

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.02.03 Моделирование и анализ 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.02.03 «Моделирование и анализ 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
ПК 2.1	Проектировать модули программного обеспечения.
ПК 2.2	Разрабатывать модули программного обеспечения.
ПК 2.3	Выполнять интеграцию модулей и компонентов программного обеспечения.
ПК 2.4	Выполнять тестирование и отладку программного обеспечения
ПК 2.5	Осуществлять документирование программных модулей программного обеспечения.
ОК 1	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
ОК 2	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ОК 3	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях
ОК 4	Эффективно взаимодействовать и работать в коллективе и команде
ОК 5	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста

ОК 7	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях
ОК 9	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.02.03 «Моделирование и анализ 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 «Моделирование и анализ ПО»

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
Тема 1. Задачи и методы моделирования и анализа программных продуктов	ПР №1, УО, СР	ПКЗ.1, ПКЗ.3, ПКЗ.4.			ДФК, Т	ПКЗ.1, ПКЗ.3, ПКЗ.4.
Тема 2. Организация ревьюирования. Инструментальные средства ревьюирования	ПР №2, УО, СР	ПКЗ.1, ПКЗ.3, ПКЗ.4.			ДФК, Т	ПКЗ.1, ПКЗ.3, ПКЗ.4.

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 «Моделирование и анализ ПО»

3.1 Типовые задания для проведения текущего контроль

3.1.1 Перечень практических работ.

Практическая работа № 1.

Задача № 1. На обработку поступает последовательность из 1010 целых чисел (каждое на отдельной строке). Нужно написать программу, которая выводит на экран количество неотрицательных чисел последовательности и их произведение. Если неотрицательных чисел нет, требуется вывести на экран «NO» (без кавычек). Программист торопился и написал программу неправильно.

Найдите все ошибки в этой программе (их ровно 4). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

При необходимости вы можете добавить необходимые строки кода.

```
count = 0
p = 0
for i in range(1, 10):
    x = int(input())
    if x > 0:
        p = p * x
        count = count + 1
if count > 0:
    print(x)
    print(p)
else:
    print('NO')
```

Решение:

```
count = 0
p = 1
for i in range(10):
    x = int(input())
    if x >= 0:
        p = p * x
        count = count + 1
if count > 0:
    print(count)
    print(p)
else:
    print('NO')
```

Задача № 2. На обработку поступает последовательность из 1010 целых чисел (каждое на отдельной строке). Известно, что вводимые числа по абсолютной величине не превышают 106106. Нужно написать программу, которая выводит на экран сумму всех отрицательных чисел последовательности и максимальное отрицательное число в последовательности. Если отрицательных чисел нет, требуется вывести на экран «NO» (без кавычек). Программист торопился и написал программу неправильно.

Найдите все ошибки в этой программе (их ровно 5). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

При необходимости вы можете добавить необходимые строки кода.

```
mx = 0
s = 0
```

```

for i in range(11):
    x = int(input())
    if x < 0:
        s = x
    if x > mx:
        mx = x
print(s)
print(mx)

```

Решение:

```

mx = -1000000000000000
sm = 0
kl=0
for i in range(10):
    x = int(input())
    if x < 0:
        sm += x
        kl+=1
    if x > mx:
        mx = x
if kl>0:
    print(sm)
    print(mx)
else:
    print("NO")

```

Практическая работа № 2.

Задача № 1. На обработку поступает последовательность из 7 целых чисел (каждое на отдельной строке). Нужно написать программу, которая подсчитывает и выводит сумму всех чётных чисел последовательности. Если таких чисел нет, программа должна вывести 0. Программист торопился и написал программу неправильно.

Найдите все ошибки в этой программе (их ровно 4). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. При необходимости вы можете добавить необходимые строки кода.

```

s = 1
for i in range(1, 7):
    n = input()
    if i % 2 == 0:
        s = s + n
print(s)

```

Решение:

```

sm = 0
for i in range(7):
    n = int(input())
    if n % 2 == 0:
        sm += n
print(sm)

```

Задача № 2. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран максимальную цифру числа, кратную 3. Если в числе нет цифр, кратных 3, требуется на экран вывести «NO» (без кавычек). Программист торопился и написал программу неправильно.

Найдите все ошибки в этой программе (их ровно 5). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

Число 0 делится на любое натуральное число.

```

n = int(input())
max_digit = n % 10

```

```

while n > 0:
    digit = n % 10
    if digit % 3 == 0:
        if digit < max_digit:
            digit = max_digit
    n = n // 10
if max_digit == 0:
    print('NO')
else:
    print(max_digit)

Решение:
n = int(input())
mx = -1
while n != 0:
    digit = n % 10
    if digit % 3 == 0 and digit >= mx:
        mx = digit
    n = n // 10
if mx == -1:
    print('NO')
else:
    print(mx)

```

3.1.2. Примерные вопросы для устного опроса обучающихся по учебной дисциплине

Тема 1. Задачи и методы моделирования и анализа программных продуктов.

- 1.1. Что такое ЕСПД?
- 1.2. Назовите стадии разработки программных продуктов? Назовите этапы работ на каждой стадии
- 1.3. Назовите стадии разработки программных документов
- 1.4. Понятия управления контроля версиями, применение контроля версий.
- 1.5. Какие настройки могут быть установлены для среды Visual Studio, влияющие на особенности командной работы?
- 1.6. Какие возможности предоставляет программа MS Visio?
- 1.7. Каким образом определяются функциональные требования к системе?
- 1.8. Что отображает диаграмма «вариантов использования», каковы её элементы, какие между ними могут быть связи?

