

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РД
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ РД
«КОЛЛЕДЖ ЭКОНОМИКИ И ПРАВА»**

УТВЕРЖДАЮ

Зам. директора по учебной работе

А.Л. Джалилова А.Л. Джалилова

подпись

ФИО

« 2 » 09 2023 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ**

ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

специальность 09.02.07 «Информационные системы и программирование»

Дербент 2023

Организация – разработчик:

Государственное бюджетное профессиональное образовательное учреждение РД «Колледж экономики и права»

Разработчики:

Качаева Р.М., преподаватель экономических дисциплин ГБПОУ РД «КЭиП»

Одобрена на заседании цикловой (предметной) комиссии профессиональных дисциплин и модулей по специальности 09.02.07 «Информационные системы и программирование»

Протокол № 1 от « 28 » 08 2023 г.

Председатель комиссии  / Гюльмагомедов Т.Х. /

Фонд оценочных средств профессионального модуля «Основы алгоритмизации и программирования» разработан на основе:

- Рабочая программа профессионального модуля разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки от 9 декабря 2016 г. № 1547 (зарегистрирован Министерством юстиции Российской Федерации 26 декабря 2016 г., регистрационный № 44936);
- Примерной основной образовательной программы (регистрационный номер в государственном реестре примерных основных образовательных программ среднего профессионального образования: П-24 от 02.02.2022г.);
- Рабочего учебного плана образовательного учреждения на 2023/2024 учебный год.

Организация – разработчик:

Государственное бюджетное профессиональное образовательное учреждение РД «Колледж экономики и права»

Разработчик:

Качаева Рамина Магомедзагировна, преподаватель информационных дисциплин ГБПОУ РД «КЭиП»

Рекомендована методическим советом ГБПОУ РД «Колледж экономики и права» для применения в учебном процессе.

Заключение Методического Совета № ____ от «__» _____ 2023 г.

СОДЕРЖАНИЕ

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ	Ошибка! Закладка не определена.
ПРОГРАММА ОЦЕНИВАНИЯ.....	Ошибка! Закладка не определена.
КОМПЛЕКТ ТЕСТОВ	Ошибка! Закладка не определена.
ПЕРЕЧЕНЬ ЛАБОРАТОРНЫХ РАБОТ И ПРАКТИЧЕСКИХ ЗАНЯТИЙ.....	3
ВОПРОСЫ ДЛЯ УСТНОГО (ПИСЬМЕННОГО ОПРОСА).....	4
ПЕРЕЧЕНЬ ВОПРОСОВ К ЭКЗАМЕНУ	Ошибка! Закладка не определена.

ПАСПОРТ
ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине ОП.04 Основы алгоритмизации и программирования

Рабочей программой дисциплины «**Основы алгоритмизации и программирования**» предусмотрено формирование следующих компетенций:

Общие компетенции:

ОК-01. Выбирать способы решения задач профессиональной деятельности, применительно различным контекстам.

ОК-02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК-03. Планировать и реализовывать собственное профессиональное и личностное развитие
ОК-04. Эффективно взаимодействовать и работать в коллективе и команде;

ОК-05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста

ОК-09. Пользоваться профессиональной документацией на государственном и иностранном языках

Профессиональные компетенции:

ПК-2.4. Осуществлять обработку, хранение и передачу информации ограниченного доступа.
ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

ПК-1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК-1.2. Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4. Выполнять тестирование программных модулей.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПРОГРАММА ОЦЕНИВАНИЯ

Требования к результатам обучения по дисциплине, формы их контроля и виды оценочных средств

№ п/п	Контролируемые модули, разделы (темы) дисциплины	Индекс контролируем о й компетенции (или её части)	Оценочные средства		Способ контроля
			наименован ие	№ заданий	
1.	Раздел 1. Введение в программирование Тема 1.1. Языки программирования.	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2	Устный опрос Защита рефератов	В соответствии с темой	Устно Письменно
2.	Тема 1.2. Типы данных	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2	Тесты	В соответствии с темой	Устно Тестирова ние
3.	Раздел 2. Основные конструкции языков программирования Тема 2.1. Операторы языка программирования	, ОК-02, ОК-03. ПК 2.1, ПК-2.2	Устный опрос Коллоквиум Контрольная работа Тесты	В соответствии с темой	Устно Письменно Тестирова ние
4.	Раздел 3. Структурное и модульное программирование Тема 3.1. Процедуры и функции	ОК-01, ОК-03. ПК 2.1, ПК-2.2	Устный опрос Тесты	В соответствии с темой	Устно Письменно Тестирова ние
5.	Тема 3.2. Структуризация в программировании	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2	Устный опрос Защита рефератов	В соответствии с темой	Устно
6.	Тема 3.3. Модульное	ОК-01, ОК-02,	Устный опрос	В	Устно

