**<p><b>Виды линейных структур данных: </b></p>**

1. <p><b>Массив</b></p>
2. <p><b>Словарь данных</b></p>
3. <p><b>Динамический массив</b></p>
4. <p><b>Связные списки</b></p>

Ответ: 1, 3, 4

Комментарий: SD00100

**<p><b>Что такое массив? </b></p>**

1. <p><b>структура данных, хранящая набор значений идентифицируемых по индексу</b></p>
2. <p><b>пользовательский(собственный) тип данных</b></p>
3. <p><b>специальный блок инструкций, вызываемый при создании объекта</b></p>
4. <p><b>область в памяти, обеспечивающая начальные значения состояний</b></p>

Ответ: 1

Комментарий: SD001002

**<p><b>Основные операции с массивами: </b></p>**

1. <p><b>Вставка элемента</b></p>
2. <p><b>Получение элемента</b></p>
3. <p><b>Получение общего количества элементов в массиве</b></p>
4. <p><b>Получение последнего элемента и его удаление</b></p>

Ответ: 1, 2, 3

Комментарий: SD001003

**<p><b>Как создать массив, вмещающий в себя 5 элементов целочисленного типа? </b></p>**

1. <p><b>float arr[5] = {0};</b></p>
2. <p><b>int arr{5} = [0]; </b></p>
3. <p><b>int arr [4] = {0};</b></p>
4. <p><b>int arr[5] = {0};</b></p>

Ответ: 4

Комментарий: SD001004

**<p><b>Каким образом можно добавить ещё один элемент в простой массив? (arr – название массива, NUM – const, кол-во элементов массива, n – элемент, который нужно добавить) </b></p>**

1. <p><b>arr[NUM + 1] = n; </b></p>
2. <p><b>arr[NUM – 1] = n; </b></p>
3. <p><b>arr[NUM] = n; </b></p>
4. <p><b>Нет правильного ответа</b></p>

Ответ: 4

Комментарий: SD001005

**<p><b>Как вывести в консоль все элементы массива? (arr – название массива, NUM – const, кол-во элементов массива) </b></p>**

1. <p><b> for (int i = 0; i < NUM; i++) cout << arr[i] << endl; </b></p>
2. <p><b>for (int i = 0; i <= NUM; i++) cout << arr[i] << endl; </b></p>
3. <p><b>for (int i = 1; i <= NUM; i++) cout << arr[i] << endl; </b></p>
4. <p><b>for (int i = 1; i < NUM; i++) cout << arr[i] << endl; </b></p>

Ответ: 1

Комментарий: SD001006

**<p><b>Недостатки линейного массива: </b></p>**

1. <p><b>Структурированность </b></p>
2. <p><b>Неизменное количество объектов</b></p>
3. <p><b>Сложность в использовании</b></p>
4. <p><b>Нагрузка на систему в отличии от создание большого количества переменных</b></p>

Ответ: 2

Комментарий: SD001007

**<p><b>Как заполнить массив, вмещающий в себя 10 элементов? </b></p>**

1. <p><b>for(int i = 1; i < 10; i++) cin >> a[i];</b></p>
2. <p><b> for(int i = 0; i < 10; i++) cin >> a[i];</</b></p>
3. <p><b> for(int i = 0; i <= 10; i++) cin >> a[i];</ </b></p>
4. <p><b> for(int i = 1; i <= 10; i++) cin >> a[i];</</b></p>

Ответ: 2

Комментарий: SD001008

**<p><b>Результатом выполнения программы ниже будет:<br>int a[1];<br>for(int i = 0; i < 5; i++)<br>&ensp;cout <&lt; a[i];</b></p>**

1. <p><b> будет выведено 5 случайны чисел, хранящихся в памяти(“мусор”)</b></p>
2. <p><b>будет выведено 5 единиц </b></p>
3. <p><b>выведет ошибку</b></p>
4. <p><b>выведет 5 нулей</b></p>

Ответ: 1

Комментарий: SD001009

**<p><b>Операторы, используемые для выделения динамического массива и работы с ним: </b></p>**

1. <p><b>new[]</b></p>
2. <p><b>array</b></p>
3. <p><b>delete[]</b></p>
4. <p><b>length</b></p>

Ответ: 1, 3

Комментарий: SD001010

**<p><b>Может ли динамический массив содержать в себе одномерные массивы? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001011

**<p><b>Виды линейных структур данных с конечными точками: </b></p>**

1. <p><b>Динамический массив</b></p>
2. <p><b>Очередь с приоритетами</b></p>
3. <p><b>Стек</b></p>
4. <p><b>Очередь</b></p>

Ответ: 2, 3, 4

Комментарий: SD001012

**<p><b>Способы реализации стека: </b></p>**

1. <p><b>С помощью одномерного массива</b></p>
2. <p><b>С помощью связанного списка</b></p>
3. <p><b>С помощью класса объектно-ориентированного программирования </b></p>
4. <p><b>Нет правильных ответов</b></p>

Ответ: 1, 2, 3

Комментарий: SD001013

**<p><b>Что из себя представляют связные списки? </b></p>**

1. <p><b>область памяти, где могут последовательно храниться несколько значений</b></p>
2. <p><b>динамические структуры данных, в которых однотипные элементы связаны между собой</b></p>
3. <p><b>имена кусочков памяти, которые могут хранить информацию</b></p>
4. <p><b>многократное прохождение по одному и тому же коду программы</b></p>

Ответ: 2

Комментарий: SD001014

**<p><b>Типы связных списков: </b></p>**

1. <p><b>Двунаправленный</b></p>
2. <p><b>Кольцевой</b></p>
3. <p><b>Перекрёстный</b></p>
4. <p><b>Однонаправленный</b></p>

Ответ: 1, 4

Комментарий: SD001015

**<p><b>Самый простой вид связных списков? </b></p>**

1. <p><b>Линейный двусвязной список</b></p>
2. <p><b>Кольцевой список</b></p>
3. <p><b>Линейный односвязный список</b></p>
4. <p><b>Многосвязный список</b></p>

Ответ: 3

Комментарий: SD001016

**<p><b>Одними из операций, которые включают в себя списки, являются: </b></p>**

1. <p><b>добавление нового звена списка</b></p>
2. <p><b>сортировка списка</b></p>
3. <p><b>создание ведущего звена</b></p>
4. <p><b>перестановка всех его звеньев в обратном порядке</b></p>

Ответ: 1, 2, 3, 4

Комментарий: SD001017

**<p><b>Недостатки односвязных списков: </b></p>**

1. <p><b>усложнение взаимодействия операций поиска и удаления</b></p>
2. <p><b>простота операций </b></p>
3. <p><b>наличие только одной связи снижает надёжность хранения данных </b></p>
4. <p><b>меньший расход памяти</b></p>

Ответ: 1, 3

Комментарий: SD001018

**<p><b>Значение, которое содержит в себе нулевой показатель в С++, это: </b></p>**

1. <p><b>NIL</b></p>
2. <p><b>NUL</b></p>
3. <p><b>0</b></p>
4. <p><b>NULL</b></p>

Ответ: 3, 4

Комментарий: SD001019

**<p><b>Пример структуры, описывающей узел связанного списка: </b></p>**

1. <p><b>struct Node <br />

{<br />

int data ; <br />

Node \*next; <br />

};</b></p>

1. <p><b>class Node<br />

{<br />

public:<br />

int firstNode;<br />

int secondNode;<br />

int thirdNode;<br />

};</b></p>

1. <p><b>struct NodeStruct<br />

{<br />

int first;<br />

int second;<br />

int third;<br />

};</b></p>

Ответ: 1

Комментарий: SD001020

**<p><b>На чем построен стек? </b></p>**

1. <p><b>На массивах</b></p>
2. <p><b>На классах</b></p>
3. <p><b>На связных списках</b></p>
4. <p><b>На циклах</b></p>

