К чему приведет уничтожение сессии?

после уничтожения сессии очищается массив \$_SESSION и больше ничего не происходит

после уничтожения сессии уничтожается ее идентификатор, мы больше не можем ни зарегистрировать переменные, ни вообще производить какие-либо действия с сессией
после уничтожения сессии массив `$_SESSION` заполняется нулями

15. С помощью какой функции можно получить идентификатор сессии?

`id_session`
`session_identificator`
`session_id`

16. Какова структура регулярного выражения?

общая структура регулярного выражения: шаблон, выделенный с помощью специального символа разделителя, модификатор, влияющий на способ обработки регулярного выражения
общая структура регулярного выражения: шаблон, заключенный в круглые скобки, и функция для его обработки
общая структура регулярного выражения: шаблон, выделенный с помощью специального символа разделителя

17. Назначение метасимвола `?-??`

вычисляет символьный класс
отрицание класса, но только если это первый символ
задает диапазон символов

18. Что возвращает метод `document_element` класса `DomDocument`?

содержимое элемента
корневой элемент
значение атрибута

19. Какая комбинация технологий позволяет получить HTML-документы?

XML + XSLT
XML + DTD
XSLT + DTD

20. Что представляет собой шаблон `Smarty`?

это набор переменных `Smarty` и `html`-тегов
это набор `html`-тегов
это набор переменных, циклов, условных операторов, операторов вставки файлов и т.д

21. Для чего используется метод `parse()` при работе с шаблонами `FastTemplate`?

обрабатывает шаблон и выводит его содержимое на экран
этот метод инициализирует обработку шаблона и сохраняет обработанный шаблон в переменную
этот метод присваивает переменной указанное значение

22. Как задается переменная в шаблоне `Smarty`?

`{имя_переменной}`
`{${имя_переменной}}`
`{#bodyBgColor#}`

23. Функция `domxml_open_file` в качестве результата возвращает объект класса `DOMDocument`. Что передается этой функции в качестве параметра?

в качестве параметра передается строка, содержащая XML-документ
в качестве параметра передается строка, содержащая путь к XML-документу
в качестве параметра передается объект класса `DOMNode`

24. С какими параметрами может вызываться функция `explode()`?

максимальное количество возвращаемых строк
максимальная длина возвращаемых строк
разделитель в виде строки
строка для разделения

25. Что такое пользовательская директория сервера и где она находится по умолчанию (для windows систем)?

это директория, выделенная пользователю для личных нужд, файлы которой обрабатываются сервером.
Находится в `?C:/Program Files/Apache Group/Apache/htdocs/users/?`

это директория, выделенная пользователю для личных нужд, файлы которой обрабатываются сервером, если он получает запрос типа ~user. Находится в ?C:/Program Files/Apache Group/Apache/users/?

это директория, выделенная пользователю для личных нужд, файлы которой никогда не обрабатываются сервером. Находится в ?C:/Program Files/Apache Group/Apache/users/?

это директория, выделенная пользователю для личных нужд, файлы которой обрабатываются сервером, если он получает запрос типа ~user. Находится в ?C:/Program Files/Apache Group/Apache/htdocs/users/?

Экзамен

Вопросы к экзамену:

1. Общий синтаксис определения функции
2. Локальные переменные
3. Глобальные переменные
4. Статические переменные
5. Рекурсия
6. Вложенные функции
7. Условно определяемые функции
8. Передача функций по ссылке
9. Возврат функцией ссылки
10. Технология отложенного копирования
11. Класс как тип данных
12. Создание нового класса
13. Работа с классами
14. Инициализация и разрушение
15. Права доступа к членам класса
16. Класс ? self, объект ? \$this
17. Константы класса
18. Перехват обращений к членам класса
19. Клонирование объектов
20. Перехват сериализации
21. Сериализация объектов
22. Расширение класса
23. Анонимные классы
24. Полиморфизм
25. Интерфейсы
26. Трейты
27. Объявление пространства имен
28. Иерархия пространства имен
29. Импортирование
30. Автозагрузка классов
31. Что такое ошибка?
32. Контроль ошибок
33. Исключения
34. Трансформация ошибок

6.4 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

В КФУ действует балльно-рейтинговая система оценки знаний обучающихся. Суммарно по дисциплине (модулю) можно получить максимум 100 баллов за семестр, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов.

Для зачёта:

56 баллов и более - "зачтено".