	программирование	ОК-03. ПК 2.1, ПК-2.2, ПК-2.3	Тесты Защита рефератов	соответствии с темой	Тестирова ние
7.	Раздел 4. Указатели и динамические структуры Тема 4.1. Указатели	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК- 2.2	Устный опрос Защита рефератов	В соответствии с темой	Устно Письменно
8.	Раздел 5. Объектно-ориентированное программирование (ООП) Тема 5.1. Основные принципы объектно-ориентированного программирования (ООП)	ОК-01, ОК-02, ПК 2.1, ПК-2.2, ПК-2.3	Устный опрос Тесты	В соответствии с темой	Устно Письменно Тестирова ние
9.	Тема 5.2. Интегрированная среда разработчика.	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2, ПК-2.3	Тесты Защита рефератов	В соответствии с темой	Устно Тестирова ние
10.	Тема 5.3. Визуальное событийно-управляемое программирование	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2, ПК-2.3	Устный опрос Контрольная работа Тесты	В соответствии с темой	Устно Письменно Тестирова ние
11.	Тема 5.4. Разработка оконного приложения	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2, ПК-2.3	Устный опрос Контрольная работа Тесты	В соответствии с темой	Устно Письменно Тестирова ние
12.	Тема 5.5. Этапы разработки приложений	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2, ПК-2.3	Устный опрос Тесты	В соответствии с темой	Устно Письменно Тестирова ние
13.	Тема 5.6. Иерархия классов.	ОК-01, ОК-02, ОК-03. ПК 2.1, ПК-2.2, ПК-2.3	Устный опрос Тесты	В соответствии с темой	Устно Письменно Тестирова ние

КОМПЛЕКТ ТЕСТОВ

Тестовые задания

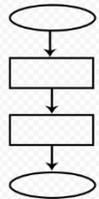
Раздел 1. Введение в программирование

Тема 1.2. Типы данных

1. Сколько возможных значений у переменной типа bool?
 - a. 2
 - b. 4
 - c. 10
 - d. Сколько угодно
2. Определить тип данных переменной $a=2.5$:
 - a. Целочисленное
 - b. Вещественное
 - c. Строковое
 - d. Булевоe
3. Определите тип данных переменной `name = 'Alexander'`:
 - a. вещественный
 - b. символьный
 - c. логический
 - d. целый
4. Определите тип данных переменной `age= 21`:
 - a. вещественный
 - b. символьный
 - c. логический
 - d. целый
5. Что обозначает тип данных `str`?
 - a. вещественный

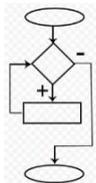
- b. символьный
 - c. логический
 - d. целый
6. Тип данных float представляет?
- a. Целочисленное
 - b. Вещественное
 - c. Строковое
 - d. Булево

7. Назовите тип алгоритма:



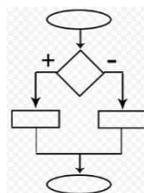
- a. Разветвляющийся
- b. Линейный
- c. Циклический
- d. Смешанный

8. Назовите тип алгоритма:



- a. Линейный
- a. Разветвляющийся
- b. Циклический
- c. Смешанный

9. Назовите тип алгоритма:



- 1. Разветвляющийся
 - 2. Линейный
 - 3. Смешанный
 - 4. Циклический
10. Какой алгоритм задает многократное выполнение команд?
- a. Разветвляющийся
 - b. Линейный
 - c. Циклический

Тема 2.1. Операторы языка программирования

- 1. В заданном примере укажите цикл:

- a) `a = int(input())`
- b) `while a != 0:`
- c) `print(a)`
- d) `a = int(input())`

2. В заданном примере укажите команду вывод результата выполнения цикла:

- a) `a = int(input())`
- b) `while a != 0:`
- c) `print(a)`
- d) `a = int(input())`

3. Укажите цикл, который проверяет условие и выполняет код, и так пока условие не перестанет выполняться?