Ответ: 3

Комментарий: SD001021

**<p><b>Правда ли, что элементы стека извлекаются из него в обратном порядке тому, в котором они добавлялись? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001022

**<p><b>Сокращенное наименование принципа работы стека: </b></p>**

1. <p><b>FLIP</b></p>
2. <p><b>LIFO</b></p>
3. <p><b>LOFI</b></p>
4. <p><b>FILO</b></p>

Ответ: 2

Комментарий: SD001023

**<p><b>Каким образом можно обратиться к элементы, находящемуся в середина стека?: </b></p>**

1. <p><b>по его собственному номеру</b></p>
2. <p><b>по его номеру относительно последнего занесенного элемента</b></p>
3. <p><b>по его номеру относительно первого занесенного элемента</b></p>
4. <p><b>Нет правильного ответа</b></p>

Ответ: 4

Комментарий: SD001024

**<p><b>Преимущества стека, реализованного с помощью односвязного списка: </b></p>**

1. <p><b>добавление элемента всегда занимает одно и то же время</b></p>
2. <p><b>элементы могут располагаться в памяти разреженно </b></p>
3. <p><b>возможность перемещения по стеку лишь в одном направлении</b></p>

Ответ: 1

Комментарий: SD001025

**<p><b> Сокращенное наименование принципа работы очереди: </b></p>**

1. <p><b>LOLO</b></p>
2. <p><b>LOLI</b></p>
3. <p><b>LILO</b></p>
4. <p><b>QUEUE</b></p>

Ответ: 3

Комментарий: SD001026

**<p><b>Виды очередей: </b></p>**

1. <p><b>Простая очередь</b></p>
2. <p><b>Смешанная очередь</b></p>
3. <p><b>Кольцевая очередь</b></p>
4. <p><b>Очередь с приоритетами</b></p>

Ответ: 1, 3, 4

Комментарий: SD001027

**<p><b>Суть кольцевой очереди: </b></p>**

1. <p><b>Элемент, который выходит из очереди перемещается в её конец</b></p>
2. <p><b>Элемент, которые заходит в очередь перемещается в её конец(реверсия) </b></p>
3. <p><b>После обращения к последнему элементу, вы обращаетесь снова к первому</b></p>
4. <p><b>После обращения к первому элементу он меняется с последний, второй с предпоследним и т.д. </b></p>

Ответ: 1

Комментарий: SD001028

**<p><b>Примеры кольцевой очереди на практике: </b></p>**

1. <p><b>Очередь у кассы в магазине</b></p>
2. <p><b>Движение трамвая по кольцевому маршруту</b></p>
3. <p><b>Очередь событий для их отработки в Windows</b></p>
4. <p><b>Принцип зарядки и разрядки обоймы пистолета</b></p>

Ответ: 2, 3

Комментарий: SD001029

**<p><b>Какой элемент выйдет последним из очереди с приоритетами? </b></p>**

1. <p><b>Тот, что был добавлен первым</b></p>
2. <p><b>Тот, что имеет наивысший приоритет</b></p>
3. <p><b>Тот, что был добавлен последним</b></p>
4. <p><b>Тот, что имеет наименьший приоритет</b></p>

Ответ: 4

Комментарий: SD001030

**<p><b>Что необходимо указывать при объявлении структуры дерева? <p><b>**

1. <p><b>потомки </b></p>
2. <p><b>число ветвей</b></p>
3. <p><b>тип данных потомков</b></p>
4. <p><b>указатель на предка</b></p>

Ответ: 1

Комментарий: SD001031

**<p><b>Способ преобразования блока данных в число – это</b></p>**

1. <p><b>Индексирование</b></p>
2. <p><b>Хеширование</b></p>
3. <p><b>Маркировка</b></p>
4. <p><b>Сортировка</b></p>

Ответ: 2

Комментарий: SD001032

**<p><b>Как называется узел дерева, не имеющий потомков? </b></p>**

1. <p><b>последний </b></p>
2. <p><b>терминальный </b></p>
3. <p><b>окончательный </b></p>
4. <p><b>финальный</b></p>

Ответ: 2

Комментарий: SD001033

**<p><b>У всех узлов левого поддерева произвольного узла X бинарного дерева поиска значения ключей данных ПРОПУСК значения ключа данных самого узла X. Заполните пропуск</b></p>**

1. <p><b>больше </b></p>
2. <p><b>меньше </b></p>
3. <p><b>больше или равно </b></p>
4. <p><b>меньше или равно</b></p>

Ответ: 2

Комментарий: SD001034

**<p><b>Какое максимальное количество узлов-потомков у каждого узла кучи? </b></p>**

1. <p><b>2</b></p>
2. <p><b>8</b></p>
3. <p><b>64</b></p>
4. <p><b>неограниченное</b></p>

Ответ: 4

Комментарий: SD001035

**<p><b>С помощью какого типа данных обычно реализуется куча? </b></p>**

1. <p><b>дерево</b></p>
2. <p><b>класс</b></p>
3. <p><b>динамический массив</b></p>
4. <p><b>стек</b></p>

Ответ: 1

Комментарий: SD001036

**<p><b>Из скольки объектов состоит элемент словаря? </b></p>**

1. <p><b>1</b></p>
2. <p><b>2</b></p>
3. <p><b>4</b></p>
4. <p><b>число неограниченно</b></p>

Ответ: 2

Комментарий: SD001037

**<p><b>Может ли соответствовать одно и то же значение двум разным ключам словаря? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001038

**<p><b>Дано пустое множество mySet. Что выведет данная программа? <br>mySet.insert('I');<br>mySet.insert('n');<br>mySet.insert('f');<br>mySet.insert('i');<br>mySet.insert('n'); <br>copy( mySet.begin(), mySet.end(), ostream\_iterator<char>(cout, " "));</b></p>**

1. <p><b>I n f i n </b></p>
2. <p><b>I f i n</b></p>
3. <p><b>f i n</b></p>
4. <p><b>I f i n n</b></p>

Ответ: 2

Комментарий: SD001039

**<p><b>Может ли множество содержать дубликаты? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD001040

**<p><b>Чему равно количество строк матрицы смежности графа? </b></p>**

1. <p><b>Количеству рёбер графа</b></p>
2. <p><b>Количеству вершин графа</b></p>
3. <p><b>Максимальной степени входа вершины графа</b></p>
4. <p><b>Минимальной степени входа вершины графа</b></p>

Ответ: 2

Комментарий: SD001041

**<p><b>В чём отличие очереди от дека? </b></p>**

1. <p><b>В очередь можно только добавлять элементы</b></p>
2. <p><b>В очереди с каждого конца можно либо только удалять, либо только добавлять элементы</b></p>
3. <p><b>В дек можно добавлять/удалять элементы только с одного конца</b></p>
4. <p><b>Они предназначены для разных типов данных</b></p>

Ответ: 2

Комментарий: SD001042

**<p><b>Почему сложных типов данных нет в стандартных библиотеках? </b></p>**

1. <p><b>они занимают слишком много места, это усложняет подключение</b></p>
2. <p><b>нет универсального решения, проще написать самому</b></p>
3. <p><b>их реализация требует подключения дополнительных библиотек с необходимыми функциями</b></p>
4. <p><b>нет единого мнения, как они должны быть реализованы</b></p>

Ответ: 2

Комментарий: SD001043

**<p><b>Тип данных, который удобно использовать при необходимости добавлять и удалять элементы в середине</b></p>**

1. <p><b>дек</b></p>
2. <p><b>одномерный массив</b></p>
3. <p><b>связанный список</b></p>
4. <p><b>очередь</b></p>

