

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
**«Ухтинский государственный технический университет»**  
**(УГТУ)**  
**филиал Ухтинского государственного технического университета**  
**в г. Усинске**  
**(УФ УГТУ)**  
(среднего профессионального образования)

УТВЕРЖДАЮ  
И. о. директора филиала  
\_\_\_\_\_ О. В. Филиппова  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

**КОМПЛЕКС ОЦЕНОЧНЫХ СРЕДСТВ**  
**ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**  
**МДК 02.01. Технология разработки программного обеспечения**  
**образовательной программы**  
**среднего профессионального образования**  
**По специальности 09.02.11 Разработка и управление программным**  
**обеспечением**

г. Усинск  
2026

# 1. ПАСПОРТ КОМПЛЕКТА ОЦЕНОЧНЫХ СРЕДСТВ

## 1.1 Область применения

Комплект оценочных средств (далее – КОС) предназначен для контроля и оценки результатов прохождения учебной дисциплины МДК 02.01. «Технология разработки программного обеспечения», образовательной программы среднего профессионального образования (далее – ОП СПО) по специальности **09.02.11** «Разработка и управление программным обеспечением».

## 1.2 Результаты освоения компетенции

В результате проведения промежуточной аттестации по дисциплине осуществляется комплексная оценка овладения следующими профессиональными и общими компетенциями:

| Код           | Результат освоения компетенции                                                                                                                                                                                                      |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>ПК 2.1</b> | Проектировать модули программного обеспечения.                                                                                                                                                                                      |
| <b>ПК 2.2</b> | Разрабатывать модули программного обеспечения.                                                                                                                                                                                      |
| <b>ПК 2.3</b> | Выполнять интеграцию модулей и компонентов программного обеспечения.                                                                                                                                                                |
| <b>ПК 2.4</b> | Выполнять тестирование и отладку программного обеспечения                                                                                                                                                                           |
| <b>ПК 2.5</b> | Осуществлять документирование программных модулей программного обеспечения.                                                                                                                                                         |
| <b>ОК 2</b>   | Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности                                                                         |
| <b>ОК 3</b>   | Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях |
| <b>ОК 4</b>   | Эффективно взаимодействовать и работать в коллективе и команде                                                                                                                                                                      |
| <b>ОК 5</b>   | Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста                                                                               |
| <b>ОК 7</b>   | Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях                                             |
| <b>ОК 9</b>   | Пользоваться профессиональной документацией на государственном и иностранном языках                                                                                                                                                 |

## **2. ФОРМЫ КОНТРОЛЯ И ОЦЕНКИ РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ**

В соответствии с учебным планом и рабочей программой предусматривается текущий и промежуточный контроль результатов освоения.

Предметом оценки служат умения и знания, предусмотренные требованиями ФГОС по дисциплине МДК 02.01. «Технология разработки программного обеспечения», направленные на формирование общих и профессиональных компетенций, а также личностных результатов в рамках программы воспитания.

### **2.1 Критерии и нормы оценки знаний обучающихся.**

**Оценка «зачтено»** выставляется, если студент твёрдо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми знаниями, умениями и навыками.

**Оценка «не зачтено»** выставляется, если студент не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы.

**Также при оценке зачёта могут учитываться:**

- Степень усвоения теоретического материала. Оценивается по правильности ответа на теоретический вопрос, последовательности, связанности и чёткости изложения материала, объёму теоретических знаний в рамках программного материала, умению выделить главные положения в изученном материале, культуре речи.
- Уровень сформированности умений и навыков. Оценивается по правильности реализации алгоритма решения практической задачи, правильности интерпретации полученных результатов, умению сделать выводы из полученных значений.