- a) While
- b) for
- c) print

d) `a = int(input())` 4. Укажите команду ввода данных:

- a) while
- b) for
- c) print
- d) `a = int(input())`

5. Укажите цикл, который выполняет один и тот же код несколько раз подряд, просто изменяя значения одной переменной :

- a) while
- b) for
- c) print
- d) `a = int(input())`

6. Как правильно объявляется множество?

- a) `a = { }`
- b) `a = []`
- c) `a = set()`
- d) `a = set`

7. Какие методы удаляют элемент из множества? а) `remove()`

- б) `discard()`
- в) `print()`
- г) `get()`

8. Для получения длины множества применяется: `len()`

`add()` `remove()` `print()`

9. Какая команда создает множества users:

- а) `users = {"Tom", "Bob", "Alice", "Tom"}` б) `users.add("Sam")`
- в) `print(users)`

10. Какая команда добавляет элемент в множество :

- a) `users = {"Tom", "Bob", "Alice", "Tom"} б) users.add("Sam")`
в) `print(users)`

Раздел 3. Структурное и модульное программирование Тема 3.1. Процедуры и функции

1. Что такое функция?
- a) Фрагмент кода для повторного использования, который применяется для выполнения одного связанного действия.
b) Структура, определяющая поведение объекта
c) Любой код в Python
d) Код, заключенный в круглые скобки
2. Какое ключевое слово используется для создания функции?
- a) fun
b) function
c) void
d) def
3. Какая функция выводит что-либо в консоль? а) log();
b) out();
c) write();
d) print();
4. Что такое рекурсивная функция?
- a) Функция, которая курсирует между модулями
b) Функция, которая возвращает саму себя
c) Функция, которая выполняется с последней строки по первую
d) Функция, написанная курсивом
5. Укажите имя функции в примере: `def summa(n):`
- a) def
b) summa
c) n
6. Укажите параметр функции в примере: `def summa(n):`
- a) def
b) summa
c) n
7. Укажите параметр функции в примере: `def sum(x, y):`
- a) def
b) sum
c) x, y
8. Укажите имя функции в примере: `def sum(x, y):`
- d) def
e) sum
f) x, y
3. Задайте функции `sum()` аргументы:
- a) def
b) Sum

- c) (a, b)
d) (1, 2)
9..Задайте функции sum() параметры:

- e) def
f) Sum
g) (a, b)
h) (1, 2)

10. Вставьте пропущенный код рекурсивной функции: _____factorial(n):

- a) def
b) factorial
c) (a, b)
d) (1, 2)

Тема 3.3. Модульное программирование

1. Как добавить модуль в программу?

- a. import math
b. import math()
c. import (math)
d. import.math

2. Какая библиотека отвечает за время?

- a) time
b) Time
c) localtime
d) clock

3. Главным элементом Tkinter является?

- a) окно
b) кнопка
c) радиокнопка
d) текстовое поле

4. Какими операторами можно импортировать модуль?

- a) import
b) from import
c) exes
d) imp

5. Что обозначает тип данных int?

- a. Целочисленное
b. Вещественное
c. Строковое
d. Булевое

6. Укажите команду ввода данных:

- a) while
b) for
c) print
d) a = int(input())

7. Что обозначает тип данных str?

- a. вещественный

- b. символьный
 - c. логический
 - d. целый
8. Какое ключевое слово используется для создания функции?
- a) fun
 - b) function
 - c) void
 - d) def
9. Какая функция выводит что-либо в консоль?
- a) log();
 - b) out();
 - c) write();
 - d) print();
10. Вставьте пропущенный код рекурсивной функции: _____factorial(n):
- a) def
 - b) factorial
 - c) (a, b)
 - d) (1, 2)

Раздел 5. Объектно-ориентированное программирование

Тема 5.1. Основные принципы объектно-ориентированного программирования(ООП)

1. Какой класс Tkinter соответствует виджету для поля ввода?
- a) Label
 - b) Text
 - c) Frame
 - d) Entry
2. Какой класс Tkinter соответствует виджету для надписи текста?
- a) Label
 - b) Text
 - c) Frame
 - d) Entry
3. Какой класс Tkinter соответствует виджету для вывода графических примитивов?
- a) Label
 - b) Text
 - c) Frame
 - d) Canvas
4. Какой метод создает текст в Tkinter?
- a) create_text
 - b) create_image
 - c) create_rectangle
 - d) create_line
5. Набор объектов одного типа под общим именем:
- a) Массив
 - b) Tkinter
 - c) for

