МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
ДЕПАРТАМЕНТ НАУЧНО-ТЕХНОЛОГИЧЕСКОЙ ПОЛИТИКИ И ОБРАЗОВАНИЯ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО Донской ГАУ)

АЗОВО-ЧЕРНОМОРСКИЙ ИНЖЕНЕРНЫЙ ИНСТИТУТ – ФИЛИАЛ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ» В Г. ЗЕРНОГРАДЕ (Азово-Черноморский инженерный институт ФГБОУ ВО Донской ГАУ)

Бутенко Александр Федорович
Подписано цифровой подписью: Бутенко Александр Федорович
Дата: 2024.04.18 09:38:21 +03'00'

УТВЕРЖДАЮ
Директор

А.Ф. Бутенко
18 апреля 2024 г.

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине

ПД.06 Информатика
программы подготовки специалистов среднего звена по специальности

23.02.07 «Техническое обслуживание и ремонт двигателей, систем и агрегатов автомобилей»

Зерноград, 2024

Разработчик: преподаватель 1 категории Емелин А.А.

Фонд контрольно-оценочных средств рассмотрен и одобрен на заседании кафедры математики и биоинформатики «25» марта 2024 г. Протокол № 8.

Заведующая кафедрой: Грачева Н.Н.

Фонд оценочных средств рассмотрен и одобрен на заседании методической комиссии по общеобразовательной подготовке от «28» марта 2024 г. Протокол № 7.

Председатель: Жолобова М.В.

1. Общие положения

Контрольно-оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, осваивающих программу дисциплины ПД.02 «Информатика».

Фонд оценочных средств включает контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации в форме зачета с оценкой.

2. Результаты освоения дисциплины, подлежащие проверке

Результаты обучения (умения, знания, общие компетенции)	**Основные показатели оценки результатов**	**Форма контроля и оценивания**
В результате освоения учебной дисциплины обучающийся должен ***уметь***:		
уметь читать и понимать программы, реализующие несложные алгоритмы обработки числовых и текстовых данных (в том числе массивов и символьных строк) на выбранном для изучения универсальном языке программирования высокого уровня (Паскаль, Python, Java, C++, C#); анализировать алгоритмы с использованием таблиц трассировки; определять без использования компьютера результаты выполнения несложных программ, включающих циклы, ветвления и подпрограммы, при заданных исходных данных; модифицировать готовые программы для решения новых задач, использовать их в своих программах в качестве подпрограмм (процедур, функций);	описать алгоритмические структуры с применением языка профессора Ершова; использовать вербальные алгоритмические структуры для решения задач;	контрольная работа контрольная работа контрольная работатать текущий контроль за выполнением самостоятельной работы контрольная работа
уметь реализовать этапы решения задач на компьютере; умение реализовывать на выбранном для изучения языке программирования высокого уровня (Паскаль, Python, Java, C++, C#) типовые алгоритмы обработки чисел, числовых последовательностей и массивов: представление числа в виде набора простых	реализовывать простейшие алгоритмические структуры на языке высокого уровня Python, Java	контрольная работа контрольная работа

сомножителей; нахождение максимальной (минимальной) цифры натурального числа, записанного в системе счисления с основанием, не превышающим 10; вычисление обобщенных характеристик элементов массива или числовой последовательности (суммы, произведения среднего арифметического, минимального и максимального элементов, количества элементов, удовлетворяющих заданному условию); сортировку элементов массива;		текущий контроль за выполнением самостоятельной работы текущий контроль за выполнением правил техники безопасности при выполнении лабораторных работ текущий контроль за выполнением самостоятельной работы
уметь создавать структурированные текстовые документы и демонстрационные материалы с использованием возможностей современных программных средств и облачных сервисов; умение использовать табличные (реляционные) базы данных, в частности, составлять запросы в базах данных (в том числе вычисляемые запросы), выполнять сортировку и поиск записей в базе данных; наполнять разработанную базу данных; умение использовать электронные таблицы для анализа, представления и обработки данных (включая вычисление суммы, среднего арифметического, наибольшего и наименьшего значений, решение уравнений);	набирать и форматировать текст в текстовом редакторе; использовать для обработки графической информации редакторы растровой и векторной графики; использовать табличный табулятор для организации вычислений и построения диаграмм; создавать базы данных, организовывать в них запросы на поиск информации, создавать формы для её ввода; вычислять объем текстовой и графической информации; использовать аппарат язык а высокого уровня Python и электронные таблицы для построения моделей	
уметь использовать компьютерно-математические модели для анализа объектов и процессов: формулировать цель моделирования, выполнять анализ результатов, полученных в ходе моделирования; оценивать адекватность модели моделируемому объекту или процессу; представлять результаты моделирования в наглядном виде		
уметь строить неравномерные	использовать для решения	

коды, допускающие однозначное декодирование сообщений (префиксные коды); использовать простейшие коды, которые позволяют обнаруживать и исправлять ошибки при передаче данных;	различных задач аппарат алгебры логики (создание сложных логических высказываний, построение таблиц истинности, преобразование формул алгебры логики).	
В результате освоения учебной дисциплины обучающийся должен ***знать***:		
понимать угрозу информационной безопасности, использовать методы и средства противодействия этим угрозам, соблюдать меры безопасности, предотвращающие незаконное распространение персональных данных; соблюдать требования техники безопасности и гигиены при работе с компьютерами и другими компонентами цифрового окружения; понимать правовые основы использования компьютерных программ, баз данных и работы в сети Интернет; уметь организовывать личное информационное пространство с использованием различных средств цифровых технологий; понимание возможностей цифровых сервисов государственных услуг, цифровых образовательных сервисов; понимать возможности и ограничения технологий искусственного интеллекта в различных областях; иметь представление об использовании информационных технологий в различных профессиональных сферах; владеть представлениями о роли информации и связанных с ней процессов в природе, технике и обществе; понятиями «информация», «информационный процесс», «система», «компоненты системы» «системный эффект», «информационная система», «система управления»; владение методами поиска информации в сети Интернет; уметь критически оценивать	правила техники безопасности при работе с ЭВМ; определение и составляющие информационной безопасности, аспекты информационной безопасности; виды технологий компьютерных преступлений; определение и виды компьютерных вирусов, способы защиты от них; назначение и возможности электронных таблиц; назначение и основные возможности баз данных; основные объекты баз данных и допустимые операции над ними; виды прикладного и служебного ПО и их назначение, возможности текстового редактора по оформлению документа; этапы информационной технологии решения задач с использованием компьютера; алфавит и синтаксис основных команд языка профессора Ершова;	текущий контроль: проверка конспекта лекций, устный опрос в начале лабораторных работ; промежуточная аттестация в виде тестирования.

