

Глава 10. Зачет: методика его проведения, теоретический минимум и типовые задания.

Зачет проводится в практикуме и состоит из двух частей: теоретической и практической.

Теоретический минимум

Проверка теоретических знаний производится в виде опроса по любым вопросам, входящим в программу первого семестра и теоретический минимум. Вопрос, как правило, формулируется в виде просьбы написать на листе бумаги короткий (2-5 строк) код, иллюстрирующий то или иное понятие языка программирования. Обычно задается от двух до пяти вопросов. Студенты, не сдавшие теоретическую часть, автоматически не допускаются к выполнению практической части зачета.

Ниже приводится перечень сведений о языке программирования из теоретического минимума знаний, которые студент должен иметь для получения зачета.

1. Основы синтаксиса языка Си, структура консольного приложения.
2. Фундаментальные типы данных (**bool**, **char**, **int**, **double**).
3. Определение переменных и констант.
4. Оператор **sizeof** ().
5. Выражения, операции, комментарии.
6. Оператор приведения типа.
7. Операторы инкремента и декремента.
8. Приоритет операторов в выражениях.
9. Блоки и правила видимости переменных.
10. Условный оператор и оператор перехода (**if**, **goto**).
11. Оператор множественной альтернативы (**switch**).
12. Цикл **while**. Прерывание цикла. Переход к следующей итерации.
13. Цикл **do ... while**. Прерывание цикла. Переход к следующей итерации.
14. Цикл **for**. Прерывание цикла. Переход к следующей итерации.
15. Математические функции стандартной библиотеки Си (**<math.h>**).
16. Форматированный консольный ввод (**<stdio.h>**): параметры функции **scanf** ().
17. Форматированный консольный вывод (**<stdio.h>**): параметры функции **printf** ().
18. Форматированный файловый ввод-вывод (**<stdio.h>**).
19. Бесформатный файловый ввод-вывод (**<stdio.h>**).
20. Массивы. Передача массивов в параметрах функции.
21. Определение функции. Прототип функции. Рекурсия.
22. Параметры функции **main** ().
23. Раздельная компиляция программных модулей. Использование ***.h** файлов.
24. Внешние (**extern**) и глобальные переменные.
25. Статические (**static**) переменные.
26. Статические (**static**) функции.
27. Указатели и операторы, с ними связанные.
28. Указатель на функцию.
29. Функции для работы с динамической памятью **malloc** () / **realloc** () / **free** () .
30. Строки Си. Функции для работы со строками (**<string.h>**).
31. Функции для работы с символами (**<ctype.h>**).
32. Пользовательский тип данных **enum**.
33. Пользовательский тип данных **struct**.
34. Пользовательский тип данных **union**.
35. Определение синонимов типов (**typedef**).
36. Директивы препроцессора для условной компиляции и их использование.
37. Директивы препроцессора для включения файлов и их использование.
38. Макроопределения препроцессора (с параметрами и без).

Вопросы по теоретической части

Здесь приводятся примерные варианты билетов с вопросами к теоретическому минимуму, на которые студент должен уметь отвечать на зачете.

