

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Физический факультет

УТВЕРЖДЕНО:
декан физического факультета
С.Н. Филимонов

Оценочные материалы по дисциплине

Технология программирования

по направлению подготовки

03.03.02 Физика

Направленность (профиль) подготовки:
«Фундаментальная и прикладная физика»

Форма обучения
Очная

Квалификация
Бакалавр

Год приема
2025

СОГЛАСОВАНО:
Руководитель ОП
С.Н. Филимонов

Председатель УМК
О.М.Сюсина

Томск – 2025

1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-3 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.

ПК-1 Способен проводить научные исследования в выбранной области с использованием современных экспериментальных и теоретических методов, а также информационных технологий.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК 3.2 Соблюдает основные требования информационной безопасности при решении задач профессиональной деятельности

ИПК 1.1 Собирает и анализирует научно-техническую информацию по теме исследования, обобщает научные данные в соответствии с задачами исследования

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- контрольная работа;
- практическая работа.

Примерные задания контрольных работ (ИПК 1.1.):

1. Перечислите возможные операции с целыми числами. Приведите пример выражений с данными операциями.

2. Что такое процедура? Приведите пример правильно оформленной процедуры. Чем процедура отличается от функции.

3. Глобальные и локальные переменные. Опишите правила их применения в процедурах и функциях.

4. Перечислите и опишите процедуры для работы с файлами.

5. Приведите 5 – 7 примеров возможных конвенций именования переменных.

6. Дайте 5 – 7 советов по сокращению имен (переменных, процедур, функций и т.д.)

7. Начертите блок-схему алгоритма, структурированного по методу Ашкрофта-Манна.

Критерии оценивания: Результаты контрольной работы определяются оценками «зачтено» или «не зачтено». Оценка «зачтено» ставится, если обучающийся ответил правильно как минимум на 60% вопросов.

Примеры задач для практических занятий (ИОПК 3.2., ИПК 1.1.):

Упражнение на цикл REPEAT. Составьте программу вычисления суммы членов бесконечного ряда

$$z = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots + (-1)^n \frac{x^{2n}}{(2n)!} + \dots$$

с точностью до члена ряда, меньшего заданной точности ϵ .

Подсказка: возможно несколько решений; одно из них очень простое и не требует вычисления больших степеней и факториала.

Поскольку это есть разложение в ряд $\cos(x)$, т.е. $z = \cos(x)$, воспользуйтесь этим знанием для проверки.

Упражнение на строки. Возьмите из директории упражнений текст 5_1.txt и скопируйте его в свою директорию. Напишите программу, которая удаляет лишние пробелы между словами этого текста (между словами должно быть не более одного пробела) и записывает результат, т.е. отредактированный текст, в другой текстовый файл.

Ограничение: В данном упражнении считается, что в начале строк пробелов нет, а длина каждой строки текста не превышает 255 символов.

Упражнения на множества. Дан текст на английском языке (6_1.txt). Программа должна подсчитать: 1) общее количество символов текста, 2) общее количество букв, 3) общее количество остальных (кроме букв) символов. Кроме того, программа должна подсчитать, сколько раз в тексте встречается каждая буква (безотносительно к регистру, естественно). Результаты вывести в текстовый файл, оформить красиво: чтобы было понятно, что есть что.

Указание: Счётчики должны быть независимыми. Т.е. показания одного счётчика не может быть вычислено через показания другого.

Подсказки: 1) Существует функция `UpCase`, 2) Индексом массива может быть любой дискретный тип (кроме `longint`).

Упражнение на указатели. В разных трансляторах матрицы располагаются в памяти по-разному: по строкам, по столбцам.

Выяснить, как ваша версия языка программирования размещает в памяти двумерную матрицу, трёхмерную матрицу.

Подсказка: нужно создать двумерную или трёхмерную матрицу и считать её из памяти как одномерную с помощью указателя соответствующего типа.

Критерии оценивания: Задача считается решенной в полном объёме, если программа работает правильно и имеет хороший стиль программирования.

Под хорошим стилем программирования понимается эффективное, универсальное, с защитой от ошибок и с обработкой особых случаев программирование. При этом код программы написан в соответствии с основными принципами, предъявляемыми к наименованию имен переменных и к форматированию.

3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания

Экзамен проводится в письменной форме по билетам. К сдаче экзамена допускаются только те студенты, кто выполнил все контрольные и практические работы. Билет содержит два теоретических вопроса и практическое задание по одной из тем дисциплины, проверяющих ИОПК 3.2., ИПК 1.1. Продолжительность экзамена 1.5 часа.