информацию, полученную из сети Интернет; характеризовать большие данные, приводить примеры источников их получения и направления использования; - понимать основные принципы устройства и функционирования современных стационарных и мобильных компьютеров; тенденций развития компьютерных технологий; владеть навыками работы с операционными системами и основными видами программного обеспечения для решения учебных задач по выбранной специализации; - иметь представления о компьютерных сетях и их роли в современном мире; об общих принципах разработки и функционирования интернет-приложений; понимать основные принципы дискретизации различных видов информации; умение определять информационный объем текстовых, графических и звуковых данных при заданных параметрах дискретизации; владеть теоретическим аппаратом, позволяющим осуществлять представление заданного натурального числа в различных системах счисления; выполнять преобразования логических выражений, используя законы алгебры логики; определять кратчайший путь во взвешенном графе и количество путей между вершинами ориентированного ациклического графа	архитектуру компьютера; назначение и основные характеристики устройств компьютера; устройство и назначение сетевых периферийных устройств; как создавать сетевой кабель на базе витой пары; преобразовывать числа из одной системы счисления в другую, выполнять арифметические операции в произвольной системе счисления; функции языка как способа представления информации; основные единицы измерения количества информации; правила выполнения арифметических операций в двоичной, восьмеричной и шестнадцатеричной системах счисления; основные логические операции, их свойства и обозначения, основные алгоритмические структуры и описание их;	

3. Контрольно-оценочные материалы текущего контроля

3.1. Оценивание решения задач

Правильное решение не менее 85% задач – 5 баллов;
Правильное решение от 70% до 85% задач – 4 балла;
Правильное решение от 55% до 60% задач – 3 балла;

Менее 55% решения задач – 2 балла.

Условия выполнения задания

1) Место (время) выполнения задания*:* задание выполняется на занятие в аудиторное время

2) Время выполнения задания: 45-90 мин.

Шкала оценки образовательных достижений:

Лабораторные работы

Критерии оценки:

Выполнение не менее 55% всей работы – положительная оценка

3.2. Примеры контрольных заданий

Задания входного контроля (один из возможных вариантов)

1. В одной из кодировок Unicode каждый символ кодируется 16 битами. Вова написал текст (в нём нет лишних пробелов): «Бор, азот, гелий, натрий, водород, кислород, рентгений, менделевий, резерфордий – химические элементы». Ученик вычеркнул из списка название одного химического элемента. Заодно он вычеркнул ставшие лишними запятую и пробел – два пробела не должны идти подряд. При этом размер нового предложения в данной кодировке оказался на 18 байт меньше, чем размер исходного предложения. Напишите в ответе вычеркнутое название элемента.

Ответ _____

2. Вася и Петя играли в шпионов и кодировали сообщение собственным шифром. Фрагмент кодовой таблицы приведён ниже.

А	Б	В	Г	Д	Е
~	#	#+	+~#	+#	~#

Расшифруйте сообщение. Получившееся слово (набор букв) запишите в качестве ответа: # ~ # + + ~ #

Ответ _____

3. Напишите наибольшее натуральное число x, для которого ИСТИННО высказывание: НЕ (x < 3) И (x < 4)

Ответ _____

4. Между населёнными пунктами A, B, C, D, E построены дороги, протяжённость которых (в километрах) приведена в таблице:

	A	B	C	D	E
A		3			
B	3		1	2	6
C		1			3
D		2			3
E		6	3	3	

Определите длину кратчайшего пути между пунктами A и E. Передвигаться можно только по дорогам, протяжённость которых указана в таблице. Каждый пункт можно посетить только один раз.

Ответ _____

5. У исполнителя Вычислитель две команды, которым присвоены номера:

1. прибавь 3

2. умножь на 2

Первая из них увеличивает число на экране на 3, вторая удваивает его. Составьте алгоритм получения из числа 1 числа 25, содержащий не более 5 команд.

В ответе запишите только номера команд.

(Например, 11221 – это алгоритм:

прибавь 3

прибавь 3

умножь на 2

умножь на 2

прибавь 3

который преобразует число 4 в 43.)

Если таких алгоритмов более одного, то запишите любой из них.

Ответ _____

Фонд оценочных средств для рубежного контроля

(Тема 1.1, Тема 1.3, Тема 1.6, Тема 1.9, Тема 3.1, Тема 3.2, Тема 3.5)

Образовательные результаты, подлежащие проверке (элементы):

Б1, Б2, Б3, Б4, Б7, Б9, Б11

ОК 01, ОК 02

Тема.1.1. Информация и информационные процессы

Ближе всего раскрывается смысл понятия «информация, используемая в бытовом общении» в утверждении:

последовательность знаков некоторого алфавита;

сообщение, передаваемое в форме знаков ли сигналов;

сообщение, уменьшающее неопределенность знаний;

сведения об окружающем мире, воспринимаемые человеком

сведения, содержащиеся в научных теориях

Информацию, не зависящую от личного мнения, называют:

достоверной;

актуальной;

объективной;

полезной;
понятной
Информацию, отражающую истинное положение дел, называют:
понятной;
достоверной;
объективной;
полной;
полезной
Информацию, существенную и важную в настоящий момент, называют:
полезной;
актуальной;
достоверной;
объективной;
полной
Информацию, дающую возможность, решать поставленную задачу, называют:
понятной;
актуальной;
достоверной;
полезной;
полной
Информацию, достаточную для решения поставленной задачи, называют:
полезной;
актуальной;
полной;
достоверной;
понятной
Информацию, изложенную на доступном для получателя языке, называют:
полной;
полезной;
актуальной;
достоверной;
понятной
По способу восприятия информации человеком различают следующие виды информации:
текстовую, числовую, символьную, графическую, табличную и пр.;
научную, социальную, политическую, экономическую, религиозную пр.;

обыденную, производственную, техническую, управленческую;
визуальную, звуковую, тактильную, обонятельную, вкусовую;
математическую, биологическую, медицинскую, психологическую и пр.
По форме представления информации можно условно разделить на следующие виды:
социальную, политическую, экономическую, техническую, религиозную и пр.;
техническую, числовую, символьную, графическую, табличную пр.;
обыденную, научную, производственную, управленческую;
визуальную звуковую, тактильную, обонятельную, вкусовую;
математическую, биологическую, медицинскую, психологическую.
Укажите лишний объект с точки зрения способа представления информации:
школьный учебник;
фотография;
телефонный разговор;
картина;
чертеж
По области применения информацию можно условно разделить на:
текстовую и числовую;
визуальную и звуковую;
графическую и табличную;
научную и техническую;
тактильную и вкусовую
Какое из высказываний ложно?
получение и обработка информации являются необходимыми условиями жизнедеятельности любого организма.
для обмена информацией между людьми служат языки.
информацию условно можно разделить на виды в зависимости от формы представления.
процесс обработки информации техническими устройствами носит осмысленный характер.
процессы управления – это яркий пример информационных процессов, протекающих в природе, обществе, технике.
Каждая знаковая система строится на основе:
естественных языков, широко используемых человеком для представления информации;

двоичной знаковой системы, используемой в процессах хранения, обработки и передачи информации в компьютере;
определенного алфавита (набора знаков) и правил выполнения операций над знаками;
правил синтаксиса алфавита.
Выбери из списка все языки, которые можно считать формальными языками:
двоичная система счисления
языки программирования
кириллица
китайский язык
музыкальные ноты
русский язык
дорожные знаки
код азбуки Морзе.
Производится бросание симметричной восьмигранной пирамидки. Какое количество информации мы получаем в зрительном сообщении о ее падении на одну из граней?
1 бит
1 байт
3 бит
3 бита.
Какое количество информации (с точки зрения алфавитного подхода) содержит двоичное число 10101001?
1 байта
2 байта
3 байта
3 бита.
Что из нижеперечисленного не является основой формирования информационной культуры?
знания о законах функционирования информационной среды
принцип узкой специализации
знания об информационной среде
умение ориентироваться в информационных потоках
Установите соответствие:

А Полнота	1 Язык понятен получателю
Б Достоверность	2 Достаточность для понимания, принятия решения

В Актуальность	3 Важность, значимость
Г Понятность	4 Неискажение истинного положения дел
Д Релевантность	5 Вовремя, в нужный срок

Ключ к тесту

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Ответ	г	в	б	б	г	в	д	г	б	в	г	г	в	абджз	в	а	б	а2 б4 в3 г1 д3

Примеры контрольных заданий по разделу 1 «Информация и информационная деятельность человека»

1) Расположите в порядке возрастания следующие объемы информации 20 бит, 2 байта, 1020 байт, 1 Кбайт.