<p>1. Вариант</p> <ol style="list-style-type: none">1. Какие базовые типы данных Вы знаете? Сколько места они занимают в памяти компьютера?2. Что означает команда «continue» и где она применяется?3. Выделите память под вещественный массив чисел двойной точности из 123 элементов.
<p>2. Вариант</p> <ol style="list-style-type: none">1. Какие операторы языка Си Вы знаете? Какой у них приоритет?2. Какие виды циклов Вы знаете? Чем они отличаются друг от друга?3. Что такое прототип функции и где он используется?
<p>3. Вариант</p> <ol style="list-style-type: none">1. Что такое операции постфиксного инкремента и префиксного декремента?2. Что означает команда «break» и где она применяется?3. Что такое рекурсивная функция? Приведите пример.
<p>4. Вариант</p> <ol style="list-style-type: none">1. Что означает строка: <code>void func(int,int*);</code>2. Что такое метка и где она используется?3. Что такое цикл «с предусловием» и цикл «с постусловием»?
<p>5. Вариант</p> <ol style="list-style-type: none">1. Сколько места в памяти компьютера занимают вещественные переменные и переменные с двойной точностью? Что такое точность представления вещественного числа?2. Что означает команда «goto» и где она применяется?3. Как найти заданную подстроку в другой строке?
<p>6. Вариант</p> <ol style="list-style-type: none">1. Что такое массив и что такое указатель? Приведите примеры.2. Верно ли синтаксически написана программа: <code>void main() { }</code>3. Какие функции из библиотеки <string.h> Вы знаете?
<p>7. Вариант</p> <ol style="list-style-type: none">1. Что такое операция получения адреса? Где она используется?2. Какие функции ввода-вывода Вы знаете?3. Что такое перечислимый тип? Как он определяется?
<p>8. Вариант</p> <ol style="list-style-type: none">1. Как получить значение по указателю? Приведите примеры.2. Какие форматирующие последовательности функции printf() Вы знаете?3. Опишите структуру, состоящую из символьного массива из 12 членов, целого и вещественного числа.
<p>9. Вариант</p> <ol style="list-style-type: none">1. В каком случае можно изменить размер ранее выделенного массива? Как это сделать?2. Напишите цикл do ... while(), меняющий порядок элементов массива на противоположный.3. Для чего нужен оператор typedef?
<p>10. Вариант</p> <ol style="list-style-type: none">1. Напишите цикл for, меняющий порядок элементов массива на противоположный.2. Как получить указатель на переменную?3. Какие функции для работы с отдельными символами Вы знаете?
<p>11. Вариант</p> <ol style="list-style-type: none">1. Как записать массив в бинарный (неформатированный) файл?2. Какие значения будут иметь все переменные в результате выполнения программы: <code>int a = 1, b = 2, c = 3;</code> <code>c -= a += b = c;</code>3. Как вернуть из функции массив?

<p>12. Вариант</p> <ol style="list-style-type: none"> 1. Откройте файл "myfile.txt" на чтение. 2. Какие значения будут иметь все переменные в результате выполнения программы: <pre>int a = 1, b = 2; a += b++;</pre> 3. Что такое union?
<p>13. Вариант</p> <ol style="list-style-type: none"> 1. Как передать функцию в другую функцию? Как создать массив функций? 2. Как с помощью функции языка Си прочитать в три переменные три вещественных числа, разделенных пробелами? 3. Как обратиться к функции, определенной в другом файле с исходным текстом?
<p>14. Вариант</p> <ol style="list-style-type: none"> 1. Что такое заголовочный файл? 2. Как определить размер вещественного массива в байтах? 3. Напишите функцию, которая складывает два вещественных числа и возвращает результат.
<p>15. Вариант</p> <ol style="list-style-type: none"> 1. Для чего нужны статические переменные? Чем они отличаются от глобальных? 2. Что такое макроопределение с параметром? Приведите пример. 3. Как распечатать на экран строки из текстового файла, открытого на чтение?
<p>16. Вариант</p> <ol style="list-style-type: none"> 1. Как найти длину текстовой строки? 2. Какие значения будут иметь все переменные в результате выполнения программы: <pre>int a = 1, b = 4, c = 3; a = --b > c;</pre> 3. Для чего применяется оператор default?
<p>17. Вариант</p> <ol style="list-style-type: none"> 1. Как реализовать компиляцию некоторой функции в зависимости от условия? 2. Напишите циклы while, транспонирующие матрицу. 3. Опишите структуру, состоящую из имени, фамилии и года рождения студента.
<p>18. Вариант</p> <ol style="list-style-type: none"> 1. Определите двумерный вещественный массив. 2. Какие значения будут иметь все переменные в результате выполнения программы: <pre>int a = 1; int* b = &a; (*b)++;</pre> 3. Определите массив из трех структур, которые представляют разные точки на плоскости.
<p>19. Вариант</p> <ol style="list-style-type: none"> 1. Как изменить значение глобальной переменной, заданной в другом файле исходного текста проекта? 2. Какие форматирующие последовательности функции printf() Вы знаете? 3. Как задать перечислимый тип из трех убывающих элементов?
<p>20. Вариант</p> <ol style="list-style-type: none"> 1. Выделить динамическую память под строку из 20 символов. 2. Как передать аргументы с командной строки в функцию main()? 3. Распечатайте число π с тремя знаками после десятичной точки.
<p>21. Вариант</p> <ol style="list-style-type: none"> 1. Как закрыть функцию от использования в других исходных текстах проекта? 2. Какое значение будет иметь переменная x: <pre>double x = 4; x *= 1/2;</pre> 3. Определите функцию, которая возвращает сумму элементов переданного массива.
<p>22. Вариант</p> <ol style="list-style-type: none"> 1. Как найти в строке любой из символов, входящих в строку "abc"? 2. Как напечатать целое число с обязательным выводом знака «плюс» для положительных чисел? Как сделать выравнивание по левой границе поля для этого числа? 3. Как принудительно преобразовать тип целого числа к вещественному типу?

