## Контрольные вопросы

## Лабораторная работа 10. Функции

1. Локальные и глобальные переменные.
2. Как выглядит определение функции?
3. Как выглядит объявление функции?
4. Что такое формальный параметр?
5. Что такое фактический параметр?
6. Что такое параметр по умолчанию? Приведите примеры.
7. Как осуществляется вызов функции?
8. Как осуществляется передача параметров в функцию?
9. Как описать функцию, не возвращающую значения?
10. В чем разница передачи параметров по значению от передачи параметров по адресу?
11. Как передать параметр по ссылке? Приведите примеры.
12. Как передать параметр через указатель? Приведите примеры.
13. Каков механизм сохранения функцией значения между вызовами функции?
14. Что такое inline-функция?
15. Что такое шаблон функции? Правила описания
16. Что такое перегрузка функции. Пример
17. Возможные ошибки при перегрузке функций.
18. Что такое рекурсия. Приведите пример

## Лабораторная работа 11. Указатели, ссылки

1. Что такое ссылка? Как объявляется ссылка?
2. Что такое указатель? Как объявляется указатель?
3. Поясните отличие ссылки от указателя.
4. Способы инициализации указателя. Приведите примеры.
5. Что такое указатели константы и указатели переменные?
6. В чем отличие указателя на константу от указателя-константы?
7. Почему указатель не может существовать как самостоятельный тип?
8. С какой целью в программе может быть использован указатель типа void?
9. Что будет являться результатом разыменования указателя типа void без приведения типов?
10. Что такое адрес переменной?
11. Объявите целочисленную переменную и проинициализируйте на нее указатель.
12. Чему будет равно значение указателя int\* ptr = 0; после выполнения операции ptr++?
13. Какие операции можно выполнять с указателями?
14. Что такое операция получения адреса и как она используется в С++ ?
15. Как осуществить доступ к переменной по указателю? Что такое операция разыменования?
16. В чем состоит отличие в использовании символа \* в операции разыменования и при объявлении указателя?
17. Какие арифметические операции можно выполнять с указателями?
18. Что будет являться результатом разыменования указателя типа void без приведения типов?
19. Как изменится значение указателя после применения к нему операции инкремента (декремента)?
20. Почему для указателей определены сложение и вычитание только с целыми константами?
21. Два указателя разных типов указывают на одно и то же место в памяти. Сравните результаты операций разыменования и взятия адреса с такими указателям. Сравните значения указателей.
22. Если объект занимает в памяти несколько байтов, то какой адрес является значением указателя на этот объект?
23. Каким образом при разыменовании указателей становится известно, сколько байтов памяти доступно?

## Лабораторная работа 12. Динамические массивы, функции

1. Каким образом можно задавать указатель на массив?
2. Каким образом можно задавать указатель на функцию?
3. Для чего предназначена функция malloc()?
4. Что делает функция free() и в какой библиотеке она определена?
5. Как можно использовать указатели при работе с массивами?
6. Как можно обращаться к элементам массива через указатель?
7. Как при помощи указателя можно передать в функцию значение переменной, массив?
8. Как при помощи указателя из функции можно возвратить два и более значений?
9. Что такое операции new - delete? Как при помощи операции new можно создать динамический массив заданного размера?
10. Как при помощи операции new можно создать двумерный динамический массив заданного размера?
11. Для чего предназначена функция malloc()?
12. Что делает функция free() и в какой библиотеке она определена?
13. Как выделяется область динамической памяти? Какие функции для этого используются?
14. Как высвобождается область динамической памяти? Какие функции для этого используются?
15. Может ли функция возвращать значение типа указатель? Если да, то как объявляется прототип такой функции?
16. Могут ли параметрами функции быть указатели на объекты? Если да, то как происходит передача фактических параметров при вызове функции?
17. Могут ли параметрами функции быть указатели на функции? Если да, то как происходит передача фактических параметров при вызове функции?
18. Как передаются массивы в качестве аргументов функции?
19. Что такое "Висячие" указатели и как этого избежать?
20. Из-за чего происходит «утечка" памяти при работе.
21. Типичные ошибки при работе с динамической памятью
22. Как осуществить проверку, произошло ли выделение динамической памяти по данному указателю?

## Лабораторная работа 13. Работа со строками

1. Функции ввода-вывода одиночного символа.
2. Функции ввода-вывода стандартного текстового (символьного) потока
3. Стандартные потоки ввода-вывода символьных данных и строк
4. Форматированный ввод-вывод символьных данных и строк
5. Что такое строка? Как задаются строки в программе на С++?
6. Запишите возможные способы начальной инициализации строки.
7. Какой символ стоит в конце строки?
8. Какая нужна вспомогательная библиотека для обработки символов и строк?
9. Какие операции определены для текстовых строк?
10. Перечислите операции над символами?
11. Можно ли выполнить присваивание символьной переменной числового значения? Почему?
12. Допустима ли операция сравнения над символами? Если да, то каким образом определены отношения "больше" и "меньше"?
13. Почему символ и строка, состоящая из одного символа, занимают разный объем памяти?
14. Как получить доступ к отдельным символам строки?
15. Как можно изменить строку в процессе выполнения программы?
16. Почему обращения к строке через ее имя и через указатель эквивалентны?
17. Почему в качестве параметра функции передается адрес строки, а не сама строка символов?
18. Почему при сравнении строк важен регистр символов?
19. Как сравниваются строки разной длины?
20. Какие возможны последствия при обращении к неинициализированной строке?
21. Почему при формировании строки без использования стандартных функций необходимо дописывать символ конца строки? Почему этого не требуется при считывании строк с клавиатуры?
22. Для чего предназначена функция strcpy() и в какой библиотеке она определена?
23. Что выполняет функция strcmp()?
24. Что возвращает функция strlen()?
25. Приведите простейший алгоритм копирования одной строки в другую.
26. Почему в С++ не выполняется операция прямого присваивания значения строке?