### **2.2 Критерии оценки за тестирование**

Оценка 5 «отлично» (85 % -100 %)

Оценка 4 «хорошо» (75 % - 84 %)

Оценка 3 «удовлетворительно» (55 % -74 %)

Оценка 2 «неудовлетворительно» (менее 54 %)

## 2.3 Формы и методы оценивания учебной дисциплины «Технология разработки программного обеспечения»

| Раздел, тема                                                                                     | Формы и методы контроля |                                    |                          |                                   |                     |                         |
|--------------------------------------------------------------------------------------------------|-------------------------|------------------------------------|--------------------------|-----------------------------------|---------------------|-------------------------|
|                                                                                                  | Текущий контроль        |                                    | Промежуточная аттестация |                                   | Итоговая аттестация |                         |
|                                                                                                  | Формы контроля          | Проверяемые Знания, умения, ОК, ЛР | Формы контроля           | Проверяемые знания, умения ОК, ЛР | Формы контроля      | Проверяемые ЗУН, ОК, ЛР |
| <b>Часть 1. Методы и этапы технологии программирования.</b>                                      |                         |                                    |                          |                                   |                     |                         |
| Тема 1. Эволюция технологии программирования. Основы программирования.                           | УО, ПР №1               | ПК2.1, ПК2.4, ПК2.5.               | ДФК, Т                   | ПК2.1, ПК2.4, ПК2.5.              | Эк                  | ПК2.1, ПК2.4, ПК2.5.    |
| Тема 2. Основные этапы технологии программирования. Основы программирования – условный оператор. | УО, ПР №2.              | ПК2.1, ПК2.4, ПК2.5.               | ДФК, Т                   | ПК2.1, ПК2.4, ПК2.5.              | Эк                  | ПК2.1, ПК2.4, ПК2.5.    |
| Тема 3. Пользовательский интерфейс. Основы программирования – типы данных.                       | УО, ПР №3, СР           | ПК2.1, ПК2.4, ПК2.5.               |                          | ПК2.1, ПК2.4, ПК2.5.              | Эк                  | ПК2.1, ПК2.4, ПК2.5.    |
| <b>Часть 2. Программирование на языке высокого уровня.</b>                                       |                         |                                    |                          |                                   |                     |                         |
| Тема 4. Программирование на языке высокого уровня Python.                                        | ПР № 4, СР.             | ПК2.1, ПК2.4, ПК2.5.               |                          | ПК2.1, ПК2.4, ПК2.5.              | Эк                  | ПК2.1, ПК2.4, ПК2.5.    |

### Кодификатор оценочных средств

| Функциональный признак оценочного средства (тип контрольного задания) | Код оценочного средства |
|-----------------------------------------------------------------------|-------------------------|
| Устный опрос                                                          | УО                      |
| Практическая работа                                                   | ПР №                    |
| Самостоятельная работа                                                | СР                      |
| Тестирование                                                          | Т                       |
| Другая форма контроля                                                 | ДФК                     |
| Экзамен                                                               | Эк                      |

### 3. Контрольно-оценочные материалы для промежуточной и итоговой аттестации по учебной дисциплине «Технология разработки ПО»

#### 3.1 Типовые задания для проведения текущего контроль

##### 3.1.1 Перечень практических работ.

###### Практическая работа № 1.

**Задача № 1.** Напишите программу, которая считает стоимость трёх компьютеров, состоящих из монитора, системного блока, клавиатуры и мыши. На вход программе подаются четыре целых числа (каждое на отдельной строке): стоимость монитора, стоимость системного блока, стоимость клавиатуры, стоимость мыши. Язык программирования: **Python**.

**Решение:**

```
a, b, c, d=int(input()), int(input()), int(input()), int(input())
print(3*(a+b+c+d))
```

###### Практическая работа № 2.

**Задача № 1.** Напишите программу, которая определяет наименьшее из двух чисел. На вход программе подаются два различных целых числа. Программа должна вывести наименьшее из двух чисел. Язык программирования: **Python**.

**Решение:**

```
n1, n2 = int(input()), int(input())
if n1<n2:
    print(n1)
else:
    print(n2)
```

###### Практическая работа № 3.

**Задача № 1.** Дано положительное действительное число. Выведите его дробную часть. На вход программе подаётся положительное действительное число. Программа должна вывести дробную часть числа. Язык программирования: **Python**.

**Решение:**

```
n= float(input())
n=n-int(n)
print(n)
```

**Задача № 2.** Напишите программу, определяющую площадь круга и длину окружности по заданному радиусу R. Программа должна вывести два числа (каждое на отдельной строке) – площадь круга и длину окружности радиуса R. Язык программирования: **Python**.