- d) Модуль
- 6. Позволяет повторять код определенное количество раз
 - a) Массив
 - b) Tkinter
 - c) Цикл
 - d) Модуль
- 7. Оформляется в виде отдельного файла с исходным кодом
 - a) Массив
 - b) Tkinter
 - c) Цикл
 - d) Модуль
- 9. Графическая библиотека Python
 - a) Массив
 - b) Tkinter
 - c) for
 - d) Модуль
- 8. Виджет, предназначенный для отображения какой-либо надписи без возможности редактирования пользователем.
 - a) Frame
 - b) Scale
 - c) Label
 - d) Button
- 9. Что делает команда import
 - a) импортирует файл модуля
 - b) создает функцию
 - c) удаляет файл
- 10. Выберите вариант правильного удаления переменной a
 - a) del(a)
 - b) delete(a)
 - c) delete=a

Тема 5.2. Интегрированная среда разработчика.

1. Какая функция выводит что-либо в консоль?
 - a) log();
 - b) out();
 - c) write();
 - d) print();
2. Укажите оператор ввода:
 - a) input()
 - b) print()
 - c) int()
 - d) random()
3. Что обозначает тип данных str?
 - а) вещественный
 - б) символьный
 - в) логический

г) целый

4. Что будет в результате выполнения следующего алгоритма:

Входные данные: $57x = \text{int}(\text{input}())$

if $x > 0$:

print(x)else:

print(-x)

5. Что будет в результате выполнения следующего алгоритма:

Входные данные: $40x = \text{int}(\text{input}())$

if $x > 50$: print(x)

else:

print("Ошибка")

6. Что будет в результате выполнения следующего алгоритма:

Входные данные: $40x = \text{int}(\text{input}())$

if $x > 50$: print(x)

else:

print("Ошибка")

7. Что будет в результате выполнения следующего алгоритма:

Входные данные: $50x = \text{int}(\text{input}())$

if $x < 100$:print(x)

else:

print("Ошибка")

8. Что будет в результате выполнения следующего алгоритма:

Входные данные: $100x = \text{int}(\text{input}())$

if $x == 100$:print(x)

else:

print("Ошибка")

9. Для получения длины множества применяется: а) len()

б) add()

в) remove()г) print()

10.Какой метод используется для добавления элемента в множество?

а) add()

б) new()

с) get()

д) remove()

Тема 5.3. Визуальное событийно-управляемое программирование

1. Команда: `from tkinter import *`
 - a) подключает библиотеку Tkinter
 - b) создает корневое окно
 - c) создает корневое окно без содержимого
2. Укажите команду, которая создает кнопку в главном окне:
 - a) `window = Tkinter.Tk()`
 - b) `frame=Tkinter.Frame(window)`
 - c) `button=Tkinter.Button(window)`
3. Какой командой модуля Turtle можно задать цвет пера?
 - a) `color = 'цвет'`
 - b) `color ('цвет')`
 - c) `color.цвет='цвет'`
4. Элемент, предназначен для организации виджетов внутри окна.
 - a) Frame
 - b) Scale
 - c) Label
 - d) Button
5. Виджет, предназначенный для отображения какой-либо надписи без возможности редактирования пользователем.
 - a) Frame
 - b) Scale
 - c) Label
 - d) Button
6. Главным элементом Tkinter является?
 - a) окно
 - b) кнопка
 - c) радиокнопка
 - d) текстовое поле
7. Какими операторами можно импортировать модуль?
 - a) `import`
 - b) `from import`
 - c) `exec`
 - d) `imp`
8. С помощью метода `geometry()` задается:
 - a) отображения окна
 - b) размер окна.
 - c) заголовок окна
 - d) создание окна
9. Сделать размер окна фиксированным метод:
 - a) `resizable()`
 - б) `mainloop()` в) `title()`
 - г) `pack()`
10. Разместит виджет в контейнере (главном окне) метод:
 - a) `resizable()`

- б) mainloop()в) title()
- г) pack()