2) Сколько символов содержит сообщение, записанное с помощью 16-ти символьного алфавита, если объем его составил 1/16 Мбайт?

3) Каков информационный объем сообщения «Саша один из 8-ми кандидатов в сборную».

4) Выполнить перевод числа $27_{10} \rightarrow N_2$.

5) Выполнить перевод числа $2E1_{16} \rightarrow N_{10}$.

6) Вычислить 1001_2+1101_2.

7) Расположите в порядке убывания следующие объемы информации 10 бит, 1 Кбайт, 2 байта, 1020 байт

8) Сколько килобайт составит сообщение из 384 символов 16-символьного алфавита?

9) В племени 21 мужчина, 18 женщин и 45 детей. Сколько бит информации содержится в сообщении «шаман племени – мужчина».

10) Расположите в порядке возрастания следующие объемы информации 20 бит, 3 байта, 1 Мбайт, 1000 Кбайт

11) Сколько символов содержит сообщение, записанное с помощью 32-х символьного алфавита, если объем его составил 1/64 Мбайт?

12) В группе 21 девочка и 3 мальчика. Каков информационный объем сообщения: «Староста группы – мальчик».

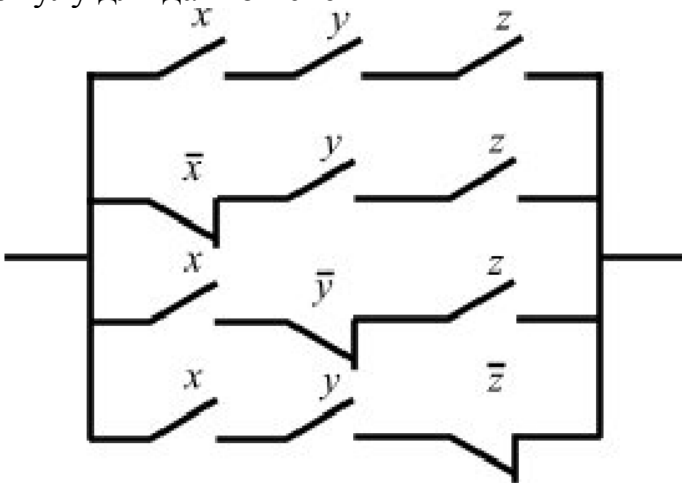
13) Расположите в порядке убывания следующие объемы информации 2 байта, 20 бит, 1 Гбайт, 200 Мбайт

14) Сколько килобайт составит сообщение из 150 символов 64-символьного алфавита?

15) В корзине 9 красных мячей, 12 желтых мячей и 3 зеленых. Сколько бит информации содержится в сообщении «достали зеленый мяч».

16) Выполнить перевод числа $126_8 \rightarrow N_{10}$.

17) Выполнить перевод числа $42_{10} \rightarrow N_{16}$.
18) Выполнить перевод числа $28_{10} \rightarrow N_{2}$.
19) Выполнить перевод числа $36_{8} \rightarrow N_{10}$.
20) Выполнить перевод числа $A2_{16} \rightarrow N_{10}$.
21) Выполнить перевод числа $29_{10} \rightarrow N_{2}$.
22) Выполнить перевод числа $34_{10} \rightarrow N_{2}$.
23) Выполнить перевод числа $1D4_{16} \rightarrow N_{10}$.
24) Составьте таблицу истинности сложного высказывания
$f = x \vee \overline{y} \wedge \overline{x} \vee y$.
25) Упростить формулу
$$f(x, y) = x \wedge y \vee \overline{x} \wedge y \vee x \wedge \overline{y} \vee \overline{x} \wedge \overline{y}.$$
26) Упростить формулу
$$(x \vee y) \wedge (x \vee \overline{y}).$$
27) Составьте формулу для данной схемы



28) Составить таблицу истинности для логической функции $A \rightarrow C \vee E$.
29) Для какого числа X истинно высказывание ((X>2)∨(X<2)) ⇒ (X>4)?
30) Составить таблицу истинности для логической функции $(A \leftarrow C) \vee E$.
31) Для какого числа X истинно высказывание (X>4)∨((X>1) ⇒ (X>4))?
32) Составить таблицу истинности для логической функции $A \rightarrow C \vee E$.
33) Для какого числа X истинно высказывание ((X>5)∧(X<5)) ∨ (X>7)?
34) Для какого числа X истинно высказывание (X>5) ⇒ ((X>3)∨(X<3))?
35) Составить таблицу истинности для логической функции $\neg B(A \leftarrow C)$.

3.3. Примеры контрольных работ раздела 2 «Использование программных систем и сервисов»

Примеры контрольной работы по текстовому процессору

Пример 1

При помощи программы LO Writer создайте новый документ. Установите поля на странице: верхнее 2,5, нижнее 1, левое 2,5, правое 2. Установите автоматическую расстановку переносов (Сервис, Язык, Расстановка переносов) и наберите следующий текст, сохраняя все параметры форматирования символов и абзацев (шрифт Courier New, размер 14):

Правила редактирования

Выравнивание и пробелы – не применяйте для форматирования текста и выравнивания формул пробел! Для выравнивания текста в текстовом процессоре есть специальные команды выравнивания.

Между словами не нужно ставить более одного пробела, тем более не нужно ставить пробелы в слове для получения разреженного текст.

Добавьте таблицу:

ВИДЫ ОСНОВНЫХ ПРОИЗВОДСТВЕННЫХ ФОНДОВ	На начало года		На конец года	
	млрд. руб.	%	млрд. руб.	%
1	2	3	4	5
Здания	872		948	
Сооружения	70		70	

Добавьте текст и отформатируйте его с использованием табуляции и маркированного списка:

Переход внутри документа можно осуществлять и с помощью клавиатуры:

- Page Down ---------- перемещение на одну запись вперед;
- Page Up ---------- переход на одну запись назад;



$$F = \begin{cases} \int_0^1 \sin(ax+5)dx \ при \ x \le 0{,}5 \\ \dfrac{x}{ax-2.3x} \ при \ x > 0{,}5 \end{cases}$$

Примеры контрольной работы по электронным таблицам

Пример 1

a) В новой книге Calc на Листе 1 создайте, начиная с ячейки A5, таблицу «Сведения об авиапассажирах» (столбец № п/п заполнить с помощью автозаполнения). В ячейку A1 введите текст: «Стоимость 1 кг багажа, руб.». В ячейку A2 введите размер стоимости – 2.

b) Рассчитайте величину оплаты за багаж по формуле:

*Оплата за багаж = Стоимость 1 кг багажа * Вес вещей*
Вычислите итоговое значение в столбцах «Вес вещей» и «Оплата за багаж».