Тема 2. Организация ревьюирования. Инструментальные средства ревьюирования.

- 2.1. Что такое жизненный цикл программного обеспечения?
- 2.2. Что такое спецификации, какие сведения они содержат?
- 2.3. Что представляет собой структурный анализ?
- 2.4. В чем заключается этап реализации программного обеспечения?
- 2.5. Какие методы оценки трудоемкости разработки программного обеспечения вы знаете
- 2.6. Какие способы записи алгоритма вы знаете?
- 2.7. Какие работы выполняются в процессе обеспечения качества программного продукта?
- 2.8. Что значит ревью кода? Основная цель ревью кода.
- 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?

3.1.3 Перечень самостоятельных работ.

Задача № 1. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран его первую (старшую) цифру. Программист торопился и написал программу неправильно.

Найдите все ошибки в этой программе (их ровно 2). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

```
n = int(input())
while n > 0:
    n %= 10
print(n)
```

Решение:

```
n = int(input())
ns=n
kl=0
while n != 0:
    kl+=1
    n = n // 10

n1 = ns // 10**(kl-1)
print(n1)
```

Задача № 2. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран произведение цифр введённого числа. Программист торопился и написал программу неправильно.

Найдите все ошибки в этой программе (их ровно 3). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

```
n = input()
product = n % 10
while n >= 10:
    digit = n % 10
    product = product * digit
    n //= 10
print(product)
```

Решение:

```
n = int(input())
product = 1
while n!=0:
    digit = n % 10
    product = product * digit
    n //= 10
print(product)
```

3.2. Типовые задания для проведения итоговой аттестации

Другая форма контроля

- 1. Выберите свойства структуры ИС (один или несколько вариантов ответов):**
 - a) функциональное;
 - b) логическое;
 - c) практические;
 - d) обеспечивающее.
- 2. Что отражает модель жизненного цикла ИС:**
 - a) различные состояния системы, начиная с момента возникновения необходимости в данной ИС и заканчивая моментом ее полного выхода из употребления;
 - b) процесс проектирования ИС;
 - c) события, происходящие с системой в процессе ее создания и использования;
- 3. Этап, занимающий наибольшее время, при разработке программы:**
 - a) тестирование;
 - b) сопровождение;
 - c) проектирование;
 - d) программирование;
- 4. Первый этап в жизненном цикле программы:**
 - a) анализ требований;
 - b) проектирование;
 - c) разработка технической документации;
- 5. Один из необязательных этапов жизненного цикла программы:**
 - a) проектирование;
 - b) тестирование;
 - c) программирование;
 - d) оптимизация
- 6. Самый большой этап в жизненном цикле программы:**
 - a) эксплуатация;
 - b) изучение предметной области;
 - c) тестирование;

d) корректировка ошибок

7. Укажите, к какому уровню детализации относится диаграмма сущность-связь:

a) Модель данных нижнего уровня (детальное представление структуры данных);

b) Модель данных верхнего уровня (слабо детализирована);

c) Модель данных среднего уровня (более подробное представление данных);

8. Самый важный критерий качества программы:

a) быстроедействие;

b) функциональность (корректность);

c) простота эксплуатации

9. Один из способов оценки качества ПО:

a) сравнение с аналогами;

b) наличие документации;

c) оптимизация программы;

d) структурирование алгоритма

10. Существует ли связь между эффективностью и оптимизацией программы:

a) да;

b) нет;

c) в случаях коллективной разработки ПО;

d) в случаях индивидуальной разработки ПО

11. Одним из способов оценки надежности ПО является:

a) трассировка;

b) оптимизация;

c) тестирование

12. Есть ли ошибка в программном коде (язык программирования C++):

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    while (n > 0)
    {
        cout << n << endl;
        n--;
    }
}
```

- a) Нет ошибки;
- b) Есть ошибка, неинициализированная локальная переменная;
- c) Есть ошибка, нет «;» после while.

13. Определите причину ошибки в программном коде на 9 строке (язык программирования C++)

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      setlocale(LC_ALL, "Russian");
6      int month;
7      cout << "month = ";
8      cin >> month;
9      if (month = 2)
10         cout << "Февраль" << endl;
11     else
12         cout << "Не февраль" << endl;
13 }
```

- a) Не верно указан тип данных переменной month;
- b) Не верно задали имя переменной;
- c) Указана не верно операция сравнения.

14. Что выполняет данная программа (язык программирования C++)

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int n;
6      cout << "n = ";
7      cin >> n;
8      if (n % 2 == 1)
9          cout << "YES" << endl;
10 }
```

- a) Программа находит нечетные числа;
- b) Программа находит четные числа;
- c) Программа выполняет расчет остатка и выводит его на экран.

15. Как выполнить правильно проверку принадлежности переменной n промежутку [1 ; 20].

- a) if (n > 1 and n < 20)
- b) if (n >= 1 and n <= 20)
- c) if (n >= 1 or n <= 20)

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

« 06 » 20 26 г.

(подпись) (И. О. Фамилия)

« ____ » ____ 20 ____ г.

(подпись) (И. О. Фамилия)

« ____ » ____ 20 ____ г.

РАБОЧАЯ ПРОГРАММА

Дисциплина:	Моделирование и анализ программного обеспечения
Индекс:	МДК 02.03
Специальность:	09.02.11 Разработка и управление программным обеспечением
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 №138 "Об утверждении федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.11 Разработка и управление программным обеспечением".

Разработчик Капембва Марине Андраниковна преподаватель СПО УФ УГТУ.

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

СОГЛАСОВАНО

Заместитель директора УФ УГТУ
по учебной работе



О.В. Филиппова