Ответ: 3

Комментарий: SD001044

**<p><b>Какой тип данных самый универсальный? </b></p>**

1. <p><b>одномерный массив</b></p>
2. <p><b>граф</b></p>
3. <p><b>динамический массив</b></p>
4. <p><b>куча</b></p>

Ответ: 3

Комментарий: SD001045

**<p><b>С помощью какого типа данных лучше всего решать численные задачи типа «Ханойская башня»? </b></p>**

1. <p><b>очередь</b></p>
2. <p><b>стек</b></p>
3. <p><b>динамический массив</b></p>
4. <p><b>дек</b></p>

Ответ: 2

Комментарий: SD001046

**<p><b>Возможно ли реализовать кучу при помощи одномерного массива? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001047

**<p><b>Пустой указатель – это</b></p>**

1. <p><b>NULL</b></p>
2. <p><b>nullptr</b></p>
3. <p><b>null</b></p>
4. <p><b>\_ptr</b></p>

Ответ: 2

Комментарий: SD001048

**<p><b>Как в памяти хранится связный список? </b></p>**

1. <p><b>последовательно</b></p>
2. <p><b>распределённо</b></p>

Ответ: 2

Комментарий: SD001049

**<p><b>Укажите неверное утверждение для полного бинарного дерева</b></p>**

1. <p><b>у каждой вершины не более двух потомков</b></p>
2. <p><b>заполнение вершин идёт сверху вниз</b></p>
3. <p><b>в пределах одного уровня заполнение идёт справа налево</b></p>
4. <p><b>является кучей</b></p>

Ответ: 3

Комментарий: SD001050

**<p><b>Какой индекс будет иметь левый потомок вершины i двоичной кучи, хранимой в виде одномерного массива? </b></p>**

1. <p><b>i + 1</b></p>
2. <p><b>i + 2</b></p>
3. <p><b>2 \* i + 1</b></p>
4. <p><b>2 \* i + 2</b></p>

Ответ: 3

Комментарий: SD001051

**<p><b>Где может храниться самый большой элемент двоичной кучи? </b></p>**

1. <p><b>на левом краю нижнего ряда</b></p>
2. <p><b>на правом краю нижнего ряда</b></p>
3. <p><b>на верхнем ряду</b></p>
4. <p><b>в любом месте</b></p>

Ответ: 3

Комментарий: SD001052

**<p><b>При добавлении в какую часть динамического массива он показывает наилучшую производительность? </b></p>**

1. <p><b>начало</b></p>
2. <p><b>середина</b></p>
3. <p><b>конец</b></p>
4. <p><b>не имеет значения</b></p>

Ответ: 3

Комментарий: SD001053

**<p><b>Что делает данная программа?<br>while (ptr\_p) {<br>&ensp;cout << p->val << " ";<br>&ensp;p = p->next; <br>}</b></p>**

1. <p><b>Выводит количество элементов списка</b></p>
2. <p><b>Проверяет, нет ли в списке пустых значений</b></p>
3. <p><b>Выводит на экран весь список</b></p>
4. <p><b>Очищает список</b></p>

Ответ: 3

Комментарий: SD001054

**<p><b>Структура FIFO – это…</b></p>**

1. <p><b>очередь</b></p>
2. <p><b>стек</b></p>
3. <p><b>дек</b></p>
4. <p><b>динамический массив</b></p>

Ответ: 1

Комментарий: SD001055

**<p><b>Как называется граф, каждой вершине которого поставлено в соответствие конкретное числовое значение</b></p>**

1. <p><b>Числовой</b></p>
2. <p><b>Определённый</b></p>
3. <p><b>Взвешенный</b></p>
4. <p><b>Нормальный</b></p>

Ответ: 3

Комментарий: SD001056

**<p><b>Как называется граф, каждой вершине которого поставлено в соответствие конкретное числовое значение</b></p>**

1. <p><b>Матрица смежности</b></p>
2. <p><b>Матрица инцидентности</b></p>
3. <p><b>Дерево</b></p>
4. <p><b>Список ребер</b></p>

Ответ: 1, 2, 4

Комментарий: SD001057

**<p><b>Какое минимальное количество путей между каждой парой вершин существует в связном графе?</b></p>**

1. <p><b>0</b></p>
2. <p><b>1</b></p>
3. <p><b>2</b></p>
4. <p><b>3</b></p>

Ответ: 2

Комментарий: SD001058

**<p><b>Можно ли применять алгоритм поиска в ширину для неориентированного графа?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001059

**<p><b>Какой тип данных эффективнее всего использовать для реализации поиска в ширину в графе?</b></p>**

1. <p><b>Стек</b></p>
2. <p><b>Дерево</b></p>
3. <p><b>Динамический массив</b></p>
4. <p><b>Очередь</b></p>

Ответ: 4

Комментарий: SD001060

<p><b> База данных - это:</b></p>

1.<p><b> специальным образом организованная и хранящаяся на внешнем носителе совокупность взаимосвязанных данных о некотором объекте</b></p>

2.<p><b> произвольный набор информации</b></p>

3.<p><b> совокупность программ для хранения и обработки больших массивов информации </b></p>

4.<p><b> интерфейс, поддерживающий наполнение и манипулирование данными </b></p>

Ответ: 1

Комментарий: SD005001

<p><b>Какой функционал предоставляет хедер fstream? </b></p>

1.<p><b>Фильтрация файлов. </b></p>

2.<p><b>Считывание данных из файлов и запись в файл. </b></p>

3.<p><b>Редактирование файлов. </b></p>

4.<p><b>Поиск файлов. </b></p>

Ответ:2

Комментарий: SD005002

<p><b>Какая функция считывает символы из stdin и помещает их в массив символов? </b></p>

1.<p><b>tellg()</b></p>

2.<p><b>is\_open()</b></p>

3.<p><b>seekg()</b></p>

4.<p><b>gets()</b></p>

Ответ:4

Комментарий: SD005003

<p><b>Какой класс предоставляет возможности для чтения ? </b></p>

1.<p><b>ifstream</b></p>

2.<p><b>olstream</b></p>

3.<p><b>sestream</b></p>

4.<p><b>usstream</b></p>

Ответ:1

Комментарий: SD005004

<p><b>Какой класс предоставляет возможности для записи? </b></p>

1.<p><b>ifstream</b></p>

2.<p><b>olstream</b></p>

3.<p><b>ofstream</b></p>

4.<p><b>usstream</b></p>

Ответ:3

Комментарий: SD005005

<p><b>Какими способами можно открыть БД? </b></p>

1.<p><b>open()</b></p>

2.<p><b>is\_open()</b></p>

3.<p><b>Указать путь к нему в конструкторе. </b></p>

4.<p><b>Варианты 1,3. </b></p>

Ответ:4

Комментарий: SD005006

<p><b>Если файл не открылся для проверки…</b></p>

1.<p><b>Использовать is\_open()</b></p>

2.<p><b>Проверить переменную файла в логическом выражении</b></p>

3.<p><b>Использовать open()</b></p>

4.<p><b>Вариант 1,2. </b></p>

Ответ.4

Комментарий: SD005007

<p><b>С помощью какого оператора можно организовать считывание</b></p>

1.<p><b>\*</b></p>

2.<p><b> +- </b></p>

3.<p><b> >> </b></p>

4.<p><b>\*</b></p>

Ответ:3

Комментарий: SD005008

<p><b>Какая функция возвращает текущую позицию для чтения? </b></p>

1.<p><b>tellg()</b></p>

2.<p><b>seekg()</b></p>

3.<p><b>getline()</b></p>

4.<p><b>eof()</b></p>

Ответ:1

Комментарий: SD005009

<p><b>Какая функция проверяет не достигнут ли конец БД? Т.е. можно ли из него продолжить чтение. </b></p>