Пример билетов к экзамену:

Билет 1.

1. Ссылочные типы
2. Циклы

Билет 2.

1. Процедуры и функции (без рекурсии)
2. Табличные методы: таблицы с прямым доступом подробно, остальные — только суть.

Билет 3.

1. Типы: целые, вещественные, символьный, булевский, ограниченные, перечислимые
2. Записи, записи с вариантами.

Билет 4.

1. Массивы
2. Строковые типы

Билет 5.

1. Операторы: присваивания, перехода, условный, выбора (case).
2. Динамические переменные

Билет 6.

1. Рекурсия
2. Принципы форматирования

Билет 7.

1. Множественные типы
2. Принципы именования

Билет 8.

1. Модули
2. Эффективность программ

Билет 9.

1. Эквивалентность типов. Совместимость типов.
2. Файловые типы. Подробно текстовые файлы.

Примеры задач на экзамене:

Задача 1. Дан целочисленный массив. Упорядочить массив так, чтобы сначала размещались все отрицательные компоненты (с сохранением порядка их следования в первоначальном массиве), а затем все положительные (также с сохранением порядка), не используя при этом вспомогательного массива. Исходный массив и результат вывести в текстовый файл.

Задача 2. Дана строка длиной L , заполненная буквами алфавита в произвольном порядке. Написать программу, проверяющую можно ли составить слово S из букв строки. Каждую букву можно использовать только один раз. Строку, слово и результат вывести в текстовый файл.

Задача 3. Напишите программу, которая читает целое положительное число $N \leq 999$ в десятичном представлении, а на выходе выдаёт это же число в десятичном представлении и на естественном языке.

Например:

7 семь
204 двести четыре
52 пятьдесят два

Задача 4. Имеется список из 10 сортов роз (названия придумайте сами). В данной местности розы выращивают три цветовода, каждый по несколько сортов. Определить те сорта, которые 1) имеются у каждого из цветоводов, 2) есть хотя бы у одного цветовода, 3) нет ни у кого.

Задача 5. В небоскребе N этажей и всего один подъезд. На каждом этаже по 3 квартиры. Лифт останавливается только на нечетных этажах. Это своеобразный лифт, где нужно набирать не номер этажа, а номер квартиры. Напишите программу для управления лифтом: на каком этаже он должен останавливаться, если набран номер квартиры M . Выход из программы, если набран 0.

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Критерии оценивания ответа:

отлично	Полный развернутый ответ на все вопросы. Задача решена в полном объеме, хороший стиль программирования
хорошо	Неполный ответ на вопросы. Программа работает правильно, имеются погрешности стиля.
удовлетворительно	Неполный ответ, не на все вопросы. Программа работает правильно, или с небольшими ошибками, но стиль программирования плох
неудовлетворительно	Нет ответа даже на общие вопросы. Программа не работает или работает неправильно

При оценке неудовлетворительно, полученной за теоретическую часть, задача не предлагается, экзамен считается не сданным, общая оценка неудовлетворительно. При оценке неудовлетворительно, полученной за практическую часть, экзамен считается не сданным независимо от оценки теоретической части, общая оценка неудовлетворительно. Во всех остальных случаях выводится средняя оценка.

4. Оценочные материалы для проверки остаточных знаний (сформированности компетенций)

Тест (ИОПК 3.2., ИПК 1.1.):

1. Что называется массивом?
 - а) Под массивом понимается совокупность конечного числа элементов различных типов.
 - б) Под массивом понимается совокупность конечного числа элементов одного типа, каждый из которых имеет свой номер, называемый индексом.
 - в) Под массивом понимается неупорядоченная совокупность числа элементов одного типа.
2. Цикл с предусловием определяется служебным словом:
 - а) for;
 - б) while;
 - в) case;
 - г) repeat;
3. В операторе присваивания `sum:= sqr(x)+3*a` переменными являются
 - а) a, x, sum
 - б) x, a
 - в) sqr, x, a
 - г) sum, sqr, x, a
4. Что производит следующий фрагмент программы?

```
min:=a[1]; max:=a[1];
for i:=1 to n do
if a[i]>= max then begin
    max:=a[i]; imax:=i;
end;
writeln(min,'номер', min);
```

 - а) Находит максимальный элемент и печатает на экране.
 - б) Находит индекс максимального элемента и печатает на экране.
 - в) Находит максимальный элемент, его индекс и печатает их на экране.
5. Укажите правильное описание массива:
 - а) `var A, B, C: array [1..50] of real;`
 - б) `var A: array [1..50 of real];`
 - в) `var a: array[1...1000] of integer;`

6. Запишите число, которое будет напечатано в результате выполнения следующей программы.