Тема 5.4. Разработка оконного приложения

1. Команда: `from tkinter import *`
 - d) подключает библиотеку Tkinter
 - e) создает корневое окно
 - f) создает корневое окно без содержимого
2. Укажите команду, которая создает кнопку в главном окне:
 - d) `window = Tkinter.Tk()`
 - e) `frame=Tkinter.Frame(window)`
 - f) `button=Tkinter.Button(window)`
3. Какой командой модуля Turtle можно задать цвет пера?
 - d) `color = 'цвет'`
 - e) `color ('цвет')`
 - f) `color.цвет='цвет'`
3. Виджет, предназначен для организации виджетов внутри окна.
 - e) Frame
 - f) Scale
 - g) Label
 - h) Button
4. Виджет, предназначенный для отображения какой-либо надписи без возможности редактирования пользователем.
 - a) Frame
 - b) Scale
 - c) Label
 - d) Button
5. Какую нужно дописать команду, чтобы запустилось графическое «окно» программы?
`import Tkinter window = Tkinter.Tk()`
`window.title("Моя программа")window.geometry("300x300")`
 - a) `window.go()`
 - b) `window.run()`
 - c) `window.mainloop()`
 - d) `window.load()`
6. Какую нужно дописать команду, чтобы установить размеры окна?
`import Tkinter`
`window = Tkinter.Tk() window.title("Моя программа")window.geometry("300x300")`
 - a) `window.geometry()`
 - b) `window.title ()`
 - c) `window.mainloop()`

- d) `window.load()`
- 7. Какую нужно дописать команду, чтобы установить заголовок окна?`import Tkinter`
`window = Tkinter.Tk()` `window.title("Моя программа")``window.geometry("300x300")`
 - a) `window.geometry()`
 - b) `window.title ()`
 - c) `window.mainloop()`
 - d) `window.load()`
- 8. Виджет, позволяющий пользователю ввести одну строку текста.
 - a) `Frame`
 - b) `Entry`
 - c) `Label`
 - d) `Button`
- 9. Виджет, который позволяет пользователю ввести любое количество текста
 - a) `Text+`
 - b) `Entry`
 - c) `Label`
 - d) `Button`
- 10. Какой командой модуля `Turtle` можно задать цвет пера?
 - g) `color = 'цвет'`
 - h) `color ('цвет')`
 - i) `color.цвет='цвет'`

Тема 5.5. Этапы разработки приложений

- 1. Диалоговое окно для информирования пользователя:
 - a) `showinfo():`
 - b) `showerror():`
 - c) `showwarning():`
- 2. Диалоговое окон для сообщения об ошибке:
 - a) `showinfo():`
 - b) `showerror():`
 - c) `showwarning():`
- 3. Диалоговое окон для отображения предупреждений:
 - a) `showinfo():`
 - b) `showerror():`
 - c) `showwarning():`
- 4. Какую нужно написать команду, чтобы установить размеры окна?
 - a) `win = tkinter.Tk ()`
 - b) `win.title ()`
 - c) `win.geometry ()`
 - d) `win.mainloop ()`
- 5. Выберите команду заголовка окна.
 - a) `win = tkinter.Tk ()`

- b) win.title () +
- c) win.geometry ()
- d) win.mainloop ()

6. Какой класс Tkinter соответствует виджету для поля ввода?

- e) Label
- f) Text
- g) Frame
- h) Entry

7. Какой класс Tkinter соответствует виджету для надписи?

- e) Label
- f) Text
- g) Frame
- h) Entry

8. Какой класс Tkinter соответствует виджету для вывода графических примитивов?

- e) Label
- f) Text
- g) Frame
- h) Canvas

9. Какой метод создает текст в Tkinter?

- a. create_text
- b. create_image
- c. create_rectangle
- d. create_line

10. Какой метод создает прямоугольник в Tkinter?

- a) create_text
- b) create_image
- c) create_rectangle
- d) create_line

Тема 5.6. Иерархия классов.