55 баллов и менее - "не зачтено".

Для экзамена:

86 баллов и более - "отлично".

71-85 баллов - "хорошо".

56-70 баллов - "удовлетворительно".

55 баллов и менее - "неудовлетворительно".

Форма контроля	Процедура оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций	Этап	Количество баллов
Семестр 7			

Форма контроля	Процедура оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций	Этап	Количество баллов
Текущий контроль			
Лабораторные работы	В аудитории, оснащённой соответствующим оборудованием, обучающиеся проводят учебные эксперименты и тренируются в применении практико-ориентированных технологий. Оцениваются знание материала и умение применять его на практике, умения и навыки по работе с оборудованием в соответствующей предметной области.	1	20
Устный опрос	Устный опрос проводится на практических занятиях. Обучающиеся выступают с докладами, сообщениями, дополнениями, участвуют в дискуссии, отвечают на вопросы преподавателя. Оценивается уровень домашней подготовки по теме, способность системно и логично излагать материал, анализировать, формулировать собственную позицию, отвечать на дополнительные вопросы.	2	20
Тестирование	Тестирование проходит в письменной форме или с использованием компьютерных средств. Обучающийся получает определённое количество тестовых заданий. На выполнение выделяется фиксированное время в зависимости от количества заданий. Оценка выставляется в зависимости от процента правильно выполненных заданий.	3	10
Экзамен	Экзамен нацелен на комплексную проверку освоения дисциплины. Экзамен проводится в устной или письменной форме по билетам, в которых содержатся вопросы (задания) по всем темам курса. Обучающемуся даётся время на подготовку. Оценивается владение материалом, его системное освоение, способность применять нужные знания, навыки и умения при анализе проблемных ситуаций и решении практических заданий.		50
Семестр 8			
Текущий контроль			
Лабораторные работы	В аудитории, оснащённой соответствующим оборудованием, обучающиеся проводят учебные эксперименты и тренируются в применении практико-ориентированных технологий. Оцениваются знание материала и умение применять его на практике, умения и навыки по работе с оборудованием в соответствующей предметной области.	1	20
Устный опрос	Устный опрос проводится на практических занятиях. Обучающиеся выступают с докладами, сообщениями, дополнениями, участвуют в дискуссии, отвечают на вопросы преподавателя. Оценивается уровень домашней подготовки по теме, способность системно и логично излагать материал, анализировать, формулировать собственную позицию, отвечать на дополнительные вопросы.	2	20
Тестирование	Тестирование проходит в письменной форме или с использованием компьютерных средств. Обучающийся получает определённое количество тестовых заданий. На выполнение выделяется фиксированное время в зависимости от количества заданий. Оценка выставляется в зависимости от процента правильно выполненных заданий.	3	10
Экзамен	Экзамен нацелен на комплексную проверку освоения дисциплины. Экзамен проводится в устной или письменной форме по билетам, в которых содержатся вопросы (задания) по всем темам курса. Обучающемуся даётся время на подготовку. Оценивается владение материалом, его системное освоение, способность применять нужные знания, навыки и умения при анализе проблемных ситуаций и решении практических заданий.		50

7. Перечень литературы, необходимой для освоения дисциплины (модуля)

Освоение дисциплины (модуля) предполагает изучение основной и дополнительной учебной литературы. Литература может быть доступна обучающимся в одном из двух вариантов (либо в обоих из них):

- в электронном виде - через электронные библиотечные системы на основании заключенных КФУ договоров с правообладателями и предоставленных доступов НЧИ КФУ;

- в печатном виде - в фонде библиотеки Набережночелнинского института (филиала) КФУ. Обучающиеся получают учебную литературу на абонементе по читательским билетам в соответствии с правилами пользования библиотекой.

Электронные издания доступны дистанционно из любой точки при введении обучающимся своего логина и пароля от личного кабинета в системе "Электронный университет". При использовании печатных изданий библиотечный фонд должен быть укомплектован ими из расчета не менее 0,5 экземпляра (для обучающихся по ФГОС 3++ - не менее 0,25 экземпляра) каждого из изданий основной литературы и не менее 0,25 экземпляра дополнительной литературы на каждого обучающегося из числа лиц, одновременно осваивающих данную дисциплину.

Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля), находится в Приложении 2 к рабочей программе дисциплины. Он подлежит обновлению при изменении условий договоров КФУ с правообладателями электронных изданий и при изменении комплектования фондов библиотеки Набережночелнинского института (филиала) КФУ.