**Решение:**

```
from math import pi
r=float(input())
s=pi*r**2
c=2*pi*r
print(s, c, sep="\n")
```

###### Практическая работа № 4.

**Задача № 1.** Даны два целых числа m и n ( $m > n$ ). Напишите программу, которая выводит все нечётные целые числа от m до n (включительно) в порядке убывания. Программа должна вывести числа, каждое на отдельной строке. Язык программирования: **Python**.

**Решение:**

```
m, n = int(input()), int(input())
```

```

for i in range(m, n-1, -1):
    if i%2!=0:
        print(i)

```

**Задача № 2.** На вход программе подаётся натуральное число  $n$ . Напишите программу, которая вычисляет сумму всех его делителей. Язык программирования: **Python**.

**Решение:**

```

n = int(input())
sm=0
for i in range(1, n+1):
    if n%i==0:
        sm+=i
print(sm)

```

**Задача № 3.** На вход программе подаётся натуральное число  $n$ . Напишите программу, которая вычисляет  $n!$ . Язык программирования: **Python**.

**Решение:**

```

n = int(input())
k=1
for i in range(2, n+1):
    k*=i
print(k)

```

**Задача № 4.** На вход программе подаётся последовательность целых чисел от 11 до 55, характеризующее оценку ученика, каждое число на отдельной строке. Концом последовательности является любое неположительное число либо число, большее 55. Напишите программу, которая выводит количество пятерок. Язык программирования: **Python**.

**Решение:**

```

n = int(input())
kl=0
while 0 < n < 6:
    if n==5:
        kl+=1
    n = int(input())
print(kl)

```

**Задача № 5.** Напишите функцию `get_circle(radius)`, которая принимает в качестве аргумента радиус окружности и возвращает два значения: длину окружности и площадь круга, ограниченного данной окружностью. Язык программирования: **Python**.

**Решение:**

```

import math

# объявление функции
def get_circle(radius):
    c = 2 * math.pi * radius
    s = math.pi * (radius ** 2)
    return c, s

# считываем данные
r = float(input())

# вызываем функцию
length, square = get_circle(r)
print(length, square)

```

### 3.1.2. Примерные вопросы для устного опроса обучающихся по учебной дисциплине

#### Тема 1. Эволюция технологии программирования

- 1.1. Какие этапы эволюции прошли технологии программирования?
- 1.2. Какие языки и методы программирования вы знаете?
- 1.3. Какие языки программирования называются языками высокого уровня?
- 1.4. Какая модель построения программ лежит в основе технологии процедурного программирования?
- 1.5. В чем преимущества и недостатки языков сценария?
- 1.6. Какова область применения языков параллельных вычислений?

#### Тема 2. Основные этапы технологии программирования

- 2.1. Каковы основные этапы решения задач на ЭВМ?
- 2.2. Что такое жизненный цикл программного обеспечения?
- 2.3. Что такое спецификации, какие сведения они содержат?
- 2.4. Что представляет собой структурный анализ?
- 2.5. В чем заключается этап реализации программного обеспечения?
- 2.6. Какие методы оценки трудоемкости разработки программного обеспечения вы знаете?
- 2.7. Какие способы записи алгоритма вы знаете?
- 2.8. Какие работы выполняются в процессе обеспечения качества программного продукта?
- 2.9. Какие основные группы ошибок в программных продуктах вы знаете?
- 2.10. Какие стадии тестирования ПО вы знаете?
- 2.11. Каковы основные методы отладки?
- 2.12. Что подразумевается под защитным программированием?
- 2.13. Каковы основные группы документации программного обеспечения?
- 2.14. Какие существуют формы записи алгоритма?
- 2.15. По каким критериям оценивается качество программы?
- 2.16. Какие факторы влияют на качество программ?
- 2.17. Что представляет собой инструментарий технологии программирования?