Паскаль	Python	Си
<pre>var n, s: integer; begin n := 0; s := 0; while s <= 35 do begin s := s + 4; n := n + 1; end; write(n) end.</pre>	<pre>n = 0 s = 0 while s <= 35: s = s + 4 n = n + 1 print(n)</pre>	<pre>#include <stdio.h> int main() { int n = 0, s = 0; while (s <= 35) { s = s + 4; n = n + 1; } printf("%d", n); return 0; }</pre>

7. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 8, 4, 3, 0, 7, 2, 1, 5, 9, 6 соответственно, т.е. $A[0]=8$, $A[1]=4$ и т.д. Определите значение переменной s после выполнения следующего фрагмента этой программы.

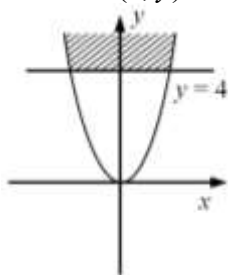
```
c := 0;
for i := 0 to 8 do
  if A[i] > A[i+1] then
  begin
    c := c + 1;
    t := A[i];
    A[i] := A[i+1];
    A[i+1] := t;
  end;
```

Ответы:

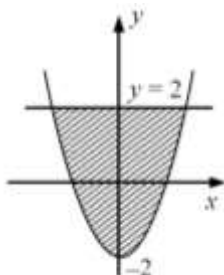
1. б; 2. б; 3. а; 4. в; 5. а; 6. 9; 7. 8.

Задачи (ИПК 1.1.):

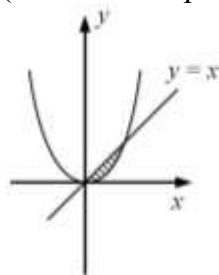
1. Напишите программу, позволяющую определить принадлежит ли точка с координатами (x, y) заштрихованной области (включая ее границы).



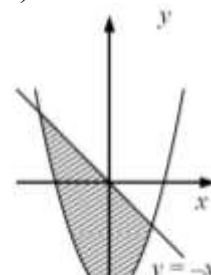
а)



б)



в)



г)

Ответы: а) $(-5, -1)$, $(4, 2)$, $(3, 8)$ – не принадлежит, $(2, 7)$, $(-3, 9)$, $(0, 4)$ – принадлежит;
 б) $(-5, -1)$, $(4, 2)$, $(3, 8)$ – не принадлежит, $(1, 1)$, $(0, -2)$, $(0, 2)$ – принадлежит;
 в) $(0.5, 0.6)$, $(-1, 3)$, $(1, -3)$ – не принадлежит, $(0.5, 0.5)$, $(0.6, 0.5)$, $(0.5, 0.25)$ – принадлежит;
 г) $(-5, 22)$, $(4, 2)$, $(3, -2)$ – не принадлежит, $(1, -1)$, $(-1, 0)$, $(-1, 1)$ – принадлежит;

2. Найти 10 первых натуральных чисел, оканчивающихся на цифру 7, кратных числу 9 и находящихся в интервале, левая граница которого равна 100. Ответ выведите в строку через запятую.

Ответ: 117, 207, 297, 387, 477, 567, 657, 747, 837, 927

3. Дан целочисленный массив из 30 элементов. Элементы массива могут принимать значения от 150 до 200 – рост учащихся выпускного класса. В команду по автогонкам входят все учащиеся, чей рост не более 175 см. Гарантируется, что такие учащиеся в классе есть. Опишите на одном из языков программирования алгоритм, который находит и выводит:

- а) номер и рост всех учащихся,
- б) номера и рост всех участников команды,
- в) рост самого высокого участника гоночной команды.

Пример работы программы для задачи 3:

а)	б)	в)
1 165	1 165	200
2 193	3 159	
3 159	6 160	
4 193	7 157	
5 199	8 158	
6 160	13 155	
7 157	14 167	
8 158	16 167	
9 188	18 173	
10 195	19 151	
11 184	20 152	
12 200	21 171	
13 155	23 167	
14 167	25 173	
15 180	27 175	
16 167	30 150	
17 187		
18 173		
19 151		
20 152		
21 171		
22 196		
23 167		
24 176		
25 173		
26 198		
27 175		
28 187		
29 194		
30 150		

Информация о разработчиках

Томилова Ирина Владимировна, к.ф.-м.н., кАиКГ ФФ, доцент,