8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

HTML CSS - <http://htmlbook.ru/>

Javascript - <https://learn.javascript.ru/>

PHP - <https://php.ru/>

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Вид работ	Методические рекомендации
лекции	Лекция - устное изложение информации, выстроенное по строго определенной логической структуре, подчиненной задаче максимально глубоко и понятно раскрыть заданную тематику. Основное назначение лекции: помощь в освоении фундаментальных аспектов; упрощение процесса понимания научно-популярных проблем; распространение сведений о новых достижениях современной науки. Функции лекционной подачи материала: информационная; стимулирующая; воспитательная; развивающая; ориентирующая; поясняющая; убеждающая. Лекция позволяет раскрыть основные понятия и проблематику изучаемой области науки, дать учащимся представление о сути предмета, продемонстрировать взаимосвязь с другими смежными дисциплинами. Использование дистанционных технологий в образовательном процессе.
лабораторные работы	Лабораторная работа требует от студентов не только наличия знаний, но еще и умений применять эти знания в новых ситуациях, сообразительности. Лабораторная работа активизирует познавательную деятельность студентов, выполняется на компьютере. Используется лабораторная работа для закрепления определенных навыков с программными средствами, когда кроме алгоритмических предписаний в задании студентов может получать консультации преподавателя. Подготовка к лабораторной работе включает 2 этапа: организационный; закрепление и углубление теоретических знаний. В процессе этой работы студент должен стремиться понять и запомнить основные положения материала, примеры, поясняющие его, а также разобраться в дополнительном материале. Использование дистанционных технологий в образовательном процессе.
самостоятельная работа	В процессе самостоятельной деятельности студент должен научиться выделять познавательные задачи, выбирать способы их решения, выполнять операции контроля за правильностью решения поставленной задачи, совершенствовать навыки реализации теоретических знаний. Формирование умений и навыков самостоятельной работы студентов может протекать как на сознательной, так и на интуитивной основе. Самостоятельная работа - деятельность обучающегося направленная на самостоятельное выполнение заданий. К самостоятельной работе относятся: подготовка к практическим, лабораторным и семинарским занятиям, подготовка к устному опросу, подготовка к тестированию. Самостоятельная работа творческого характера позволяет развивать творческие способности студентов. Использование дистанционных технологий в образовательном процессе.
устный опрос	Устный опрос как контроль знаний студентов осуществляется в виде фронтальной и индивидуальной проверки. При фронтальном опросе за короткое время проверяется состояние знаний студентов всей группы по определенному вопросу или группе вопросов. Эта форма проверки используется для: выяснения готовности группы к изучению нового материала; определения сформированности понятий; проверки домашних заданий; поэтапной или окончательной проверки учебного материала, только что разобранный на занятии; при подготовке к выполнению практических и лабораторных работ. Индивидуальный устный опрос позволяет выявить правильность ответа по содержанию, его последовательность, самостоятельность суждений и выводов, степень развития логического мышления, культуру речи студентов. Использование дистанционных технологий в образовательном процессе.

Вид работ	Методические рекомендации
тестирование	Тест представляет собой кратковременное технически сравнительно просто составленное испытание, проводимое в равных для всех условиях. Различают следующие виды тестов. Избирательный тест состоит из системы заданий, к каждому из которых прилагаются как верные, так и неверные ответы. Из них студент выбирает тот, который считает верным для данного вопроса. При этом неверные ответы содержат такую ошибку, которую студент может допустить, имея определенные пробелы в знаниях. Закрытые тесты не содержат вариантов ответов. Студенты предлагают свой вариант ответа. Имеются тесты перекрестного выбора, в которых требуется установить соответствие между элементами множества ответов. Встречаются также тесты идентификации, в которых в качестве ответов приводятся графики, схемы, чертежи. Использование дистанционных технологий в образовательном процессе.
экзамен	Экзаменом заканчивается изучение предмета или его основного раздела. Наиболее распространённые формы экзамена: экзамен по билетам, экзамен-беседа, письменный экзамен, коллоквиум, тестовый контроль, программированный контроль. Успешно сдает экзамен тот, кто умеет периодически повторять материал, который был прослушан на лекциях, законспектирован и закреплен на самостоятельных занятиях. Повторение предполагает обобщение, углубление, а также расширение усвоенных знаний. При подготовке к экзамену необходимо опираться на лекции, а также на источники, которые разбирались на практических, семинарских и лабораторных занятиях. Использование дистанционных технологий в образовательном процессе.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем, представлен в Приложении 3 к рабочей программе дисциплины (модуля).