1. Для создания класса в Python используются Ключевое слово:

- a) class
- b) print()
- c) int
- d) str

2. Создаем атрибуты класса:

- a) class
- b) name = "Aliev"
- c) print()

d) def stop(): 3. Создаем методы класса:

- a) class
- b) name = "Aliev"
- c) button
- d) def stop():

4. Виджет в графическом приложении:

- a) class
 - b) button
 - c) print()
 - d) def stop():
5. Вывод результата:
- a) class
 - b) button
 - c) print()
 - d) def stop():
6. Определите имя класса: class Person:
- ```
def say_hello(self):print("Hello")
tom = Person()tom.say_hello()
```
7. Определите метод: class Person:
- ```
def say_hello(self):print("Hello")
tom = Person()tom.say_hello()
```
8. Определите объект класса: class Person:
- ```
def say_hello(self):print("Hello")
tom = Person()tom.say_hello()
```
9. Определите атрибуты объекта: class Person:
- ```
company = "Microsoft"print(tom.company)
```
10. Определите объект класса: class Person:
- ```
def say(self, message):print(message)
tom = Person() tom.say("Hello ")
```

**Критерии оценки:**

| № п/п | Наименование оценочного средства | Критерии оценивания на «неудовлетв-но» | Критерии оценивания на «удовлетв-но» | Критерии оценивания на «хорошо»    | Критерии оценивания на «отлично» |
|-------|----------------------------------|----------------------------------------|--------------------------------------|------------------------------------|----------------------------------|
| 1     | Тестирование                     | 0% -50%<br>правильных<br>ответов       | 51% - 64%<br>правильных<br>ответов   | 65% - 84%<br>правильных<br>ответов | 85% - 100%<br>правильных ответов |

## ПЕРЕЧЕНЬ ЛАБОРАТОРНЫХ РАБОТ И ПРАКТИЧЕСКИХ ЗАНЯТИЙ

1. Практическое занятие №1. «Знакомство со средой программирования»
2. Практическое занятие №2. «Составление программ линейной структуры»
3. Практическое занятие №3. «Составление программ разветвляющейся структуры»
4. Практическое занятие №4. «Составление программ циклической структуры»
5. Практическое занятие №5. «Составление программ усложненной структуры»
6. Практическое занятие №6. «Разработка алгоритма для конкретной задачи»
7. Практическое занятие №7. «Определение сложности работы алгоритмов»
8. Практическое занятие №8. «Графическое отображение алгоритмов»
9. Практическое занятие №9. «Обработка одномерных массивов»
10. Практическое занятие №10. «Обработка двумерных массивов»
11. Практическое занятие №11. «Работа со строками»
12. Практическое занятие №12. «Работа с данными типа множество»
13. Практическое занятие №13. «Работа с файлами последовательного доступа»
14. Практическое занятие №14. «Работа с типизированными файлами»
15. Практическое занятие №15. «Работа с нетипизированными файлами»
16. Практическое занятие №16. «Организация процедур»
17. Практическое занятие №17. «Организация функций»
18. Практическое занятие №18. «Применение рекурсивных функций»
19. Практическое занятие №19. «Программирование модуля»
20. Практическое занятие №20. «Создание библиотеки подпрограмм»
21. Практическое занятие №21. «Использование указателей для организации связанных списков»
22. Практическое занятие №22. «Изучение интегрированной Среды разработчика»
23. Практическое занятие №23. «Создание проекта с использованием компонентов для работы с текстом»
24. Практическое занятие №24. «Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени»
25. Практическое занятие №25. «События компонентов (элементов управления)»
26. Практическое занятие №26. «Оформление кода программы в соответствии со стандартом кодирования»
27. Практическое занятие №27. «Выполнение проверки кода программы»
28. Практическое занятие №28. «Выполнение отладки кода программы»
29. Практическое занятие №29. «Создание процедур на основе событий»
30. Практическое занятие №30. «Создание проекта с использованием кнопочных компонентов»
31. Практическое занятие №31. «Создание проекта с использованием компонентов стандартных диалогов и системы меню»

32. **Практическое занятие №32.** «Разработка функциональной схемы работы приложения»
33. **Практическое занятие №33.** «Разработка оконного приложения с несколькими формами»
34. **Практическое занятие №34.** «Разработка игрового приложения»
35. **Практическое занятие №35.** «Создание процедур обработки событий»
36. **Практическое занятие №36.** «Компиляция и запуск приложения»
37. **Практическое занятие №37.** «Разработка интерфейса приложения»
38. **Практическое занятие №38.** «Программирование приложений»
39. **Практическое занятие №39.** «Объявления класса»
40. **Практическое занятие №40.** «Создание наследованного класса»
41. **Практическое занятие №41.** «Тестирование приложения»
42. **Практическое занятие №42.** «Отладка приложения»
43. **Практическое занятие №43.** «Перегрузка методов»

Выполнение заданий к практическим занятиям, ответы на контрольные вопросы к ним способствуют контролю **умений студентов** по учебной дисциплине ОП.04 Основы алгоритмизации и программирования.