<p>23. Вариант</p> <ol style="list-style-type: none"> 1. Как получить адрес третьего элемента вещественного массива? Если этот адрес уменьшить на единицу – куда он будет указывать? 2. Какое значение получит переменная <i>z</i> в результате выполнения программы: <pre>int a = 1, b = 2, c = 3, d = 4; bool z; z = a < b b < c && d < c;</pre> 3. Как определить, является ли заданный символ пробельным?
<p>24. Вариант</p> <ol style="list-style-type: none"> 1. Создайте макроопределение, которое вычисляет сумму двух своих аргументов. 2. Что делает функция <code>scanf()</code>? 3. Откройте файл "myfile.txt" на запись в режиме «запись в конец файла».
<p>25. Вариант</p> <ol style="list-style-type: none"> 1. Как проверить, открылся ли файл? 2. Какие значения будут иметь все переменные в результате выполнения программы: <pre>int a = 0, b = 1, c = 1; a = b << c++;</pre> 3. Как обратиться к переменной, определенной в ином исходном тексте проекта?
<p>26. Вариант</p> <ol style="list-style-type: none"> 1. Как объявить и инициализировать указатель на переменную? 2. Какие форматирующие последовательности функции <code>printf()</code> Вы знаете? 3. Объявите структуру, состоящую из даты (3 числа) и названия месяца (массив символов).
<p>27. Вариант</p> <ol style="list-style-type: none"> 1. Как выделить память под двумерный динамический массив? 2. Напишите цикл <code>while()</code>, меняющий порядок элементов массива на противоположный. 3. Приведите пример использования перечисления <code>enum</code>.
<p>28. Вариант</p> <ol style="list-style-type: none"> 1. Напишите цикл <code>for</code>, находящий последний минимальный элемент массива. 2. Как передать в функцию переменную по указателю? 3. Как объединение <code>union</code> располагает элементы в памяти?
<p>29. Вариант</p> <ol style="list-style-type: none"> 1. Как прочитать содержимое бинарного (неформатированного) файла с целыми числами? 2. Какие значения будут иметь все переменные в результате выполнения программы: <pre>int a = 1, b = 2, c = 4; c -= a += b++;</pre> 3. Как передать из функции созданный в ней динамический массив?
<p>30. Вариант</p> <ol style="list-style-type: none"> 1. Как добавить данные в уже имеющийся файл? 2. Какие значения будут иметь все переменные в результате выполнения программы: <pre>int a = 1, b = 2; a += ++ b;</pre> 3. Приведите пример использования объединения <code>union</code>.

Практические задания

Практические навыки программирования проверяются на одной типовой задаче, подобной тем, что выполнялись в течение семестра. Студент должен ее выполнить в течение одной пары (два академических часа) от начала до конца.