Сведения об авиапассажирах

№ п/п	Фамилия	Имя	Пункт назначения	Вес вещей, кг	Оплата за багаж
1	**Петухов**	**Иван**	**Германия**	**16,8**	
2	**Воронов**	**Сергей**	**Польша**	**20,5**	
3	**Соркин**	**Василий**	**Германия**	**18,4**	
4	**Гусев**	**Роман**	**Канада**	**14,4**	
5	**Воронов**	**Алексей**	**Канада**	**22,6**	
6	**Воронов**	**Александр**	**Германия**	**17,5**	
7	**Гусев**	**Александр**	**Польша**	**21,1**	
8	**Грачев**	**Иван**	**Канада**	**13,9**	
Итого:					

Задание 2

a) Начиная с ячейки H5 создайте и заполните, используя Мастер функций, таблицу следующего вида:

Максимальный вес вещей	

b) Постройте диаграмму для сравнения оплаты за багаж пассажиров и поместите ее на отдельном листе.*Задание 3*

a) Сделайте копию Листа 1, переименуйте его на «Сортировка1» и отсортируйте таблицу так, чтобы пункты назначения располагались по алфавиту.

b) Выполните фильтрацию: сделайте копию Листа 1, переименуйте его на «Фильтр1» и оставьте на нем данные о пассажирах по фамилии Гусев, у которых вес вещей больше 20 кг.

Примеры контрольной работы по базам данных

Пример 1

1) Создайте базу данных с именем Экзамен, содержащую таблицу на основе исходной таблицы. В соответствии с таблицей установите правильные типы данных полей, их размер и необходимое форматирование.

Распределение посевных площадей в процентах (%)

Номер культуры	Культуры	Сельхозорганизации	Хозяйства населения	Фермерские
00100	Зерновые	84,8	0,9	14,1
00101	Технические	74,4	1,7	23,9
00102	Картофель	20,0	32,0	20,9
00103	Овощебахчевые	20,0	77,2	2,8
00104	Кормовые	95,3	2,2	2,5

00105	Зернобобовые	40,8	3,3	10,3

2) Создайте запрос: Выведите данные по культурам, у которых процент площадей на сельхозогранизации равен 20, или хозяйства населения меньше 1 или фермерские равен 2,5;

3) Создайте запрос, запрашивающий при запуске название культуры.

4) Создайте базу данных с именем Экзамен, содержащую одну таблицу на основе исходной таблицы. В соответствии с таблицей установите правильные типы данных полей, их размер и необходимое форматирование.

Анализ алгоритмов в профессиональной области

Предлагается некоторая операция над двумя произвольными трехзначными десятичными числами:

1) записывается результат сложения старших разрядов этих чисел;

2) к нему дописывается результат сложения средних разрядов по такому правилу: если он меньше первой суммы, то полученное число приписывается к первому слева, иначе — справа;

3) итоговое число получают приписыванием справа к числу, полученному после второго шага, суммы значений младших разрядов исходных чисел.

Какое из перечисленных чисел могло быть построено по этому правилу?

141310

102113

101421

101413

Определите значение целочисленных переменных x, y и t после выполнения фрагмента программы:

```
x := 5;
y := 7;
t := x;
x := y mod x;
y := t;
```

x=2, y=5, t=5

x=7, y=5, t=5

x=2, y=2, t=2

x=5, y=5, t=5

Определите значение переменной c после выполнения следующего фрагмента программы:

```
a := 6;
b := 15;
a := b – a*2;
if a > b
then c := a + b
else c := b – a;
```

–3

33
18
12
У исполнителя Калькулятор две команды, которым присвоены номера:
1. прибавь 2
2. умножь на 3
Выполняя первую из них, Калькулятор прибавляет к числу на экране 2, а выполняя вторую, утраивает его. Запишите порядок команд в программе получения из 0 числа 28, содержащей не более 6 команд, указывая лишь номера команд.
Например, программа 21211 — это программа:
умножь на 3
прибавь 2
умножь на 3
прибавь 2
прибавь 2
которая преобразует число 1 в 19.
Ответ _____
Определите значение переменной y, которое будет получено в результате выполнения следующей программы:

```
var i, y: integer;
begin
y := 0;
for i := 1 to 4 do
begin
y := y * 10;
y :=y + i;
end
end.
```

Ответ _____
Определите значение переменной y, которое будет получено в результате выполнения следующей программы:

```
var y : real; i : integer;
begin
y := 0;
i := 1;
repeat
i :=2*i;
y := y + i
until i > 5;
end.
```

Ответ _____
Определите значение переменной y, которое будет получено в результате выполнения следующей программы:

```
var y : real; i : integer;
begin
y := 0;
i := 5;
while i>2 do
begin
i:=i − 1;
y := y + i * i
end;
end.
```

Ответ _____

В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do
A[i]:= i + 1;
for i:=0 to 10 do
A[i]:= A[10-i];
```

Чему будут равны элементы этого массива?

10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0

11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1

11, 10, 9, 8, 7, 6, 7, 8, 9, 10, 11

10, 9, 8, 7, 6, 5, 6, 7, 8, 9, 10

Все элементы двумерного массива A размером 5x5 равны 0. Сколько элементов массива после выполнения фрагмента программы будут равны 1?

```
for n:=1 to 5 do
for m:=1 to 5 do
A[n,m] := (m – n)*(m – n);
```

2

5

8

14

В программе описан одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do
A[i]:=i-1;
for i:=1 to 10 do
A[i-1]:=A[i];
A[10]:=10;
```

Как изменятся элементы этого массива после выполнения фрагмента программы?

все элементы, кроме последнего, окажутся равны между собой

все элементы окажутся равны своим индексам

все элементы, кроме последнего, будут сдвинуты на один элемент вправо

все элементы, кроме последнего, уменьшатся на единицу
Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:
F(1) = 1
F(n) = F(n–1) * (2*n + 1), при n > 1
Чему равно значение функции F(4)?
27
9
105
315
Дан рекурсивный алгоритм:
```
procedure F(n: integer);
begin
writeln('*');
if n > 0 then begin
F(n-3);
F(n div 2);
end
end;
```
Сколько символов «звездочка» будет напечатано на экране при выполнении вызова F(7)?
7
10
13
15
Дан рекурсивный алгоритм:
```
procedure F(n: integer);
begin
writeln(n);
if n < 5 then begin
F(n+3);
F(n*3)
end
end;
```
Найдите сумму чисел, которые будут выведены при вызове F(1).
Ответ _____
Ниже записан рекурсивный алгоритм F:
```
function F(n: integer): integer;
begin
if n > 2 then
F := F(n-1)+F(n-2)+F(n-3)
else
F := n;
end;
```

Чему будет равно значение, вычисленное алгоритмом при выполнении вызова F(6)?