#### Тема 3. Пользовательский интерфейс

- 3.1. Каковы основные типы пользовательских интерфейсов?
- 3.2. В чем преимущество интерфейса со свободной навигацией по сравнению с интерфейс-меню?
- 3.3. Какие интерфейсы называются графическими?
- 3.4. Какие интерфейсы используются при объектно-ориентированном подходе к программированию?
- 3.5. Что такое диалог?
- 3.6. Какие элементы пользовательских интерфейсов относятся к интеллектуальным?

### 3.1.3 Перечень самостоятельных работ.

**Задача № 1.** Напишите программу, которая определяет, является ли год с данным номером високосным. Если год является високосным, то выведите «YES» (без кавычек), иначе выведите «NO» (без кавычек). Год является високосным, если его номер кратен 4, но не кратен 100, или если он кратен 400. Язык программирования: **Python**.

**Решение:**

```
g=int(input())
if g%4==0 and g%100!=0 or g%400==0:
    print("YES")
else:
    print("NO")
```

**Задача № 2.** Напишите программу, которая считывает с клавиатуры два целых числа и строку. Если эта строка является обозначением одной из четырёх математических операций (+, -, \*, /), то выведите результат применения этой операции к введённым ранее числам, в противном случае выведите «Неверная операция» (без кавычек). Если пользователь захочет поделить на ноль, выведите текст «На ноль делить нельзя!» (без кавычек). Язык программирования: **Python**.

**Решение:**

```
a= int(input())
b= int(input())
oper=input()
if oper=="-":
    print(a-b)
elif oper=="+":
    print(a+b)
elif oper=="*":
    print(a*b)
elif oper=="/":
    if b==0:
        print("На ноль делить нельзя!")
    else:
        print(a/b)
else:
    print("Неверная операция")
```

### **3.2. Типовые задания для проведения промежуточной аттестации**

#### **Часть 1. Методы и этапы технологии программирования.**

##### **Тема 1. «Эволюция технологии программирования»**

**Выберите один правильный ответ**

- 1) Как называется совокупность методов и средств, используемых в процессе разработки программного обеспечения?**
  - a) структура программирования
  - b) условия программирования
  - c) технология программирования**
  - d) операции программирования
- 2) Что позволило программистам вместо кодов использовать мнемонические обозначения кодов операций и символические имена данных?**
  - a) разработка аппаратных средств
  - b) появление ассемблеров**
  - c) создание совместной разработки ПО
  - d) возникновение библиотеки подпрограмм
- 3) Как называется совокупность объектов, имеющих одинаковые свойства и методы?**
  - a) класс**
  - b) вид
  - c) атрибут
- 4) Какое программирование возникло как упрощение функционального программирования для математиков и лингвистов, решающих задачи символьной обработки?**
  - a) логическое**
  - b) многослойное
  - c) ориентированное
  - d) наследственное

##### **Тема 2. «Основные этапы технологии программирования»**

- 5) Какая модель жизненного цикла программного обеспечения предполагает переход на следующий этап после завершения всех операций предыдущего этапа?**
  - a) ядерная модель
  - b) спиральная модель
  - c) модель с промежуточным контролем
  - d) каскадная модель**
- 6) Как называется совокупность взаимосвязанных программ, совместно решающих небольшой класс задач некоторой прикладной области?**
  - a) программные комплексы**
  - b) программные системы
  - c) архитектура программ

- 7) Как именуется процесс формулировки назначения программного обеспечения и основных требований к нему?
- a) постановка задачи
  - b) реализация
  - c) тестирование

- 8) Как называется процесс поиска и исправления ошибок, обнаруженных при тестировании программы? (в поле ответа введите одно слово)

Ответ: отладка

## Часть 2. Основы программирования.

- 9) Какая команда используется для вывода (печати) данных (язык программирования Python) :

- a) `print()`
- b) `cin()`
- c) `Console.WriteLine()`

- 10) Что покажет приведённый ниже код (язык программирования Python)?

```
print('Hello, it's me!')
```

- a) Hello, it's me!
- b) 'Hello, it's me!'
- c) произойдёт ошибка

- 11) Какая команда используется для считывания данных с клавиатуры (язык программирования Python):

- a) `scanf()`
- b) `cin`
- c) `Console.ReadLine()`
- d) `input()`

- 12) Какая из указанных строк считывает целое число в переменную n (язык программирования Python)?