1.<p><b>seekg()</b></p>

2.<p><b>eof()</b></p>

3.<p><b>getline()</b></p>

4.<p><b>tellg()</b></p>

Ответ:2

Комментарий: SD005010

<p><b>Какая функция считывает не одно слово, а целую строку до встречи с первым переводом на новую строку? </b></p>

1.<p><b>getline()</b></p>

2.<p><b>cin()</b></p>

3.<p><b>tolower()</b></p>

4.<p><b>tellg()</b></p>

Ответ:1

Комментарий: SD005011

<p><b>Какая функция используется для закрытия файла? </b></p>

1.<p><b>open()</b></p>

2.<p><b>tellg()</b></p>

3.<p><b>close()</b></p>

4.<p><b>eof()</b></p>

Ответ:3

Комментарий: SD005012

<p><b>Класс ,для установки режима открытия файлов для чтения. </b></p>

1.<p><b>**ios\_base::in**</b></p>

2.<p><b>**ios\_base::out**</b></p>

3.<p><b>**ios\_base::app**</b></p>

4.<p><b>**ios\_base::trunc**</b></p>

Ответ:1

Комментарий: SD005013

<p><b>Класс ,для установки режима открытия файлов для записи. </b></p>

1.<p><b>**ios\_base::in**</b></p>

2.<p><b>**ios\_base::out**</b></p>

3.<p><b>**ios\_base::app**</b></p>

4.<p><b>**ios\_base::trunc**</b></p>

Ответ:2

Комментарий: SD005014

<p><b>Класс ,для установки режима открытия файлов для записи в конце файла. </b></p>

1.<p><b>**ios\_base::in**</b></p>

2.<p><b>**ios\_base::out**</b></p>

3.<p><b>**ios\_base::app**</b></p>

4.<p><b>**ios\_base::trunc**</b></p>

Ответ:3

Комментарий: SD005016

<p><b> Какой из вариантов не является функцией СУБД?</b></p>

1.<p><b> реализация языков определения и манипулирования данными</b></p>

2. <p><b>. обеспечение пользователя языковыми средствами манипулирования данными </b></p>

3. <p><b> защита и целостность данных </b></p>

4. <p><b> координация проектирования, реализации и ведения БД </b></p>

Ответ: 4

Комментарий: SD005017

</b></p>Какой метод производит установку текущей позиции в нужную, указываемую числом? </b></p>

</b></p>**eof()**</b></p>

</b></p>**get()**</b></p>

</b></p>**tellp()**</b></p>

</b></p>**seekg()**</b></p>

Ответ:4

Комментарий: SD005018

<p><b> Как называется набор хранимых записей одного типа?</b></p>

1. <p><b> хранимый файл </b></p>

2. <p><b> представление базы данных </b></p>

3. <p><b> логическая таблица базы данных </b></p>

4. <p><b> физическая таблица базы данных </b></p>

Ответ: 1

Комментарий: SD005019

</b></p>Как отсчитать новую позицию с конца файла с помощью метода seekg()?</b></p>

1.</b></p>ios\_base::end</b></p>

2.</b></p>ios\_base::beg</b></p>

3.</b></p>ios\_base::start</b></p>

4.</b></p>ios\_base::cur</b></p>

Ответ:1

Комментарий: SD005020

</b></p>Функция считывает count байт из файла…</b></p>

1.</b></p>read()</b></p>

2.</b></p>write()</b></p>

3.</b></p>gets()</b></p>

4.</b></p>tellg()</b></p>

Ответ:1

Комментарий: SD005021

<p><b> К чему приведет отсутствие логической и физической независимости данных?</b></p>

1. <p><b> к необходимости изменения прикладных программ при изменении физического представления базы данных</b></p>

2. <p><b> к большей достоверности данных</b></p>

3. <p><b> к возможному изменению физического представления данных при изменении прикладных программ </b></p>

4. <p><b> Вариант 1 3. </b></p>

Ответ:4

Комментарий: SD005022

</b></p>Какая функция переписывает count байт из буфера? </b></p>

1.</b></p>write()</b></p>

2.</b></p>read()</b></p>

3.</b></p>**tellp()**</b></p>

4.</b></p>tellg()</b></p>

Ответ:1

Комментарий: SD005023

</b></p>Класс ,для установки режима открытия файлов в бинарном. </b></p>

1.</b></p>**ios\_base::in**</b></p>

2.</b></p>**ios\_base::out**</b></p>

3.</b></p>**ios\_base::binary**</b></p>

4.</b></p>**ios\_base::trunc**</b></p>

**Ответ:3**

Комментарий: SD005024

<p><b> Какая наименьшая единица хранения данных в БД?</b></p>

1. <p><b> хранимое поле </b></p>

2. <p><b> хранимый файл</b></p>

3. <p><b> хранимая запись </b></p>

4. <p><b> хранимый байт </b></p>

Ответ: 1

Комментарий: SD005025

<p><b> Верно ли, что для считывания данных из БД, необходимо: описать переменную типа **ofstream**. </b></p>

1.<p><b> Да</b></p>

2.<p><b> Нет</b></p>

Ответ:2

Комментарий: SD005026

<p><b> Верно ли, что для записи данных из БД, необходимо:описать переменную типа **ofstream**. </b></p>

1.<p><b> Да</b></p>

2.<p><b> Нет</b></p>

Ответ:1

Комментарий: SD005027

<p><b> Причинами низкой эффективности проектируемых БД могут быть:</b></p>

1. <p><b> количество подготовленных документов </b></p>

2. <p><b> большая длительность процесса структурирования </b></p>

3. <p><b> недостаточно глубокий анализ требований </b></p>

4. <p><b> Вариант 2 3</b></p>

Ответ:4

Комментарий: SD005028

<p><b> Обязательна ли функция close()?</b></p>

1.<p><b> Да </b></p>

2.<p><b> Нет </b></p>

3.<p><b> Такой функции не существует </b></p>

Ответ:2

Комментарий: SD005029

<p><b>На какой хедер похож **fstream**</b></p>

1. <p><b>**iostream**</b></p>

2. <p><b>string</b></p>

3. <p><b>iterator</b></p>

Ответ:1

Комментарий: SD005030

**<p><b> База данных - это:</b></p>**

1. <p><b> специальным образом организованная и хранящаяся на внешнем носителе совокупность взаимосвязанных данных о некотором объекте</b></p>
2. <p><b> произвольный набор информации</b></p>
3. <p><b> совокупность программ для хранения и обработки больших массивов информации </b></p>
4. <p><b> интерфейс, поддерживающий наполнение и манипулирование данными </b></p>

Ответ: 1

Комментарий: SD005031

<**p><b>В записи файла реляционной базы данных (БД) может содержаться:</b></p>**

1. <p><b> исключительно однородная информация (данные только одного типа)</b></p>
2. <p><b> только текстовая информация </b></p>
3. <p><b> неоднородная информация (данные разных типов) </b></p>
4. <p><b> исключительно числовая информация </b></p>

Ответ: 3

Комментарий: SD005032

**<p><b>Предположим, что некоторая база данных содержит поля ФАМИЛИЯ, ГОД РОЖДЕНИЯ, ДОХОД. При поиске по условию ГОД РОЖДЕНИЯ> 1958 AND ДОХОД<3500 будут найдены фамилии лиц:</b></p>**

1. <p><b> имеющих доход не менее 3500, и старше тех, кто родился в 1958 году</b></p>
2. <p><b> имеющих доход менее 3500, ипи тех, кто родился е 1958 году и позже </b></p>
3. <p><b> имеющих доход менее 3500, и родившихся в 1958 году и позже </b></p>
4. <p><b> имеющих доход менее 3500, и родившихся в 1959 году и позже </b></p>