Ответ _____

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на четырех языках программирования

Алгоритмический язык	Паскаль	Python	C++

алг
нач
цел n, s
s := 0
n := 95
нц пока s + n < 177
s := s + 10
n := n - 5
кц
вывод n
кон
	var s, n: integer;
	begin
	s := 0;
	n := 95;
	while s + n < 177
	do
	begin
	s := s + 10;
	n := n - 5
	end;
	writeln(n)
	end.
		s = 0
		n = 95
		while s + n < 177:
		s = s + 10
		n = n - 5
		print(n)
			#include
			<iostream>
			using namespace
			std;
			int main() {
			int s = 0, n = 95;
			while (s + n <
			177) {
			s = s + 10;
			n = n - 5;
			}
			cout << n <<
			endl;
			return 0;
			}

Ответ _____

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на четырех языках программирования

Алгоритмический язык	Паскаль	Python	C++

алг
нач
цел n, s
s := 0
n := 90
нц пока s + n < 145
s := s + 15
n := n - 5
кц
вывод n
кон
	var s, n: integer;
	begin
	s := 0;
	n := 90;
	while s + n < 145
	do
	begin
	s := s + 15;
	n := n - 5
	end;
	writeln(n)
	end.
		s = 0
		n = 90
		while s + n < 145:
		s = s + 15
		n = n - 5
		print(n)
			#include
			<iostream>
			using namespace
			std;
			int main() {
			int s = 0, n = 90;
			while (s + n <
			145) {
			s = s + 15;
			n = n - 5;
			}
			cout << n <<
			endl;
			return 0;
			}

Ответ _____

Ниже приведена программа, записанная на пяти языках программирования

Алгоритмический язык	Паскаль	Python	C++

алг
нач
цел s, t, A
ввод s
ввод t
ввод A
если s > 10 или t > A
то вывод "YES"
иначе вывод "NO"
все
кон
	var s, t, A: integer;
	begin
	readln(s);
	readln(t);
	readln(A);
	if (s > 10) or (t > A)
	then
	writeln("YES")
	else
	writeln("NO")
	end.
		s = int(input())
		t = int(input())
		A = int(input())
		if (s > 10) or (t > A):
		print("YES")
		else:
		print("NO")
			#include <iostream>
			using namespace std;
			int main(){
			int s, t, A;
			cin >> s;
			cin >> t;
			cin >> A;
			if (s > 10) or (t > A)
			cout << "YES" << endl;
			else
			cout << "NO" << endl;
			return 0;
			}

Было проведено 9 запусков программы, при которых в качестве значений переменных s и t вводились следующие пары чисел: (1, 2); (11, 2); (1, 12); (11, 12); (–11, –12); (–11, 12); (–12, 11); (10, 10); (10, 5). Укажите количество целых значений параметра A, при которых для указанных входных данных программа напечатает «YES» шесть раз.

Ответ _____

Ниже приведена программа, записанная на пяти языках программирования

Алгоритмический язык	Паскаль	Python	C++

алг
нач
цел s, t, A
ввод s
ввод t
ввод A
если s > 10 или t > A
то вывод "YES"
иначе вывод "NO"
все
кон
	var s, t, A: integer;
	begin
	readln(s);
	readln(t);
	readln(A);
	if (s > 10) or (t > A)
	then
	writeln("YES")
	else
	writeln("NO")
	end.
		s = int(input())
		t = int(input())
		A = int(input())
		if (s > 10) or (t > A):
		print("YES")
		else:
		print("NO")
			#include <iostream>
			using namespace std;
			int main(){
			int s, t, A;
			cin >> s;
			cin >> t;
			cin >> A;
			if (s > 10) or (t > A)
			cout << "YES" << endl;
			else
			cout << "NO" <<

```
endl;
 return 0;
 }
```

Было проведено 9 запусков программы, при которых в качестве значений переменных s и t вводились следующие пары чисел: (1, 2); (11, 2); (1, 12); (11, 12); (–11, –12); (–11, 12); (–12, 11); (10, 10); (10, 5). Укажите наименьшее целое значение параметра A, при котором для указанных входных данных программа напечатает «NO» семь раз.

Ответ _____

Ключ к тесту

№	Ответ
	г
	а
	г
	121211
	1234
	14
	29
	в
	в
	б
	г
	г
	42
	20
	10
	60
	5
	12

Модельные примеры оценочных средств для проведения промежуточного и рубежного контроля по прикладному модулю 2 «Аналитика и визуализация данных на Python»

Выполнение контрольной работы предусмотрено после изучения темы 2.3. Работа со списками и словарями

1. Определите, что будет напечатано в результате выполнения программы:

```
z = 30
for n in range(10):
        if n < 0:
                z = z – 2 * n
else:
                z = n - z
```

```
print(z)
```
Ответ: -35

2. Определите, что будет напечатано в результате выполнения программы:

```
a = 23
b = 4
while a > b:
        if a % 2 == 0:
                b = b + a
else:
                a = a – 2 * b + 1
print(b)
```
Ответ: 20

3. Определите, что будет напечатано в результате выполнения программы:

```
s = 0
m = 123
while m > 0:
        d = m % 10
        s = s + d
m = m // 10
print(s)
```
Ответ: 6

4. Дан список из чисел и индекс элемента в списке k. Удалите из списка элемент с индексом k, сдвинув влево все элементы, стоящие правее элемента с индексом k. Программа получает на вход список, затем число k. Программа сдвигает все элементы, а после этого удаляет последний элемент списка при помощи метода pop() без параметров.

Программа должна осуществлять сдвиг непосредственно в списке, а не делать это при выводе элементов. Также нельзя использовать дополнительный список. Также не следует использовать метод pop(k) с параметром.

Решение:

```
a = [int(s) for s in input().split()]
k = int(input())
for i in range(k, len(a) - 1):
  a[i] = a[i + 1]
```

```
a.pop()
print(' '.join([str(i) for i in a]))
```

5. Предприятие производит оптовую закупку некоторых изделий A и B, на которую выделена определённая сумма денег. У поставщика есть в наличии партии этих изделий различных модификаций по различной цене. На выделенные деньги необходимо приобрести как можно больше изделий A независимо от модификации. Если у поставщика закончатся изделия A, то на оставшиеся деньги необходимо приобрести как можно больше изделий B. Известны выделенная для закупки сумма, а также количество и цена различных модификаций данных изделий у поставщика. Необходимо определить, сколько будет закуплено изделий B и какая сумма останется неиспользованной.

Входные данные.

Первая строка входного файла содержит два целых числа: N — общее количество партий изделий у поставщика и M — сумма выделенных на закупку денег (в рублях). Каждая из следующих N строк описывает одну партию и содержит два целых числа (цена одного изделия в рублях и количество изделий в партии) и один символ (латинская буква A или B), определяющий тип изделия. Все данные в строках входного файла отделены одним пробелом.

В ответе запишите два целых числа: сначала количество закупленных изделий типа B, затем оставшуюся неиспользованной сумму денег.

Пример входного файла:

```
4 1000
30 8 A
50 12 B
40 14 A
30 60 B
```

В данном случае сначала нужно купить изделия A: 8 изделий по 30 рублей и 14 изделий по 40 рублей. На это будет потрачено 800 рублей. На оставшиеся 200 рублей можно купить 6 изделий B по 30 рублей. Таким образом, всего будет куплено 6 изделий B и останется 20 рублей. В ответе надо записать числа 6 и 20.