Рекомендуется основную массу заданий выдавать по образцу, приведённому в девятой главе: написать программу, которая читает данные (например, матрицу чисел произвольной размерности) из одного файла, как-то преобразует прочитанные данные (например, транспонирует прочитанную матрицу) и записывает получившийся результат в другой файл.

В приводимых ниже типовых заданиях для зачета необходимо написать законченную программу, которая тестирует заданную функцию. Это примеры зачетных заданий, но на реальном зачете могут быть даны другие, примерно такой же сложности.

<p>1. Вариант Функция записывает в файл все четырехзначные натуральные числа из диапазона (2000 – 3000), в записи которых нет двух одинаковых цифр, подсчитывает количество таких чисел, возвращает в <code>main()</code>.</p>
<p>2. Вариант Функция считывает из файла вещественный массив неизвестной длины (до конца файла). Надо найти в массиве два элемента, модуль разности которых имеет наименьшее значение. Напечатать эти элементы и их индексы.</p>
<p>3. Вариант Функция, работающая как простейший калькулятор, выполняющий действия «+», «-», «*», «/» над двумя целыми массивами одного размера, считанными из файла. Результат записать в файл.</p>
<p>4. Вариант Функция считывает из файла целый массив неизвестной длины (до конца файла), находит в массиве минимальный по модулю элемент и заменяет им все элементы массива с четными номерами. Измененный массив записывается в файл.</p>
<p>5. Вариант Функция считывает из файла целый массив неизвестной длины (до конца файла). Необходимо найти максимальный по модулю элемент этого массива и заменить им все нулевые элементы массива. Подсчитать и вернуть в <code>main()</code> число таких элементов. Измененный массив сохранить в другой файл.</p>
<p>6. Вариант Функция считывает из файла целый массив неизвестной длины (до конца файла). Определить и напечатать три наибольших элемента этого массива. Подсчитать их сумму, вернуть в <code>main()</code>.</p>
<p>7. Вариант Функция считывает из файла вещественный массив неизвестной длины (до конца файла), находит в массиве минимальный по модулю элемент и заменяет им все элементы с нечетными номерами. Измененный массив записывается в файл.</p>
<p>8. Вариант Функция получает два целых числа, m и n, создает массив из простых чисел, расположенных в интервале от m до n, подсчитывает их количество, массив записывает в файл.</p>
<p>9. Вариант Функция считывает из файла целый массив неизвестной длины (до конца файла), запрашивает у пользователя целое число k и находит элемент массива, значение которого наиболее близко к введенному числу и возвращает его индекс.</p>
<p>10. Вариант Функция, вычисляющая методом трапеций определённый интеграл от непрерывной положительной $f(x)$ на отрезке $[a, b]$. Шаг разбиения уменьшать, пока площадь от итерации к итерации изменяется более, чем на ϵ. Параметры a, b и ϵ передаются из <code>main()</code>, куда и возвращается результат.</p>
<p>11. Вариант Функция, вычисляющая методом верхних прямоугольников определённый интеграл от непрерывной положительной $f(x)$ на отрезке $[a, b]$. Шаг разбиения уменьшать, пока площадь от итерации к итерации изменяется более, чем на ϵ. Параметры a, b и ϵ передаются из <code>main()</code>, куда и возвращается результат.</p>