- a) `n = integer(input())`
- b) `n = int(input())`
- c) `n = input()`

- 13) Укажите подряд цифры с верными утверждениями (язык программирования Python):

- (1) Имя переменной не может начинаться с цифры
- (2) Имя переменной не может оканчиваться цифрой
- (3) Имя переменной может начинаться с символа подчёркивания (`_`)
- (4) Имя переменной не должно совпадать с ключевым (зарезервированным) словом

- 14) Что покажет приведённый ниже код (язык программирования Python)?

```
a = 4  
print(a, 'a')
```

- a) 4 4
- b) a a
- c) 4 a
- d) a 4

- 15) Что покажет приведённый ниже код (язык программирования Python)?

```
print(3 ** 2 * 3 - 1)
```

- a) 56
- b) 47
- c) 26

**16) Вызов команды print() с пустыми скобками (язык программирования Python):**

- a) вставит один пробел
- b) выполнит перевод строки
- c) ничего не выведет

**17) Параметр sep в команде print() выполняет (язык программирования Python):**

- a) вставку символов в конец всех аргументов;
- b) разделение между аргументов;
- c) перевод строки.

**18) Укажите количество видов деления в Python:**

- a) 1
- b) 2
- c) 3
- d) 4

**19) Какой арифметических операторов имеет высший приоритет после скобок (язык программирования Python):**

- a) Деление и умножение
- b) Сумма и разность
- c) Возведение в степень

**20) Какой арифметический оператор выполняет целочисленное деление (язык программирования Python):**

- a) знак процента (%)
- b) слэш (/)
- c) двойной слэш (//)

**21) Какой арифметический оператор находит остаток от деления (язык программирования Python):**

- a) знак процента (%)
- b) слэш (/)
- c) двойной слэш (//)

**22) Какой логический оператор имеет наивысший приоритет (язык программирования Python):**

- a) логическое сложение or
- b) логическое отрицание not
- c) логическое умножение and

**23) Чтобы значение выражения с оператором and было истинным, должны быть (язык программирования Python):**

- a) Истинно хотя бы одно из выражений
- b) Истины все выражения
- c) Ни одно выражение не истинно

**24) Чтобы значение выражения с оператором or было истинным, должны быть (язык программирования Python)**

- a) Истинно хотя бы одно выражение
- b) Истины все выражения
- c) Ни одно выражение не истинно

**25) Верно ли указан оператор сравнения (язык программирования Python):**

```
if age >= 7 and <= 9:
```

- a) Верно
- b) Неверно**

## **3.2 Организация и проведение итоговой аттестации**

### **3.2.1 Условия проведения итоговой аттестации. Подготовка к проведению итоговой аттестации.**

Итоговая аттестация проводится в период экзаменационной сессии, установленной календарным графиком учебного процесса рабочего учебного плана. С формами проведения промежуточной аттестации обучающиеся знакомятся в течение двух месяцев с начала обучения.

Для подготовки к итоговой аттестации студентом (не позднее чем за 20 дней до проведения экзамена в соответствии с календарным графиком учебного процесса) выдается перечень тем к практическим заданиям, составленный исходя из требований ФГОС и рабочей программы предмета к уровню умений и знаний. Количество вопросов и практических задач в перечне для подготовки к промежуточной аттестации превышает количество вопросов и практических задач, необходимых для составления контрольно-измерительных материалов.

На основе разработанного и объявленного обучающимся перечня вопросов и практических задач, рекомендуемых для подготовки к итоговой аттестации, составлены задания, содержание которых до обучающихся не доводится. Вопросы и практические задачи носят равноценный характер. Формулировки вопросов четкие, краткие, понятные, исключают двойное толкование.