Цели, задачи, задания, порядок проведения, контрольные вопросы, а также критерии оценки лабораторных работ представлены в методических указаниях к выполнению ЛПЗ по учебной дисциплине.

### **ВОПРОСЫ ДЛЯ УСТНОГО (ПИСЬМЕННОГО ОПРОСА)**

1. Основные понятия алгоритмизации.
2. Способы описания алгоритмов.
3. Свойства алгоритмов.
4. Линейные алгоритмы.
5. Алгоритмы с ветвлениями.
6. Циклические алгоритмы.
7. Базовые конструкции структурного программирования.
8. Цикл.
9. Цикл с предусловием.
10. Цикл с постусловием
11. Компилятор
12. Интерпретатор

13. Структурное программирование
14. Принципы структурного программирования.
15. Переменные.
16. Операция присваивания.
17. Типы данных.
18. Арифметические операции.
19. Оператор присваивания.
20. Символьный тип данных.
21. Конструкция последовательность
22. Конструкция ветвление
23. Конструкция цикл
24. Оператор выбора.
25. Конструкция if...else. Синтаксис конструкции.
26. Цикл. For. Синтаксис конструкции.
27. Цикл с предусловием (цикл пока).
28. Цикл с пост условием (цикл до).
29. Оператор.
30. Выражение.
31. Цикл с параметром.

**Устный опрос** – это средство контроля, организованное как специальная беседа преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний, обучающегося по определенному разделу, теме, проблеме и т. п.

***Критерии оценки устного опроса студентов:***

Оценка «отлично»:

- глубокое и прочное усвоение материала темы или раздела;
  - полные, последовательные, грамотные, логически излагаемые аргументированные ответы;
  - демонстрация обучающимся знаний в объеме пройденной программы и дополнительно рекомендованной литературы;
  - воспроизведение учебного материала с требуемой степенью точности.
- Оценка «хорошо»:

- наличие несущественных ошибок, не достаточно аргументированные ответы на вопросы;
- демонстрация обучающимся знаний в объеме пройденной программы;

- четкое изложение учебного материала. Оценка «удовлетворительно»:
  - наличие несущественных ошибок в ответе, отсутствие аргументации, но достаточно грамотное и логичное изложение;
  - демонстрация обучающимся недостаточно полных знаний по пройденной программе, отсутствие аргументации;
  - не структурированное, не грамотное и не логичное изложение учебного материала при ответе.

Оценка «неудовлетворительно»:

- незнание материала темы или раздела;
- серьезные ошибки при ответе.

### **ПЕРЕЧЕНЬ ВОПРОСОВ К ЭКЗАМЕНУ**

Промежуточная аттестация по учебной дисциплине ОП.04. «Основы алгоритмизации и программирования» проводится в форме экзамена.

#### **Перечень вопросов для проведения экзамена**

1. Развитие языков программирования. Обзор языков программирования.
2. Области применения языков программирования.
3. Стандарты языков программирования. Среда проектирования.
4. Компиляторы и интерпретаторы.
5. Жизненный цикл программы.
6. Программа. Программный продукт и его характеристики
7. Типы данных.
8. Простые типы данных.
9. Производные типы данных.
10. Структурированные типы данных.
11. Операции и выражения.
12. Правила формирования и вычисления выражений.
13. Структура программы. Ввод и вывод данных.
14. Оператор присваивания.
15. Составной оператор.
16. Условный оператор.
17. Оператор выбора.