Ответ: 4

Комментарий: SD005033

**<p><b> Какой из вариантов не является функцией СУБД?</b></p>**

1. <p><b> реализация языков определения и манипулирования данными</b></p>
2. <p><b>. обеспечение пользователя языковыми средствами манипулирования данными </b></p>
3. <p><b> защита и целостность данных </b></p>
4. <p><b> координация проектирования, реализации и ведения БД </b></p>

Ответ: 4

Комментарий: SD005034

**<p><b> Система управления базами данных представляет собой программный продукт, входящий в состав: </b></p>**

1. <p><b> прикладного программного обеспечения </b></p>
2. <p><b> уникального программного обеспечения </b></p>
3. <p><b> системного программного обеспечения </b></p>
4. <p><b> систем программирования </b></p>

Ответ: 4

Комментарий: SD005035

**<p><b> Какая наименьшая единица хранения данных в БД?</b></p>**

1. <p><b> хранимое поле </b></p>
2. <p><b> хранимый файл</b></p>
3. <p><b> хранимая запись </b></p>
4. <p><b> хранимый байт </b></p>

Ответ: 1

Комментарий: SD005036

**<p><b> Что обязательно должно входить в СУБД?</b></p>**

1. <p><b> процессор языка запросов </b></p>
2. <p><b> командный интерфейс </b></p>
3. <p><b> визуальная оболочка </b></p>
4. <p><b> система помощи </b></p>

Ответ: 1, 2

Комментарий: SD005037

**<p><b> Перечислите преимущества централизованного подхода к хранению и управлению данными </b></p>**

1. <p><b> возможность общего доступа к данным </b></p>
2. <p><b> поддержка целостности данных </b></p>
3. <p><b> соглашение избыточности </b></p>
4. <p><b> сокращение противоречивости </b></p>

Ответ: 1, 2, 3, 4

Комментарий: SD005038

**<p><b> Предположим, что некоторая база данных описывается следующим перечнем записей: <br> 1 Иванов, 1956, 2400, <br> 2 Сидоров, 1957, 5300, <br> 3 Петров, 1956, 3600, <br>4 Козлов, 1952, 1200. <br> Какие из записей этой БД поменяются местами при сортировке по возрастанию, произведенной по первому полю:</b></p>**

1. <p><b> 3 и 4</b></p>
2. <p><b> 2 и З </b></p>
3. <p><b> 2 и 4 </b></p>
4. <p><b> 1 и 3 </b></p>

Ответ: 3

Комментарий: SD005039

**<p><b> Структура файла реляционной базы данным (БД) меняется:</b></p>**

1. <p><b> при изменении любой записи </b></p>
2. <p><b> при уничтожении всех записей </b></p>
3. <p><b> при удалении любого поля </b></p>
4. <p><b> при добавлении одной или нескольких записей </b></p>

Ответ: 3

Комментарий: SD005040

**<p><b> Как называется набор хранимых записей одного типа?</b></p>**

1. <p><b> хранимый файл </b></p>
2. <p><b> представление базы данных </b></p>
3. <p><b> логическая таблица базы данных </b></p>
4. <p><b> физическая таблица базы данных </b></p>

Ответ: 1

Комментарий: SD005041

**<p><b> Причинами низкой эффективности проектируемых БД могут быть:</b></p>**

1. <p><b> количество подготовленных документов </b></p>
2. <p><b> большая длительность процесса структурирования </b></p>
3. <p><b> скорость работы программных средств </b></p>
4. <p><b> недостаточно глубокий анализ требований </b></p>

Ответ: 2, 4

Комментарий: SD005042

**<p><b Основные требования, предъявляемые к базе данных?</b></p>**

1. <p><b> адаптивность и расширяемость </b></p>
2. <p><b> восстановление данных после сбоев </b></p>
3. <p><b> распределенная обработка данных </b></p>
4. <p><b> контроль за целостностью данных </b></p>

Ответ: 1, 2, 3, 4

Комментарий: SD005043

**<p><b> Система управления базами данных (СУБД) - это?</b></p>**

1. <p><b> это совокупность баз данных </b></p>
2. <p><b> это совокупность нескольких программ предназначенных для совместного использования БД многими пользователями </b></p>
3. <p><b> состоит из совокупности файлов расположенных на одной машине </b></p>
4. <p><b> это совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями</b></p>

Ответ: 4

Комментарий: SD005044

**<p><b> База данных — это средство для </b></p>**

1. <p><b> хранения, поиска и упорядочения данных </b></p>
2. <p><b> поиска данных </b></p>
3. <p><b> хранения данных </b></p>
4. <p><b> сортировки данных </b></p>

Ответ: 1

Комментарий: SD005045

**<p><b> Что входит в функции СУБД? </b></p>**

1. <p><b> создание структуры базы данных </b></p>
2. <p><b> загрузка данных в базу данных </b></p>
3. <p><b> предоставление возможности манипулирования данными </b></p>
4. <p><b> проверка корректности прикладных программ, работающих с базой данных </b></p>

Ответ: 1, 2, 3

Комментарий: SD005046

**<p><b> Основные средства СУБД для работы пользователя с базой данных:</b></p>**

1. <p><b> язык запросов </b></p>
2. <p><b> графический интерфейс </b></p>
3. <p><b> алгоритмический язык Паскаль </b></p>
4. <p><b> разрабатываемые пользователем программы </b></p>

Ответ: 1, 2

Комментарий: SD005047

**<p><b> Что дает логическая и физическая независимость данных?</b></p>**

1. <p><b> изменение прикладных программ не приводит к изменению физического представления базы данных </b></p>
2. <p><b> изменение программ СУБД не приводит к изменению физического представления данных </b></p>
3. <p><b> изменение физического представления данных не приводят к изменению прикладных программ </b></p>

Ответ:1, 3

Комментарий: SD005048

**<p><b> При каких условиях система меняет данные в базе данных?</b></p>**

1. <p><b> по завершению транзакции </b></p>
2. <p><b> по указанию администратора </b></p>
3. <p><b> по оператору модификации данных </b></p>

Ответ: 1

Комментарий: SD005049

**<p><b> Какие средства используются для синхронизации?</b></p>**

1. <p><b> блокировки </b></p>
2. <p><b> транзакции </b></p>
3. <p><b> пароли </b></p>
4. <p><b> описание полномочий </b></p>

Ответ: 1

Комментарий: SD005050

**<p><b>то входит в понятие банка данных? </b></p>**

1. <p><b> база данных</b></p>
2. <p><b> прикладные программы работы с базой данных</b></p>
3. <p><b> СУБД</b></p>
4. <p><b> компьютеры с базой данных</b></p>

Ответ: 1, 3

Комментарий: SD005051

**<p><b>  Для чего предназначена СУБД?</b></p>**

1. <p><b>для создания базы данных </b></p>
2. <p><b> для ведения базы данных</b></p>
3. <p><b> для использования базы данных</b></p>
4. <p><b>для разработки прикладных программ </b></p>

Ответ: 1, 2, 3

Комментарий: SD005052

**<p><b>  К чему приведет отсутствие логической и физической независимости данных?</b></p>**

1. <p><b> к необходимости изменения прикладных программ при изменении физического представления базы данных</b></p>
2. <p><b> к большей достоверности данных</b></p>
3. <p><b> к возможному изменению физического представления данных при изменении прикладных программ </b></p>
4. <p><b>к более эффективному взаимодействию пользователей с базой данных </b></p>

Ответ: 1, 3

Комментарий: SD005053

**<p><b>Поле базы данных</b></p>**

1. <p><b>это столбец таблицы, содержащий значения определенного свойства </b></p>
2. <p><b> это таблица, содержащая значения определенного свойства</b></p>
3. <p><b>это последнее значение в базе данных </b></p>
4. <p><b> это столбец таблицы, содержащий все типы данных, используемых в базе данных </b></p>