Решение

```
f = open('26.txt')
x, y = f.readline().split()
y = int(y)
for_B_price = []
for_B_kol = []
```

```
for i in f:
  if 'A' in i:
    a, b, c = i.split()
    y -= int(a) * int(b)
  else:
    a2, b2, c2 = i.split()
    for_B_price.append(int(a2))
    for_B_kol.append(int(b2))
mini = min(for_B_price)
index_mini = 0
for i in range(len(for_B_price)):
  if mini == for_B_price[i]:
    index_mini = i
kol_B = 0
while y > for_B_price[index_mini]:
  y -= for_B_price[index_mini]
  for_B_kol[index_mini] -= 1
  kol_B += 1
  if for_B_kol[index_mini] == 0:
    for_B_price[index_mini] = 1000000000
    mini = min(for_B_price)
    for i in range(len(for_B_price)):
      if mini == for_B_price[i]:
        index_mini = i
print(kol_B, y)
```

Отметка	**Число баллов, необходимое для получения отметки**
«3» (удов.)	Выполнены три задания (№ 1, 2, 3)
«4» (хорошо)	Выполнены четыре задания (№ 1, 2, 3, 4) Выполнены два задания (№ 4, 5)
«5» (отлично)	Выполнены все пять заданий

Рубежный контроль

Рубежный контроль проходит в форме выполнения и защиты проектной работы Проектная работа «Анализ больших данных в профессиональной сфере»

Основные этапам работы над проектом: определение проблемы в предметной области; извлечение данных; подготовка данных — очистка

данных и преобразование данных; исследование и визуализация данных; построение предсказательной модели; интерпретация результатов.

Проект выполняется на примере открытых данных, например, Сбербанка https://www.kaggle.com/code/umkkka/eda-sberbank-open-data-russian-version/data. Обучающиеся самостоятельно подбирают набор данных на https://www.kaggle.com/ в профессиональной сфере и выполняют исследование и визуализацию данных.

Пример выполнение проекта **«Прогнозирование инсульта»**

Определение проблемы в предметной области. По данным Всемирной организации здравоохранения (ВОЗ) инсульт является 2-й ведущей причиной смерти в мире, ответственной примерно за 11% всех смертей. Определим вероятность развития инсульта среди пациентов. Узнаем, что является фактором возникновения инсульта в большей степени.

Каждая строка данных содержит соответствующую информацию о пациенте.

id: уникальный идентификатор

gender: «Male», «Female» or «Other» - пол: «Мужской», «Женский» или «Другой»

age: age of the patient - возраст пациента

hypertension: 0, если у пациента нет гипертонии, 1, если у пациента гипертония

heart disease: 0, если у пациента нет сердечных заболеваний, 1, если у пациента сердечное заболевание

ever married: «No» or «Yes» – состоят ли в браке: «нет», «да»

work type: «children», «Govt job», «Never_worked», «Private» or «Self-employed» – тип работы: «с детьми», «работа в правительстве», «никогда не работал», «частное предпринимательство», «самозанятый».

Residence type: «Rural» or «Urban» – тип местности: сельский, городской.

Avg glucose level: average glucose level in blood - средний уровень глюкозы в крови.

bmi: body mass index – индекс массы тела

smoking status: «formerly smoked», «never smoked», «smokes» or «Unknown»* - «ранее курил», «никогда не курил», «курит» или «Неизвестно»*

stroke: 1, если у пациента был инсульт, или 0, если нет.

Note: «Unknown» (неизвестный) в состоянии курения означает, что информация недоступна для данного пациента.

С использованием набора данных проанализирована вероятность появления инсульта у пациента на основе входных параметров, таких как пол (gender), возраст (age), различные заболевания: гипертония (hypertension), сердечные заболевания (heart disease), а также статус курения (smoking status).

Извлечение данных.

В качестве основной библиотеки графики используется Plotly. Так как этот пакет с открытым исходным кодом, который предлагает сложные графики, которые могут удовлетворить требования любого проекта. Ведь одно из достоинств Plotly - это интерактивность исследования данных с помощью графиков и диаграмм.

Seaborn (sns) - библиотека визуализации данных Python, основанная на matplotlib. Она предоставляет высокоуровневый интерфейс для рисования привлекательной и информативной статистической графики. (Рис. 1)

```
In [2]: import numpy as np
        import matplotlib.pyplot as plt
        import pandas as pd
        import seaborn as sns
        plt.style.use('seaborn-whitegrid')

        import plotly.express as px
        import plotly.graph_objects as go
```

Рис. 1

После того, как скачаны и загружены данные из kaggle.com, следующим шагом стало создание DataFrame для панд на основе данных CSV. (Рис. 2)

```
In [3]: dataset = df=pd.read_csv(r'C:\Users\Светлана\Desktop\healthcare-dataset-stroke-data.csv')
```

Рис. 2

Необходимо, чтобы все столбцы (и числовые, и строковые) входных данных были включены в выходные данные. А выход будет варьироваться в зависимости от того, что предусмотрено. (Рис. 3)

```
В [4]: dataset.describe(include = 'all')
```

	я бы	Пол	возраст	гипертония	болезнь сердца	когда-либо женат	work_type	Residence_type	avg_gluosis_level	bmi	Курение_статус
считать	5110.000000	5110	5110.000000	5110.000000	5110.000000	5110	5110	5110	5110.000000	4909.000000	5110
ликальный	NaN	3	NaN	NaN	NaN	2	5	2	NaN	NaN	4
вершина	NaN	женский	NaN	NaN	NaN	да	Частный	Городской	NaN	NaN	никогда не курил
частота	NaN	2994	NaN	NaN	NaN	3353	2925	2596	NaN	NaN	1892
иметь в виду	36517.829354	NaN	43.226614	0.097456	0.054012	NaN	NaN	NaN	106.147677	28.893237	NaN
часы	21161.721625	NaN	22.612647	0.296607	0.226063	NaN	NaN	NaN	45.283560	7.854067	NaN
мин	67.000000	NaN	0.080000	0.000000	0.000000	NaN	NaN	NaN	55.120000	10.300000	NaN
25%	17741.250000	NaN	25.000000	0.000000	0.000000	NaN	NaN	NaN	77.245000	23.500000	NaN
50%	36932.000000	NaN	45.000000	0.000000	0.000000	NaN	NaN	NaN	91.885000	28.100000	NaN
75%	54682.000000	NaN	61.000000	0.000000	0.000000	NaN	NaN	NaN	114.090000	33.100000	NaN
Максимум	72940.000000	NaN	82.000000	1.000000	1.000000	NaN	NaN	NaN	271.740000	97.600000	NaN

Рис. 3

Подготовка данных.

Проверим качество данных и типы каждого столбца. В данном коде содержится информация о фрейме данных, а также количество памяти, используемой данным кадром. (Рис. 4)

```
В [5]: dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   id                 5110 non-null   int64  
 1   gender             5110 non-null   object 
 2   age                5110 non-null   float64
 3   hypertension       5110 non-null   int64  
 4   heart_disease      5110 non-null   int64  
 5   ever_married       5110 non-null   object 
 6   work_type          5110 non-null   object 
 7   Residence_type     5110 non-null   object 
 8   avg_glucose_level  5110 non-null   float64
 9   bmi                4909 non-null   float64
 10  smoking_status     5110 non-null   object 
 11  stroke             5110 non-null   int64  
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

Рис. 4

С помощью метода head() можно запросить отображения первых пяти строк кадра данных. (Рис. 5)