18. Цикл с постусловием.
19. Цикл с предусловием.
20. Цикл с параметром.
21. Вложенные циклы.
22. Массивы. Двумерные массивы.
23. Строки. Стандартные процедуры и функции для работы со строками
24. Структурированный тип данных – множество.
25. Операции над множествами.
26. Комбинированный тип данных – запись.
27. Файлы последовательного доступа. Файлы прямого доступа
28. Общие сведения о подпрограммах. Определение и вызов подпрограмм.
29. Область видимости и время жизни переменной.
30. Механизм передачи параметров.
31. Организация функций.
32. Рекурсия. Программирование рекурсивных алгоритмов
33. Основы структурного программирования. Методы структурного программирования
34. Модульное программирование. Понятие модуля.
35. Структура модуля. Компиляция и компоновка программы. Стандартные модули
36. Указатели. Описание указателей.
37. Основные понятия и применение динамически распределяемой памяти.
38. Создание и удаление динамических переменных.
39. Структуры данных на основе указателей. Задача о стеке.
40. История развития объектно-ориентированного программирования (ООП).
41. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.
42. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.
43. Классы объектов. Компоненты и их свойства.
44. Событийно-управляемая модель программирования. Компонентно-ориентированный подход
45. Требования к аппаратным и программным средствам интегрированной среды разработчика.
46. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты.
47. Форма и размещение на ней управляющих элементов. Панель компонентов и их свойства. Окно кода проекта.
48. Состав и характеристика проекта. Выполнение проекта.

49. Настройка среды и параметров проекта. Панель компонентов и их свойства.
50. Окно кода проекта. Состав и характеристика проекта.
51. Выполнение проекта. Настройка среды и параметров проекта.
52. Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение. Дополнительные элементы управления.
53. Свойства компонентов. Виды свойств.
54. Синтаксис определения свойств. Назначения свойств и их влияние на результат.
55. Управление объектом через свойства. События компонентов (элементов управления), их сущность и назначение.
56. Создание процедур на основе событий.
57. Разработка функционального интерфейса приложения.
58. Создание интерфейса приложения. Разработка функциональной схемы работы приложения. Разработка игрового приложения
59. Разработка приложения. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя.
60. Тестирование, отладка приложения
61. Классы ООП: виды, назначение, свойства, методы, события.
62. Перегрузка методов. Тестирование и отладка приложения.

#### Критерии оценки:

| № п/п | Наименование оценочного средства | Критерии оценивания на «неудовлетворительно»                                                                                                                                       | Критерии оценивания на «удовлетворительно»                                                                                                                                                        | Критерии оценивания на «хорошо»                                                                                                                                                                           | Критерии оценивания на «отлично»                                                                                                                                                                           |
|-------|----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1     | Экзамен                          | выставляется студенту, обнаружившему пробелы в знаниях основного учебного материала по дисциплине. При ответе обнаружено непонимание студентом основного содержания теоретического | выставляется студенту, когда студент обнаруживает знание основного программного материала по дисциплине, но допускает погрешности в ответе. Ответ недостаточно логически выстроен, самостоятелен. | выставляется студенту, обнаруживает полное знание учебного материала, демонстрирует систематический характер знаний по дисциплине. Ответ полный и правильный, подтвержден примерами; но их обоснование не | выставляется студенту, когда студент обнаруживает систематическое и глубокое знание программного материала по дисциплине, умеет свободно ориентироваться в вопросе. Ответ полный и правильный на основании |

|  |  |                                                                                                                                                                                                                                                           |                                                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                         |
|--|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  |  | <p>материала или допущен ряд существенных ошибок, которые студент не может исправить при наводящих вопросах экзаменатора, затрудняется в ответах на вопросы. Ответ носит поверхностный характер; наблюдаются неточности в использовании терминологии.</p> | <p>Основные понятия употреблены правильно, но обнаруживается недостаточное раскрытие теоретического материала. Выдвигаемые положения недостаточно аргументированы и не подтверждены примерами; ответ носит преимущественно описательный характер. Студент испытывает достаточные трудности в ответах на вопросы. Терминология используется недостаточно.</p> | <p>аргументирован, отсутствует собственная точка зрения. Материал изложен в определенной логической последовательности, при этом допущены 2-3 несущественные погрешности, исправленные по требованию экзаменатора. Студент испытывает незначительные трудности в ответах на дополнительные вопросы. Материал изложен осознанно, самостоятельно, с использованием современных научных терминов, литературным языком.</p> | <p>изученного материала. Выдвинутые положения аргументированы и иллюстрированы примерами. Материал изложен в определённой логической последовательности, осознанно, литературным языком, с использованием современных научных терминов; ответ самостоятельный. Студент уверенно отвечает на дополнительные вопросы.</p> |
|--|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

ДОКУМЕНТ ПОДПИСАН  
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 133600552358087161194895262509558337786447861787

Владелец Гайдаров Насир Алиевич

Действителен с 22.03.2024 по 22.03.2025