Ответ: 1

Комментарий: SD005054

**<p><b> С помощью системы управления базами данных пользователь может ...:</b></p>**

1. <p><b> Устанавливать защиту базы данных</b></p>
2. <p><b> Создавать структуру базы данных</b></p>
3. <p><b>Просматривать веб страницы</b></p>
4. <p><b> Выполнять сортировку данных</b></p>

Ответ: 2, 4

Комментарий: SD005055

**<p><b> Основные цели обеспечения логической и физической целостности базы данных?</b></p>**

1. <p><b>защита от неправильных действий прикладного программиста </b></p>
2. <p><b> защита от неправильных действий администратора баз данных</b></p>
3. <p><b> защита от возможных ошибок ввода данных</b></p>
4. <p><b> защита от машинных сбоев</b></p>

Ответ: 3, 4

Комментарий: SD005056

**<p><b>Что не входит в функции СУБД? </b></p>**

1. <p><b>создание структуры базы данных </b></p>
2. <p><b>загрузка данных в базу данных </b></p>
3. <p><b>проверка корректности прикладных программ, работающих с базой данных </b></p>
4. <p><b>обеспечение логической и физической независимости данных </b></p>

Ответ: 3

Комментарий: SD005057

**<p><b>  Основные требования, побуждающие пользователя к использованию СУБД:</b></p>**

1. <p><b> необходимость представления средств организации данных прикладной программе</b></p>
2. <p><b> большой объем данных в прикладной программе</b></p>
3. <p><b> большой объем сложных математических вычислений</b></p>
4. <p><b>необходимость решения ряда задач с использованием общих данных </b></p>

Ответ: 4

Комментарий: SD005058

**<p><b>Есть ли какая-либо информация в таблице, в которой нет полей?</b></p>**

1. <p><b>Содержит информацию о структуре базы данных </b></p>
2. <p><b>Не содержит никакой информации </b></p>
3. <p><b>Таблица без полей существовать не может </b></p>
4. <p><b> Содержит информацию о будущих записях</b></p>

Ответ: 1

Комментарий: SD005059

**<p><b> Как соотносятся понятия база данных и банк данных?</b></p>**

1. <p><b> одно и то же</b></p>
2. <p><b> база данных включает банк данных</b></p>
3. <p><b>банк данных включает базу данных </b></p>
4. <p><b> не связанные понятия</b></p>

Ответ: 3

Комментарий: SD005060

### <p><b>БД, как правило, одновременно является и интегрированной, и общедоступной. Под "интегрированностью" имеется в виду то, что базу данных можно представить как</b></p>

1. <p><b> объединение нескольких отдельных неизменяемых файлов данных</b></p>
2. <p><b> объединение нескольких отдельных изменяемых файлов данных</b></p>
3. <p><b>объект, характеризующийся набором значений некоторой совокупности атрибутов</b></p>

Ответ: 2

Комментарий: SD005061

<p><b>Что же такое файл?</b></p>

1. <p><b>Именованный набор байтов, который может быть сохранен на некотором накопителе.</b></p>
2. <p><b>Набор байтов, который может быть сохранен на некотором накопителе.</b></p>
3. <p><b>Именованный набор нулей и единиц, который может быть сохранен на некотором накопителе.</b></p>

Ответ: 1

Комментарий: SD005062

<p><b>Задачи редактирования файлов в БД?</b></p>

1. <p><b>Исключение из файла фрагмента текста, получение информации о количестве символов текста.</b></p>
2. <p><b> Вставка в файл фрагмента текста, исключение из файла фрагмента текста, упорядочивание элементов файла по ключу.</b></p>
3. <p><b>Упорядочивание элементов файла по определенному ключу.</b></p>

Ответ: 2

Комментарий: SD005063

<p><b>Операцию замены можно выполнить с помощью</b></p>

1. <p><b>исключение из файла фрагмента текста и вставки в файл фрагмента текста.</b></p>
2. <p><b>упорядочивание элементов файла по определенному ключу и вставки в файл фрагмента текста.</b></p>
3. <p><b>исключение из файла фрагмента текста и замена фрагмента текста файла на другой фрагмент.</b></p>

Ответ: 1

Комментарий: SD005064

<p><b>Во время добавления данных в БД происходит аварийное закрытия файла. Но данные были сохраненный и не испорчены. По какой причине данные сохранились?</b></p>

1. <p><b>Утверждение не является верным. Данные были испорчены.</b></p>
2. <p><b>Пользователь успел применить сочетание клавиш ctrl+s.</b></p>
3. <p><b> в конце файла был записан *управляющий символ* конца файла.</b></p>

Ответ: 3

Комментарий: SD005065

<p><b>Работа с файлом осуществляется через (работает осуществляется при помощи консоли) </b></p>

1. <p><b>таблицы</b></p>
2. <p><b>периферийное устройство</b></p>
3. <p><b>буфер</b></p>

Ответ: 3

Комментарий: SD005066

<p><b>Что произойдет после записи символа в требуемом месте?</b></p>

1. <p><b>Файла больше считываться не будет.</b></p>
2. <p><b>Файл считается до конца.</b></p>
3. <p><b>Файл закроется (символ не добавиться).</b></p>

Ответ: 1

Комментарий: SD005067

<p><b>Таким образом, для выполнения операций исключения, вставки и замены фрагментов текста в файлах БД можно использовать?</b></p>

1. <p><b>Два варианта верные.</b></p>
2. <p><b>Использовать вспомогательный файл.</b></p>
3. <p><b>Пользоваться массивами указателей.</b></p>

Ответ: 1

Комментарий: SD005068

<p><b>Если *файл* целиком нельзя разместить в памяти программы то</b></p>

1. <p><b>файл приходится считывать *по* частям и осуществлять перенос информации из одной части в другую.</b></p>
2. <p><b>файл приходится считывать полностью и делить его на части</b></p>
3. <p><b>файл самостоятельно уменьшиться в размере и считается</b></p>

Ответ: 1

Комментарий: SD005069

<p><b>Чтобы вставить строку в середину файла, используя вспомогательный файл, необходимо проделать следующий алгоритм. Является ли алгоритм верным?

1. Открыть входной файл в режиме чтения и вспомогательный файл<br> в режиме записи (этот файл будет создан автоматически). <br>

2.Прочесть из входного файла текст до места вставки. <br>

3. Записать прочитанный фрагмент во вспомогательный файл. <br>

4 Записать во вспомогательный файл вставляемый фрагмент. <br>

5. Прочесть из входного файла остаточный фрагмент и записать его<br> во вспомогательный файл. <br>

6. Закрыть оба файла<br></b></p>

1. <p><b>нет</b></p>
2. <p><b>частично</b></p>
3. <p><b>да</b></p>

Ответ: 3

Комментарий: SD005070

<p><b>Редактирование файла на программном уровне это</b></p>

1. <p><b>изменения, производимые с элементами файла после завершения программы </b></p>
2. <p><b> изменения, производимые с элементами файла в процессе работы</b></p>
3. <p><b>изменения, производимые с файлом в процессе работы</b></p>

Ответ: 2

Комментарий: SD005071

<p><b>Как поможет Метод seekg() в редактировании файл?</b></p>

1. <p><b>Произведёт установку текущей позиции в нужную.</b></p>
2. <p><b>Проверяет не достигнут ли конец файла.</b></p>
3. <p><b>Закрывает файл.</b></p>

Ответ: 1

Комментарий: SD005072

<p><b>Что значит строчка file.seekg(30,ios\_base::beg);</b></p>

1. <p><b>Стать на 31-й байт</b></p>
2. <p><b>Стать на 31 байт с конца</b></p>
3. <p><b>Перепрыгнуть через 3 байта</b></p>