<p>12. Вариант Функция, вычисляющая методом деления отрезка пополам с точностью эпсилон корень уравнения $f(x) = 0$ на отрезке $[a, b]$. Параметры a, b и ϵ передаются из <code>main()</code>, куда и возвращается результат. Функция $f(x)$ вычисляется в отдельном блоке.</p>
<p>13. Вариант Функция, которая считывает массив чисел неизвестной размерности из файла, сортирует его и записывает в другой файл. Функция принимает название входного и выходного файла. Необходимо использовать функции выделения динамической памяти.</p>
<p>14. Вариант Функция, которая считывает из файла численные данные, записанные в 2 колонки, и переписывает их в другой файл в строку через запятую. Необходимо использовать функции выделения динамической памяти.</p>
<p>15. Вариант Функция, которая находит в текстовом файле заданную пользователем строчку и указывает номер строки файла, в которой строка была найдена.</p>
<p>16. Вариант Функция, подсчитывающая количество букв в тексте, находящемся в указанном файле. Функция принимает название входного файла и возвращает количество букв (цифры и пунктуацию не учитывать).</p>
<p>17. Вариант Функция, обрабатывающая текстовый файл. В файле имеется массив x координат, размер которого заранее не известен. Функция считывает эти значения и записывает в выходной файл x и $\sin(x/2)$ в 2 колонки. Необходимо использовать функции выделения динамической памяти.</p>
<p>18. Вариант Функция, обрабатывающая текстовый файл. В файле имеются две колонки вещественных чисел (пары координат x, y). Надо отсортировать пары чисел по координате x и записать отсортированные таким образом две колонки в другой файл.</p>
<p>19. Вариант Имеется 2 текстовых файла, в которых записаны два n-мерных вектора. Значение n заранее неизвестно. Создать функцию, которая считывает вектора из файлов, производит сложение или поэлементное умножение векторов и записывает результат в 3-й файл. Функция принимает название входного и выходного файлов и знак операции («+» или «*»). Необходимо использовать функции выделения динамической памяти.</p>
<p>20. Вариант Функция, перемножающая две матрицы вещественных чисел произвольного размера. Каждая матрица записана в своем текстовом файле. Функция должна считывать эти матрицы, перемножать их и записывать результат в третий файл. Если размерности матрицы не позволяют выполнить умножение (количество столбцов первой матрицы не равно количеству строк второй матрицы), то вывести соответствующее сообщение. Функция принимает название входных и выходного файлов.</p>
<p>21. Вариант Создать структуру, представляющую собой рациональное число (пара целых чисел: числитель и знаменатель). Написать функции для выполнения четырех арифметических действий с такими структурами. Написать тестовую программу, в которой нужное арифметическое выражение вводит пользователь.</p>
<p>22. Вариант Написать функцию, которая считывает из указанного ей файла матрицу произвольной размерности, считает среднее арифметическое и стандартное отклонение для каждой колонки этой матрицы и результат распечатывает на экране.</p>
<p>23. Вариант Написать функцию, которая сортирует в алфавитном порядке переданный ей массив англоязычных текстовых строк. Строки считываются из указанного файла, а результат выводится на экран.</p>

24. Вариант	Создать структуру, которая будет содержать вещественный динамический вектор произвольной размерности. Написать функцию вставки нового элемента в любое место этого вектора.
25. Вариант	Написать функцию определяющую количество различных чисел матрицы.
26. Вариант	Дан одномерный массив. Сформировать новый массив из элементов десятичное представление которых содержит только чётные цифры (0,2,4,6,8).
27. Вариант	Написать функцию сжимающую массив, удалив из него все элементы, которые находятся между первым и вторым нулевыми элементами.
28. Вариант	Написать функцию, которая в каждой строке матрицы находит наибольший элемент и меняет его местами с элементом главной диагонали.
29. Вариант	Написать функцию вставляющую нулевую строку перед строкой, где находится первый минимальный элемент матрицы.
30. Вариант	Написать функцию находящую произведение тех элементов квадратной матрицы, которые расположены над главной диагональю и сумму элементов, расположенных под побочной диагональю.
31. Вариант	Написать функцию удаляющую из массива все элементы, значения которых больше трех среднеквадратичных отклонений (больше трех сигма).
32. Вариант	Написать функцию определяющую количество «особых» элементов матрицы, считая элемент «особым», если он больше суммы остальных элементов своего столбца.
33. Вариант	Написать функцию создающую массив из порядковых номеров максимальных по модулю элементов в столбцах матрицы.
34. Вариант	Написать функцию оставляющую в массиве только те элементы, которые удовлетворяют двойному неравенству $A_0 < A_k < A_n$. $k=1,2,\dots,(n-1)$
35. Вариант	Написать функцию определяющую сумму элементов в тех столбцах матрицы, которые не содержат отрицательных элементов.
36. Вариант	Написать функцию упорядочивающую массив целых положительных чисел по возрастанию методом выбора и возвращающую номер минимального простого числа в полученном массиве.
37. Вариант	Написать функцию, которая добавляет минимальный элемент матрицы к элементам четных столбцов, значения которых находятся в заданном интервале $[a,b]$.
38. Вариант	Написать функцию находящую номер первой строки матрицы, не содержащей ни одного четного элемента, и циклически сдвинуть в ней элементы так, чтобы элемент главной диагонали стал первым элементом этой строки.
39. Вариант	Написать функцию удаляющую из матрицы строку, в которой больше всего отрицательных элементов.
40. Вариант	Написать функцию сжимающую динамический массив с большим количеством нулей. Необходимо в каждой группе нулей первый ноль оставить, второй заменить на кол-во нулей в группе, остальные удалить.