### **3.2.2 Проведение итоговой аттестации**

Студенты для сдачи итоговой аттестации распределяются по времени. На выполнение задания студенту отводится не более двух академических часов. Оценка, полученная на итоговой аттестации, заносится преподавателем в зачетную книжку студента (кроме неудовлетворительной) и экзаменационную ведомость (в том числе и неудовлетворительные). Экзаменационная оценка по предмету за данный семестр является определяющей, независимо от полученных в семестре оценок текущего контроля по предмету

**ВОПРОСЫ ДЛЯ ПОДГОТОВКИ К ЗАЧЕТУ  
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ  
МДК 02.01. Технология разработки программного обеспечения**

**Теоретические вопросы:**

**Часть 1. Методы и этапы технологии программирования.**

1. Какая модель построения программ лежит в основе технологии процедурного программирования? Каковы основные методы процедурного программирования?
2. На чем основывается концепция объектно-ориентированного программирования? Назовите основные принципы объектно-ориентированного программирования.
3. Что такое спецификации, какие сведения они содержат?
4. Какие диаграммы включает язык UML?
5. Что представляет собой структурный анализ? Какие правила лежат в основе структурного анализа?
6. Какая модель построения программы используется при объектно – ориентированном подходе?
7. В чем заключается этап реализации программного обеспечения?
8. Назовите методы оценки трудоемкости разработки программного обеспечения.
9. Назовите виды организации коллектива разработчиков программного обеспечения.

**Часть 2. Программирование на языке высокого уровня.**

1. Что выполняют команды `print()` и `input()`? Как влияет на команду `print()` параметры `sep` и `end`?
2. Условный оператор. Чем отличается вложенные от каскадных операторов.
3. Условный оператор. Какие логические операции возможно использовать в условном операторе.
4. Целочисленный тип данных и числа с плавающей точкой. Арифметические операторы.
5. Целочисленный тип данных и числа с плавающей точкой. Явное и неявное преобразование типов данных.
6. Строковый тип данных. Как определить длину строки. Конкатенация строк. Умножение строки на число.
7. Модуль `math`. Особенности подключения модулей. Основные функции и константы модуля `math`.
8. Цикл `for`. Какие параметры используются в функции `range()`.
9. Какое назначение имеют переменные: счетчик, сумматор, мультипликатор, Алгоритм обмена значений переменных.
10. Какое назначение имеют переменные: счетчик, сумматор, мультипликатор. Расширенные операторы присваивания.
11. Операторы `break`, `continue`. Блок `else` в циклах.
12. Вложенные циклы. Операторы `break` и `continue` во вложенных циклах.
13. Индексация строк. Итерирование строк.
14. Срезы строк. Параметра среза строк. Срез до конца, от начала. Отрицательные индексы в срезе.
15. Срезы строк. Параметра среза строк. Как выполнить изменение символа строки.
16. Строковый тип данных, основные методы конвертации регистра.
17. Строковый тип данных, основные методы поиска и замены.
18. Строковый тип данных, основные методы классификации символов.

19. Строковый тип данных, основные методы форматирования строк.
20. Сравнение строк единичной длины. Сравнение строк не единичной длины.
21. Списки. Создание и вывод списка. Встроенная функция list().
22. Основы работы со списками. Функция len(). Оператор in. Индексация.
23. Основы работы со списками. Срезы. Операция конкатенации и умножения на число. Встроенные функции sum(), min(), max().
24. Добавление элементов в список. Удаление элементов из списка.
25. Способы вывод элементов списка и строки.
26. Строковые методы split() и join().
27. Основные назначения методов списков: insert(), index(), remove(), pop(), reverse().
28. Основные назначения методов списков: count(), clear(), copy(), sort(), reverse().
29. Создание пользовательских функций без параметров.
30. Создание пользовательских функций с параметрами.

## Практические задания:

1. Напишите программу, которая принимает три положительных числа и определяет, существует ли невырожденный треугольник с такими сторонами. Треугольник существует, если выполняется неравенство треугольника:  $a+b > c$ ;  $a+c > b$ ;  $b+c > a$ , где  $a, b, c$  - стороны треугольника (**язык программирования Python**)/

**Решение:**

```
a, b, c, = int(input()), int(input()), int(input())
if a+b>c and b+c>a and c+a>b:
    print("YES")
else:
    print("NO")
```

2. Напишите программу, которая упорядочивает три числа в порядке убывания (**язык программирования Python**).