Ответ: 1

Комментарий: SD005073

<p><b>Выберите простой способ форматирования данных</b></p>

1. <p><b>cin,cout</b></p>
2. <p><b>printf scanf</b></p>
3. <p><b>оба варианта верны</b></p>

Ответ: 3

Комментарий: SD005074

<p><b>Считаются ли флаги форматированием частью редактирования файла?</b></p>

1. <p><b>да</b></p>
2. <p><b>нет</b></p>
3. <p><b>зависит от использования</b></p>

Ответ: 3

Комментарий: SD005075

<p><b>Флаги форматирования класса ios dec выполняет функцию</b></p>

1. <p><b>перевода значения в десятичную форму</b></p>
2. <p><b>перевода значения в восьмеричную форму</b></p>
3. <p><b>перевода значения в шестнадцатеричную форму</b></p>

Ответ: 1

Комментарий: SD005076

<p><b>Флаги форматирования класса ios right выполняет функцию</b></p>

1. <p><b>выводит первое слово</b></p>
2. <p><b>выравнивание по правому краю</b></p>
3. <p><b>выводит последние слово</b></p>

Ответ: 2

Комментарий: SD005077

<p><b>Флаги форматирования класса ios fixed выполняет функцию</b></p>

1. <p><b>Экспоненциальный вывод чисел с плавающей запятой</b></p>
2. <p><b>Фиксированный вывод чисел с плавающей запятой</b></p>
3. <p><b>нету верного утверждения</b></p>

Ответ: 2

Комментарий: SD005078

<p><b>Флаги форматирования класса ios skipws выполняет функцию</b></p>

1. <p><b>перевод в восьмеричную форму</b></p>
2. <p><b>показывать десятичную точку при выводе</b></p>
3. <p><b> пропуск пробелов при вводе</b></p>

Ответ: 3

Комментарий: SD005079

<p><b>Инструкции редактирования, которые вставляются прямо в поток.</b></p>

1. <p><b>Манипуляторы</b></p>
2. <p><b>Структуры</b></p>
3. <p><b>Классы</b></p>

Ответ: 1

Комментарий: SD005080

<p><b>Манипуляторы бывают двух видов:(укажите не верный вариант)</b></p>

1. <p><b>константным</b></p>
2. <p><b>с аргументом</b></p>
3. <p><b>без аргумента</b></p>

Ответ: 1

Комментарий: SD005081

<p><b>Манипуляторы ios setw() используется для</b></p>

1. <p><b>Устанавливает ширину поля для вывода данных</b></p>
2. <p><b>Устанавливает указанные флаги форматирования</b></p>
3. <p><b>Устанавливает указанные флаги форматирования</b></p>

Ответ: 1

Комментарий: SD005082

<p><b>Какая функция ostream устанавливает позицию в байтах файлового указателя относительно указанного места в файле?</b></p>

1. <p><b>seekp(position, seek\_dir)</b></p>
2. <p><b>tellp()</b></p>
3. <p><b>write()</b></p>

Ответ: 1

Комментарий: SD005083

<p><b>Какая функция ostream устанавливает SIZE символов из массива str в файл</b></p>

1. <p><b> seekp(position, seek\_dir)</b></p>
2. <p><b>write()</b></p>
3. <p><b>flush()</b></p>

Ответ: 2

Комментарий: SD005084

<p><b>Для чего создается массив с именем buff</b></p>

1. <p><b>для хранение измененных данных</b></p>
2. <p><b>для хранения всех данных</b></p>
3. <p><b>для передачи данных в другом порядке</b></p>

Ответ: 1

Комментарий: SD005085

<p><b>Для манипулирования данными, хранящимися в БД, используется группа операторов</b></p>

1. <p><b>sql</b></p>
2. <p><b>oracle</b></p>
3. <p><b>два ответа являются верными</b></p>

Ответ: 3

Комментарий: SD005086

<p><b>Файл test1.txt заполнены цифрами, все четные цифры записываться в test2.txt, а оставшиеся в test3.txt. Запишите строчку с ошибкой.<br>

void main(){<br>

ifstream ifs;<br>

ofstream ofs1; <br>

ofstream ofs2; <br>

string first\_file\_name = "C:\\programming\\test1.txt";<br>

string second\_file\_name = "C:\\programming\\test2.txt";<br>

string third\_file\_name = "C:\\programming\\test3.txt";<br>

ifs.open(first\_file\_name.c\_str());<br>

ofs1.open(second\_file\_name.c\_str());<br>

ofs2.open(third\_file\_name.c\_str());<br>

int buf; <br>

while (ifs >> buf) {<br>

if (buf % 2 != 0) <br>

ofs1 << buf; <br>

else<br>

ofs2 << buf; }<br>

ifs.close();<br>

ofs1.close();<br>

ofs2.close();}<br></b></p>

1. p><b>while (ifs >> buf)</b></p>
2. <p><b>if (buf % 2 != 0)</b></p>
3. <p><b>ifs.open(first\_file\_name.c\_str())</b></p>

Ответ: 2

Комментарий: SD005087

<p><b>Создан файл, содержащий десять цифр. цифры, отсортированные по возрастанию записываться в другой файл. Укажите строчку в которой есть ошибка. <br>

int main(){<br>

std::ifstream fin("number.txt");<br>

int m[10]; <br>

for (int i = 0; i < 10; i++) {<br>

fin >> m[i]; }<br>

bool flag = false; // сортировка пузырьком<br>

while (!flag) {<br>

flag = true; <br>

for (int i = 0; i < 10; i++) {<br>

if (m[i] > m[i + 1]) {<br>

int tmp = m[i]; <br>

m[i] = m[i + 1]; <br>

m[i + 1] = tmp; <br>

flag = false;}}} <br>

std::ofstream fout("result.txt"); <br>

for (int i = 0; i < 10; i++) {<br>

fout << m[i] << " ";}<br>

return 0; }<br></b></p>

1. <p><b>flag = true; </b></p>
2. <p><b>if (m[i] > m[i + 1]) </b></p>
3. <p><b>bool flag = false; </b></p>

Ответ: 3

Комментарий: SD005088

<p><b>Пример чтения из одного файла и записи в другой с переводом символов в верхний регистр. Укажите строчку, в которой есть ошибка. <br>

int main() {<br>

ifstream fin; <br>

ofstream fout; <br>

string fin\_name("input.txt"), fout\_name("output.txt"), content("");<br>

fin.open(fin\_name); <br>

fout.open(fout\_name); <br>

if (fin.is\_open() || fout.is\_open()) {<br>

getline(fin, content, '\0'); <br>

fin.close();<br>

transform(content.begin(), content.end(), content.begin(), ::toupper); <br>

fout << content; <br>

fout.close();}<br>

return 0;}<br></b></p>

1. <p><b>string fin\_name("input.txt"), fout\_name("output.txt"), …</b></p>
2. <p><b> if (fin.is\_open() || fout.is\_open())</b></p>
3. <p><b>fout.close();</b></p>