41. Вариант Написать функцию удаляющую элементы динамического массива, расположенные между его минимальным и максимальным элементами.
42. Вариант Написать функцию, которая в квадратной матрице находит строку с наибольшей суммой элементов и поменяет её с первым столбцом.
43. Вариант Написать функцию меняющую местами левую верхнюю и правую нижнюю четверти матрицы.
44. Вариант Написать функцию удаляющую последний столбец матрицы, содержащий только отрицательные элементы.
45. Вариант Написать функцию находящую сумму элементов каждой диагонали матрицы, параллельной главной (начиная с одноэлементной диагонали $A_{0,m-1}$).
46. Вариант Написать функцию зеркально отражающую элементы квадратной матрицы относительно побочной диагонали. Вспомогательную матрицу не использовать.
47. Вариант Написать функцию поворачивающую матрицу на угол 180° (при этом элемент $A_{0,0}$ поменяется местами с $A_{m-1,m-1}$ и т. д.). Вспомогательную матрицу не использовать.
48. Вариант Дана матрица вещественных чисел. В каждом столбце найти максимальный и минимальный элементы, и сумму элементов, заключенных между ними.
49. Вариант Написать функцию, которая в каждой строке матрицы подсчитывает количество элементов, предшествующих максимуму.
50. Вариант Написать функцию, которая сортирует методом пузырька строки матрицы, все элементы которых четные.
51. Вариант Написать функцию находящую максимальное из чисел, встречающихся в массиве более одного раза.
52. Вариант Написать функцию обнуляющую элементы матрицы, лежащие одновременно выше главной и выше побочной диагоналей. Условный оператор внутри циклов не использовать (индексы внутреннего цикла начинать и заканчивать с учетом индексов внешнего цикла).
53. Вариант Написать функцию которая заменяет каждый элемент массива, кроме начального и конечного, на его среднее арифметическое с предыдущим и последующим элементом (выполняет скользящее усреднение).
54. Вариант Написать функцию, которая в массиве находит два числа, сумма которых наибольшая и делится нацело на 12.
55. Вариант Написать функцию возвращающую массив из элементов исходного массива, которые встречается более одного раза.
56. Вариант Написать функцию меняющая местами строки матрицы, содержащие минимальный и максимальный элементы.
57. Вариант Написать функцию, которая заменяет значение каждого не положительного элемента массива $X(n)$ абсолютной величиной соответствующего по номеру элемента массива $Y(n)$.
58. Вариант Написать функцию, которая формирует массив из элементов исходного массива, удовлетворяющего двойному неравенству $A_0 < A_K < A_{10}$.

59. Вариант

Упорядочить массив целых положительных чисел по возрастанию методом выбора и определить номер минимального простого числа

60. Вариант

Написать функцию, заменяющую числа массива X на соответствующие числа массива Y, если разница между ними больше заданного значения.