**Решение:**

```
a1, a2, a3 = int(input()), int(input()), int(input())
mx=max(a1, a2, a3)
mn=min(a1, a2, a3)
sr=a1+a2+a3-mn-mx
print(mx, sr, mn, sep="\n")
```

3. На вход программе подаётся натуральное число  $n$ . Напишите программу, которая вычисляет сумму всех его делителей (**язык программирования Python**).

**Решение:**

```
n = int(input())
sm=0
for i in range(1, n+1):
    if n%i==0:
        sm+=i
print(sm)
```

4. На вход программе подаётся натуральное число  $n$ . Напишите программу, которая для каждого из чисел от 0 до  $n$  (включительно) выводит текст в следующем формате: Квадрат числа <текущее число> равен <квадрат текущего числа> (**язык программирования Python**).

**Решение:**

```
n= int(input())
for i in range(n+1):
    print("Квадрат числа", i, "равен", i*i)
```

5. На вход программе подаётся последовательность слов, каждое слово на отдельной строке. Концом последовательности является слово «КОНЕЦ» (без кавычек). При этом само слово «КОНЕЦ» не входит в последовательность, лишь символизируя её окончание. Напишите программу, которая выводит члены данной последовательности.

**Решение:**

```
ln = input()
while ln!="КОНЕЦ":
    print(ln)
    ln = input()
```

6. На вход программе подаётся число  $n$  ( $n>1$ ). Напишите программу, которая выводит его наименьший отличный от 1 делитель (**язык программирования Python**).

**Решение:**

```
n = int(input())
for i in range(2, n+1):
    if n%i==0:
```

```
print(i)
break
```

7. Даны два целых числа  $m$  и  $n$  ( $m \leq n$ ). Напишите программу, которая выводит все целые числа от  $m$  до  $n$  включительно (**язык программирования Python**).

**Решение:**

```
m, n = int(input()), int(input())
for i in range(m, n+1):
    print(i)
```

8. На вход программе подаются два целых числа  $a$  и  $b$  ( $a \leq b$ ). Напишите программу, которая подсчитывает количество чисел в диапазоне от  $a$  до  $b$  (включительно), куб которых оканчивается на 4 или 9 (**язык программирования Python**).

**Решение:**

```
a, b = int(input()), int(input())
b+=1
k=0
for i in range(a, b):
    if i**3%10==4 or i**3%10==9:
        k+=1
print(k)
```

9. На вход программе подаётся натуральное число  $n$ . Напишите программу, которая подсчитывает сумму тех чисел от 1 до  $n$  (включительно), квадрат которых оканчивается на 2, 5 или 8 (**язык программирования Python**).

**Решение:**

```
n=int(input())
sm=0
for i in range(1,n+1):
    if i**2%10==2 or i**2%10==5 or i**2%10==8:
        sm+=i

print(sm)
```

10. Напишите программу, которая считывает 8 чисел и выводит произведение отличных от нуля чисел (**язык программирования Python**).

**Решение:**

```
pr=1
for i in range(8):
    n = int(input())
    if n!=0:
        pr*=n
print(pr)
```

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
**«Ухтинский государственный технический университет»**  
**(УГТУ)**  
филиал Ухтинского государственного технического университета  
в г. Усинске  
**(УФ УГТУ)**  
(среднего профессионального образования)

УТВЕРЖДАЮ

И. о. директора филиала  
**О.В. Филиппова**

« 24 » 20 26 г.



«    » 20    г.

(подпись) (И. О. Фамилия)

«    » 20    г.

## РАБОЧАЯ ПРОГРАММА

Дисциплина:            Технология разработки программного обеспечения

Индекс:                МДК 02.01.

Специальность:        09.02.11 Разработка и управление ПО

Форма обучения:      очная

Курс (ы)                2

Семестр (ы):            4

г. Усинск  
2026

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта среднего общего образования, утвержденного приказом Минпросвещения России от 24.02.2025 N 138 "Об утверждении федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением.

Разработчик Капембва М.А. преподаватель УФ УГТУ (СПО).

Рассмотрено на заседании ученого совета УФ УГТУ (протокол от 26.05.2026 № 06).

СОГЛАСОВАНО

Заместитель директора УФ УГТУ  
по учебной работе

О.В. Филиппова