```
В [7]: dataset.head()
```

	я бы	Пол	возраст	гипертония	болезнь сердца	когда-либо женат	work_type	Residence_type	avg_glucosis_level	bmi	Курение_статус	stroke
0	9046	Мужчина	67.0	0	1	да	Частный	Городской	228.69	36.6	ранее курил	1
1	51676	женский	61.0	0	0	да	Частный предприниматель	Деревенский	202.21	NaN	никогда не курил	1
2	31112	Мужчина	80.0	0	1	да	Частный	Деревенский	105.92	32.5	никогда не курил	1
3	60182	женский	49.0	0	0	да	Частный	Городской	171.23	34.4	курит	1
4	1665	женский	79.0	1	0	да	Частный предприниматель	Деревенский	174.12	24.0	никогда не курил	1

Рис. 5

В проекте нужно узнать общее количество людей и их особенностей. (Рис. 6)

```
В [12]: print("Number of records (patients): ", df.shape[0])
        print("Number of columns (patients): ", df.shape[1])

        Number of records (patients):  5110
        Number of columns (patients):  12
```

Рис. 6

Исследование и визуализация данных.

Рассмотрим распределение пациентов по полу.

Основным элементом изображения, которое строит pyplot является фигура (Figure), на нее накладывается график, оси, надписи и т.д.

Для построения графика используется команда countplot() библиотеки Seaborn. В коде использовался размер графического элемента, был выбран столбец "Пол" в dataset, подписан заголовок и обозначен размер текста, введено название для x и y, и выведено на экран. (Рис. 7)

```
[13]: plt.figure(figsize = (10, 6))
      sns.countplot(x = dataset['gender'])
      plt.title('Распределение пациентов по полу', fontsize = 30)
      plt.xlabel('Пол пациента')
      plt.ylabel('Количество пациентов')
      plt.show()
```

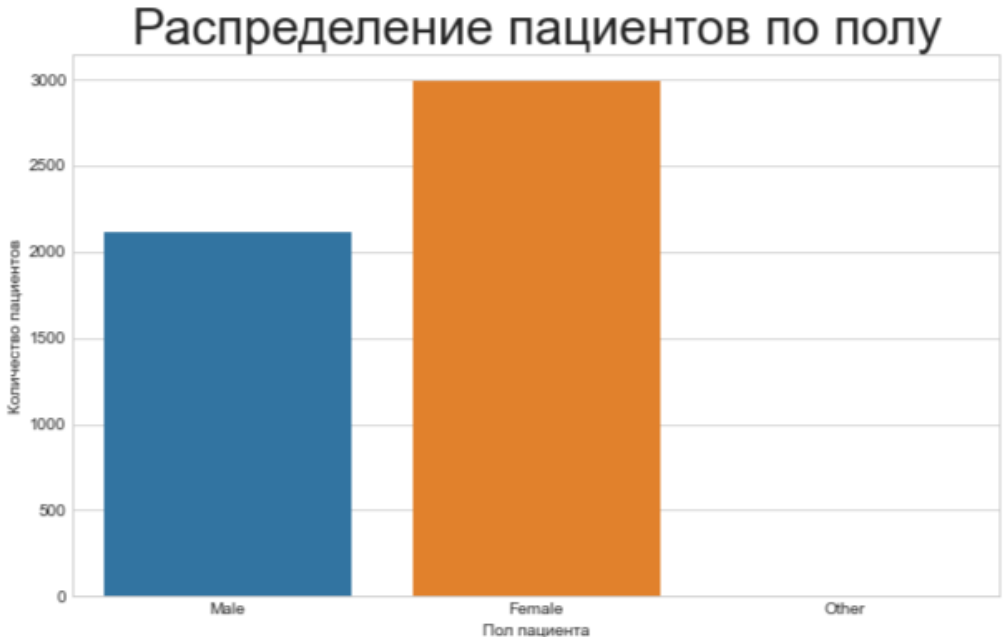


Рис. 7

По диаграмме можно увидеть, что женский пол преобладает, но в соотношении 100% не совсем понятно. Поэтому воспользуемся библиотекой plotly.graph_objects для создания круговой диаграммы. Для того, чтобы ее сделать необходимо использовать функцию go.Pie.

Для создания кода применяем атрибут labels значений "Пола пациента" из dataset. Задано наглядное поведение графика с помощью update_traces: а именно размер текста значений, цвет и жирность границ. А с помощью update_layout выведен текст заголовка. (Рис. 8)

```
In [14]: colors = ['RoyalBlue', 'crimson', 'green']
         fig = go.Figure(data = [go.Pie(labels = dataset['gender'])])
         fig.update_traces(textfont_size = 25, marker = dict(colors = colors, line = dict(color = '#000000', width = 3)))
         fig.update_layout(title_text = 'Соотношение пациентов по полу')
         fig.show()
```

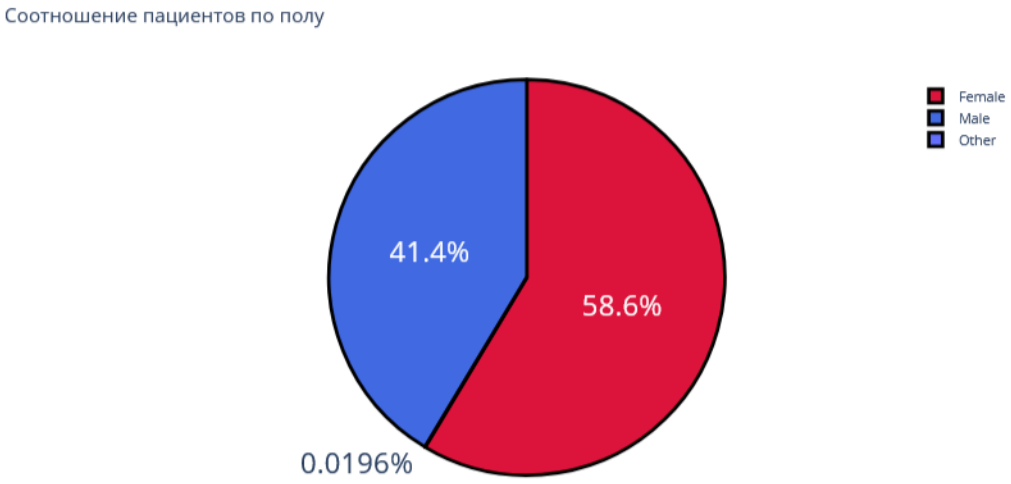


Рис. 8

Модельные примеры оценочных средств для проведения промежуточного и рубежного контроля по прикладному модулю 7 «Введение в веб-разработку на языке JavaScript»

Автоматизированная часть проектной работы

1. Перейдите по адресу https://node-server.online/r?id= #student|37 и в последнем-нижнем поле напишите код функции task(x), возвращающий true, только если в качестве аргумента x передана строка, которая начинается и кончается заглавными латинскими буквами, а между ними находятся только цифры в количестве от 4 до 8, иначе false.
2. Перейдите по https://node-server.online/r?id=x#y|1 и в последнем-нижнем поле напишите код функции task(x), возвращающей массив из числа x, переданного ей в качестве аргумента, и ещё 9 подряд идущих за x чисел, т.е. [x, x + 1, …, x + 9]

3. Перейдите по адресу https://node-server.online/r?id= - student|2 и в последнем-нижнем поле напишите код функции task(x), возвращающей сумму элементов массива, переданного ей в качестве аргумента x

4. Перейдите по адресу https://node-server.online/r?id=x#student|27 и убедившись что в выпадающем списке выбрано obj-001, в последнем-

нижнем поле напишите код функции task(x), возвращающей JSON в виде правильной строки; у исходного объекта должно быть свойство love со строковым значением javascript и свойство year с числовым значением равным текущему году (4 цифры) и свойство arg со значением x.

5. Перейдите по адресу https://node-server.online/r?id=x - student|28 и убедившись что в выпадающем списке выбрано obj-002, в последнем-нижнем поле напишите код функции task(x), возвращающей массив собственных (не унаследованных) перечислимых ключей объекта, переданного ей в качестве аргумента x

6. Перейдите по адресу https://node-server.online/r?id=x - student|12 и убедившись что в выпадающем списке выбрано dom-001, в последнем-нижнем поле напишите код функции task(x), возвращающей
созданный с помощью метода createElement новый DOM-элемент типа span у которого значением атрибута id является x

Творческая часть проектной работы

Посмотрите видеоуроки по начальным шагам работы с серверным JavaScript

https://node-server.online/r/assets/server1-readfile.mp4
https://node-server.online/r/assets/server2-http.mp4

Перейдите по адресу
https://node-server.online/r?id=_#student|20 и убедившись что в выпадающем списке выбрано http-001, в последнем-нижнем поле напишите код функции task(x), возвращающей экземпляр http-сервера, созданный с помощью встроенного Node.js-модуля http (он будет доступен в функции). Этот сервер не должен быть запущен / слушать какой-либо порт.
Используйте только res.end (не нужны заголовки или теги HTML). Он должен обрабатывать два маршрута:
(1) /challenge – в ответ на такой запрос он должен возвращать x
(2) /api/rv/abc

где abc – произвольная строка длиной не менее 1 символа, состоящая только из строчных латинских букв.

В ответ на такой запрос сервер должен возвращать перевёрнутую строку.

Например: https://kodaktor.ru/api/rv/abc

Для других маршрутов предусмотрен ответ No

Чтобы создать основу приложения, выполните следующие шаги:

1. Создайте новый проект:

mkdir $(date +%Y%m%d_%H%M%S) && cd $_ && yarn init –y или
mkdir $(date +%Y%m%d_%H%M%S) && cd $_ && npm init -y
(https://kodaktor.ru/g/init).

2. Установите инструмент nodemon для автоматизации перезапуска сценария и moment для работы с датой и временем: yarn add --dev nodemon или npm i -D nodemon и yarn add moment или npm i moment