## Лабораторная работа 14. Текстовые файлы

1. Текстовый файл. Его особенности.
2. В чем отличия текстовых и двоичных файлов с точки зрения представления данных?
3. Перечислите стандартные потоки ввода-вывода?
4. Файловый поток, виды, создание.
5. Основные алгоритмы работы с файлами используя файловые потоки: запись данных в файл, чтение данных из файла.
6. Основные методы класса ifstream
7. Основные методы класса ofstream
8. Какие значения возвращает метод проверки открытия файла в зависимости от результата?
9. Какие значения возвращает метод проверки достижения конца файла в зависимости от результата?
10. Какие режимы открытия файлов существуют?
11. Назовите возможные причины ошибок, возникающих при открытии файлов для чтения и для записи.
12. Можно ли один и тот же файл открыть несколько раз, не закрывая после каждого открытия?
13. Можно ли один и тот же файл одновременно открыть для записи и для чтения? Если да, то где будет находиться файловый указатель?
14. Сохранится ли информация в файле, если его не закрыть в программе с помощью функции? Обоснуйте ответ.
15. Как осуществляется чтение символов и строк из файла?
16. Что такое последовательный доступ к текстовым файлам?
17. В чем особенность произвольного доступа к текстовым файлам?

## Лабораторная работа 15. Структуры

1. С какой целью обычно используют [структуры](https://study.chuvsu.ru/mod/resource/view.php?id=42063)?
2. К каким типам данных относятся [структуры](https://study.chuvsu.ru/mod/resource/view.php?id=42063)?
3. Способы [объявления](https://study.chuvsu.ru/mod/forum/view.php?id=14085) структур и определения структурных объектов
4. Инициализация [структуры](https://study.chuvsu.ru/mod/resource/view.php?id=42063). Способы.
5. Присваивание структур
6. Как осуществлять доступ к полям [структуры](https://study.chuvsu.ru/mod/resource/view.php?id=42063)?
7. Определение размера памяти, выделяемой под структуру.
8. Выравнивание полей в структурах в памяти
9. Массивы структур
10. Как осуществить доступ к отдельной записи массива структур
11. Как описать доступ к полю отдельной записи массива структур
12. Как описать указатель на структуру?
13. Какие данные о структуре содержит указатель на эту структуру?
14. Как выполняется инициализация указателя на структуру?
15. Доступ к элементам структур через указатель. Способы
16. Операции над указателями на [структуры](https://study.chuvsu.ru/mod/resource/view.php?id=42063)
17. [Структуры](https://study.chuvsu.ru/mod/resource/view.php?id=42063) и функции
18. Какие существуют способы передачи структур функциям?
19. Какие существуют способы передачи массива структур функциям?
20. Что такое битовые поля?
21. Как обратится к элементу битового поля?

## Лабораторная работа 16. Бинарные файлы

1. Что такое бинарный файл?
2. Как изменить текущую позицию в файле?
3. Как узнать текущую позицию в файле?
4. Для чего используется метод clear()?
5. Объясните понятие прямой доступ к файлу.
6. Перечислите команды, обеспечивающие прямой доступ к файлу.
7. Перечислите команды, обеспечивающие считывание и запись двоичных блоков.
8. Какая команда обеспечивает проверку достижения конца файла.
9. Какие режимы открытия файлов существуют?
10. Какие значения возвращает метод открытия файла в зависимости от результата?
11. Какие значения возвращает метод проверки достижения конца файла?
12. Функции ввода-вывода данных файла: символьный ввод-вывод, строковый ввод-вывод, блоковый ввод-вывод.
13. Основные алгоритмы работы с файлами: запись данных в файл, чтение данных из файла.
14. В чем основные отличия в организации символьного, строкового, блокового и форматированного ввода-ввода в файлы.

## Лабораторная работа 17-18. Динамические структуры данных

1. Динамические структуры данных, виды.
2. Размещение в памяти динамических величин. За счет каких ресурсов выделяется память под динамические структуры?
3. Доступ к динамическим структурам из программного кода.
4. Смежное и связное представление данных в динамических структурах.
5. Какого типа может быть поле данных в динамической структуре?
6. Односвязные списки. Операции добавления, удаления и поиска данных.
7. Двусвязные списки. Операции добавления, удаления и поиска данных.
8. В чем отличие первого элемента однонаправленного (двунаправленного) списка от остальных элементов этого же списка?
9. В чем отличие последнего элемента однонаправленного (двунаправленного) списка от остальных элементов этого же списка?
10. Стек и очередь. Операции добавления, удаления и поиска данных.
11. С какой целью в программах выполняется удаление однонаправленного (двунаправленного) списка по окончании работы с ним? Как изменится работа программы, если операцию удаления списка не выполнять?
12. Контейнеры в STL, виды контейнеров.
13. Понятие итератора в библиотеке STL. Операции над итераторами.
14. Контейнер list в библиотеке STL. Основные методы.
15. Итераторы для list. Допустимые операции над итераторами.
16. Контейнер stack в библиотеке STL. Основные методы.
17. Контейнер vector в библиотеке STL. Основные методы.
18. Итераторы для vector. Допустимые операции над итераторами.