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Материально-техническое обеспечение образовательного процесса по дисциплине (модулю) включает в себя следующие компоненты:

Помещения для самостоятельной работы обучающихся, укомплектованные специализированной мебелью (столы и стулья) и оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду КФУ.

Учебные аудитории для контактной работы с преподавателем, укомплектованные специализированной мебелью (столы и стулья).

Компьютер и принтер для распечатки раздаточных материалов.

Мультимедийная аудитория.

Компьютерный класс.

12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;
- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;
- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;
- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;

- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;
- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;
- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи:
- продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;
- продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;
- продолжительности выступления обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 09.03.03 "Прикладная информатика" и профилю подготовки "Прикладная информатика в экономике".

Перечень литературы, необходимой для освоения дисциплины (модуля)

Направление подготовки: 09.03.03 - Прикладная информатика

Профиль подготовки: Прикладная информатика в экономике

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2020

Основная литература:

1. Шакин В. Н. Объектно-ориентированное программирование на Visual Basic в среде Visual Studio.NET : учебное пособие / В.Н. Шакин, А.В. Загвоздкина, Г.К. Сосновиков. - Москва : ФОРУМ : ИНФРА-М, 2019. - 398 с. - (Высшее образование. Бакалавриат). - ISBN 978-5-00091-048-1. - URL: <https://znanium.com/catalog/product/1010028> (дата обращения: 05.10.2020). - Текст : электронный.
2. Лисьев Г.А. Программное обеспечение компьютерных сетей и web-серверов : учебное пособие / Г. А. Лисьев, П. Ю. Романов, Ю. И. Аскерко. - Москва : ИНФРА-М, 2020. - 145 с. - (Высшее образование: Бакалавриат). - ISBN 978-5-16-013565-6. - URL: <https://znanium.com/catalog/product/1068576> (дата обращения: 05.10.2020). - Текст : электронный.
3. Мартишин С. А. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench : учебное пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. - Москва : ФОРУМ : ИНФРА-М, 2019. - 160 с. - (Высшее образование). - ISBN 978-5-8199-0517-3. - URL: <https://znanium.com/catalog/product/1007949> (дата обращения: 05.10.2020). - Текст : электронный.

Дополнительная литература:

1. Шакин В. Н. Базовые средства программирования на Visual Basic в среде Visual Studio .NET. Практикум : учебное пособие / В. Н. Шакин. - Москва : ФОРУМ : ИНФРА-М, 2019. - 288 с. - (Высшее образование. Бакалавриат). - ISBN 978-5-00091-054-2. - URL: <https://znanium.com/catalog/product/983576> (дата обращения: 18.08.2021). - Текст : электронный.
2. Алибеков Б. И. Лабораторный практикум по Web-программированию на PHP : учебное пособие / Б. И. Алибеков. - Махачкала : ДГУ, 2018. - 273 с. - URL: <https://e.lanbook.com/book/158357> (дата обращения: 18.08.2021). - Текст : электронный.
3. Побединский Е. В. Проектирование веб-сайтов с использованием технологий PHP, HTML, CSS и WordPress : учебное пособие / Е. В. Побединский, В. В. Побединский. - Екатеринбург : УГЛТУ, 2018. - 115 с. - ISBN 978-5-94984-651-3. - URL: <https://e.lanbook.com/book/142518> (дата обращения: 18.08.2021). - Текст : электронный.

Приложение 3
к рабочей программе дисциплины (модуля)
Б1.В.04 Web-программирование

Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Направление подготовки: 09.03.03 - Прикладная информатика

Профиль подготовки: Прикладная информатика в экономике

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2020

Освоение дисциплины (модуля) предполагает использование следующего программного обеспечения и информационно-справочных систем:

Операционная система Microsoft Windows 7 Профессиональная или Windows XP (Volume License)

Пакет офисного программного обеспечения Microsoft Office 365 или Microsoft Office Professional plus 2010

Браузер Mozilla Firefox

Браузер Google Chrome

Adobe Reader XI или Adobe Acrobat Reader DC

Kaspersky Endpoint Security для Windows