```
"scripts" : {
  "start": "nodemon"
},
```

3. Установите настройки линтера и создайте нужный файл .eslintrc.

Создайте в папке проекта файл index.js с содержимым:

```
const http = require('http');
const moment = require('moment');

http.createServer((req, res) => {
  res.end(moment().format('DD.MM.YYYY HH:mm:ss'));
}).listen(4321);
```

4. Запустите сценарий yarn start и выполните curl localhost:4321.

Убедитесь, что в консоли отображается текущая дата и время.

Добавьте к проекту поддержку выдачи данных в формате JSON с выдачей соответствующего заголовка и кодировки UTF-8: http://kodaktor.ru/gitcheckout.gif

```
const http = require('http');
const moment = require('moment');

http.createServer((req, res) => {
  res.setHeader('Content-Type', 'application/json; charset=utf-8');
  res.end(JSON.stringify({ date: moment().format('DD.MM.YYYY HH:mm:ss') }));
}).listen(4321);
```

5. Перейдитепо адресу localhost:4321 в браузере и убедитесь, что выдаётся ответ в формате JSON.

6. Осуществите рефакторинг кода так, чтобы коллбэк, отвечающий на запросы, явным образом указывался для события request:

```
const http = require('http');
const moment = require('moment');

const server = http.createServer();
server.listen(4321);
server.on('request', (req, res) => {
  res.setHeader('Content-Type', 'application/json; charset=utf-8');
  res.end(JSON.stringify({ date: moment().format('DD.MM.YYYY HH:mm:ss') }));
});
```

Типовая процедура защиты результата проектной работы

При подготовке выступления для защиты проекта следует руководствоваться следующей дорожной картой презентации:

- обзор по теме;
- демонстрация в live-режиме (slides.com, например http://slides.com/elizabethanatskaya-1/deck-2#/12 и др.);
- выводы;
- поддержка в репозитории (ссылки на слайды / ресурсы / ...).

Критерии оценивания. Задание считается выполненным, если программа (сценарий) разработана и соответствует заданию, не содержит синтаксических ошибок, а также сопровождается репозиторием (в том числе, возможно, отчётом в форме слайдов). Часть заданий проверяется (или дополнительно проверяется) с помощью средств автоматизированной проверки. Веб-сценарии должны быть размещены на ресурсе, допускающем обращение через Интернет (веб-портфолио, например на Яндекс.Диске или с помощью GitHub Pages)

Шкала итоговой оценки

Шкала	Критерии
зачтено	**обучающийся должен** : продемонстрировать знания изучаемых аспектов JavaScript в полном объёме: дать корректные ответы на 60% тестовых заданий и более, распознавать операторы и структуры данных JavaScript, исправлять ошибки в языковых конструкциях, быть способным корректно сформулировать определения, воспроизвести по запросу информацию о функциональности всех изучаемых средств разработки и проектирования веб-приложений, должно быть выполнено 85% и более заданий самостоятельной работы , представленных в виде корректно функционирующего веб-портфолио.

не зачтено	**обучающийся**: не знает значительной части программного материала (ответил менее чем на 60% тестовых заданий); не владеет понятийным аппаратом дисциплины; не способен продемонстрировать знание ни одной синтаксической конструкции JavaScript и не способен оценить корректность работы веб -сценария; выполнил менее 85% заданий самостоятельной работы, которые не отражены в веб-портфолио.

3.6. Примерные темы для индивидуального проекта

1) Вавилонская система счисления.
2) Фрактальная графика.
3) Применение фрактальной графики.
4) Реализации модели в LibreOffice Calc.
5) Сравнение Gimeric и Calc.
6) Создание и редактирование музыкальных файлов.
7) MAC и IP адреса. Что это такое?
8) Решение линейных алгебраических уравнений в ПК МВТУ.
9) Различие в описании алгоритмических структур в учебниках по информатике и ГОСТе.
10) Обзор и сравнение процессоров AMD линейки Rizen и A.
11) Создание фирмы IBM.
12) Для чего нужны различные форматы видео файлов?

4. Контрольно-оценочные средства для промежуточной аттестации

Основным критерием оценки ответа на тест является процент набранных баллов от максимально возможного количества. На кафедре была утвержден следующий критерий оценивания результатов тестирования:

- 85% и более – «отлично»;
- от 70 % до 84 % – «хорошо»;
- от 55 % до 69% - «удовлетворительно»;
- менее 55% - «неудовлетворительно».

Вопросы тестирования изложены в опубликованной литературе:

1. Ялтанцева В.В. Грачева Н.Н., Руденко Н.Б. УМК по дисциплине «Информатика». Тесты по теоретическому курсу: учебно-методическое пособие. – Зерноград: АЧГАА, 2009. – 136 с.

2. Ялтанцева В.В. УМК по дисциплине «Информатика». Тесты к лабораторным работам: методические указания. – Зерноград: АЧГАА, 2009. – 60 с.

Оценка за семестр(ы).

Семестровая оценка определяется как округленное до целого числа среднее геометрическое оценок текущего контроля, полученных в течение семестра.