Ответ: 2

Комментарий: SD005089

<p><b>Дан файл f, компоненты которого являются целыми числами. Файл f содержит столько же отрицательных чисел, сколько и положительных. Используя вспомогательный файл h, переписать компоненты файла f в файл g так, чтобы в файле g:сначала шли положительные, потом отрицательные числа. Укажите строчку, в которой есть ошибка. <br>

int main(){<br>

int a = 0; <br>

ifstream fin; <br>

ofstream fout; <br>

fout.open("g.txt");<br>

fin.open("f.txt"); <br>

if (!(fin.is\_open())){<br>

Sleep(5000); <br>

return -1;} <br>

const int kol = 10; <br>

int A[kol]; <br>

for (int i(0); i < 0; i++)<br>

{<br>

a = 0; <br>

fin >> a; <br>

A[i] = a; <br>

cout << A[i] << " ";}<br>

int i = 0; <br>

for (int i(0); i < kol; i++){<br>

while (A[i] > 0) {<br>

cout << A[i] << " ";<br>

i++;<br>

if (i > 2) <br>

break;} <br>

while (A[i] < 0) {<br>

cout << A[i] << " ";<br>

i++;<br>

if (i > 2) break;}} <br></b></p>

1. <p><b>fout.open("g.txt”)</b></p>
2. <p><b>const int kol = 10;</b></p>
3. <p><b>for (int i(0); i < 0; i++)</b></p>

Ответ: 3

Комментарий: SD005090

**<p><b>** [**#include**](https://vk.com/im?sel=523705059&st=%23include)**<stdio.h> <br>   
Int remove(const char**[**\*filename**](https://vk.com/id20019287)**) <br>   
Что делает данная программа? </b></p>**

1. <p><b>Изменяет файл</b></p>
2. <p><b>Удаляет файл</b></p>
3. <p><b>Ничего из выше перечисленного</b></p>­­­

Ответ: 2

Комментарий: SD005091

**<p><b> Что делает функция remove()? </b></p>**

1. <p><b>Перемещает файл </b></p>
2. <p><b>Добавляет файл </b></p>
3. <p><b>Удаляет файл </b></p>

Ответ: 3

Комментарий: SD005092

**<p><b> Какое значение возвращает функция remove() при успешном удалении? </b></p>**

1. <p><b>0 </b></p>
2. <p><b>1 </b></p>
3. <p><b>ничего не возвращает</b></p>

Ответ: 1

Комментарий: SD005093

**<p><b>В чем отличие DELETE от NULL? </b></p>**

1. <p><b>DELETE удаляет данные,а NULL нет </b></p>
2. <p><b>DELETE удаляет данные, а NULL удаляет сам файл </b></p>
3. <p><b>Они ни чем не отличаются </b></p>

Ответ: 2

Комментарий: SD005094

**<p><b> Сколько основных способов удаления из БД существует? </b></p>**

1. <p><b>6 </b></p>
2. <p><b>5 </b></p>
3. <p><b>2 </b></p>

Ответ: 3

Комментарий: SD005095

**<p><b> Что делает параметр where? </b></p>**

1. <p><b>Удаляет все данные </b></p>
2. <p><b>Удаляет лишь те элементы, которые подходят под условие. </b></p>
3. <p><b>Находит элементы, которые подходят под условие </b></p>

Ответ: 2

Комментарий: SD005096

**<p><b> Что означает параметр fname? <br>**

**int remove( const char \* fname )**

**</b></p>**

1. <p><b>Имя файла для удаления </b></p>
2. <p><b>Имя файла для изменения </b></p>
3. <p><b>Имя файла для добавления </b></p>

Ответ: 1

Комментарий: SD005097

**<p><b> Какое значение функция remove() возвращает в результате ошибки? </b></p>**

1. <p><b>-1 </b></p>
2. <p><b>NULL </b></p>
3. <p><b>ERROR </b></p>

Ответ: 1

Комментарий: SD005098

**<p><b> Какой заголовочный файл нужно подключить для функции remove() в языке C? </b></p>**

1. <p><b>stdio.h </b></p>
2. <p><b>windos.h</b></p>
3. <p><b>iostream </b></p>

Ответ: 1

Комментарий: SD005099

**<p><b> Функция remove() происходит из языка: </b></p>**

1. <p><b>С++ </b></p>
2. <p><b>С </b></p>
3. <p><b>Pascal </b></p>

Ответ: 2

Комментарий: SD005100

**<p><b> Какой заголовочный файл нужно подключить для функции remove() в языке C++? </b></p>**

1. <p><b>stdio.h </b></p>
2. <p><b>cstdio </b></p>
3. <p><b>iomanip </b></p>

Ответ: 2

Комментарий: SD005101

**<p><b> Что означает значение ENOENT ошибки errno? </b></p>**

1. <p><b>Файл не существует </b></p> (верный ответ )
2. <p><b>Доступ запрещен </b></p>
3. <p><b>Ничего из перечисленного </b></p>

Ответ: 1

Комментарий: SD005102

**<p><b> Что означает значение EACESS ошибки errno? </b></p>**

1. <p><b>Доступ запрещен</b></p>
2. <p><b>Файл не существует</b></p>
3. <p><b>Файл пуст</b></p>

Ответ: 1

Комментарий: SD005103

**<p><b> Какие достаточные условия для удаления?</b></p>**

1. <p><b>Файл существовал до запуска программы.</b></p>
2. <p><b>Файл существовал до запуска программы, к нему есть доступ.</b></p>
3. <p><b>В любых случаях удаление возможно.</b></p>

Ответ: 2

Комментарий: SD005104

**<p><b> Что выполняет функция remove\_if()? </b></p>**

1. <p><b>Удаляет все элементы, для которых выполняется указанное условие.</b></p>
2. <p><b>Удаляет все элементы,если хотя бы для одного элемента выполняется указанное условие</b></p>
3. <p><b>Удаляет все элементы, если элементы подходят под достаточные условия удаления.</b></p>

Ответ: 1

Комментарий: SD005105

**<p><b> #include <stdio.h><br>  
int main(void) <br>  
<br>{<br>  
char fname[80]; <br>  
printf ("Name of file to remove: ");<br>  
gets(fname); <br>   
if(remove(fname))**

**<br>{<br>   
printf("Error removing file");<br>  
return 1; <br>  
<br>}<br>  
else return 0; <br>  
}<b></p>**

1. <p><b>программа использует remove() для удаления файла, определяемого пользователем</b></p>
2. <p><b>программа использует remove() для сохранения файла, определяемого пользователем</b></p>

Ответ: 1

Комментарий: SD0051016

**<p><b> Оператор Delete освобождает память из кучи?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 1

Комментарий: SD005107

**<p><b> Команда DELETE удаляет данные из БД ?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 1

Комментарий: SD005108

**<p><b>При удалении записи генерируются события BeforeDelete и AfterDelete типа?</b></p>**

1. <p><b> TDataSetNotifyEvent </b></p>
2. <p><b>AFTER</b></p>

Ответ: 1

Комментарий: SD005109

**<p><b>Delete это?</b></p>**

1. <p><b>команда DELETE,обеспечивающая удаление информации из базы данных </b></p>
2. <p><b> команда DELETE,обеспечивающая обновление информации из базы данных </b></p>

Ответ: 1

Комментарий: SD005110

**<p><b>**

**Что делает программа ? <br>**

**void \_\_fastcall TForm1::Button1Click(TObjееt \*Sender) <br>**

**{<br>**

**int n; <br>**

**Table1->RecNo; <br>**

**while (n >= 1) <br>**

**{<br>**

**Table1->Delete();<br>**

**Table1->RecNo = Table1->RecNo - 1; <br>**

**n--;<br>**

**}<br>**

**}<br>**

**</b></p>**

1. <p><b> удаления записей набора данных с текущей по первую</b></p>
2. <p><b> сохранения записей набора данных с текущей по первую</b></p>

Ответ: 1

Комментарий: SD005111

**<p><b> верно ли написан код if(remove\_\_remo(file) == 0) <br>**

**printf("Файл %s удален.\n");?”</b></p>**

1. <p><b>нет </b></p>
2. <p><b>да </b></p>

Ответ: 1

Комментарий: SD005112

**<p><b> #include <stdio.h><br>**

**String FileName = "c:\\readme.txt";<br>**

**remove(FileName.c\_str());<br>**

**Файлы удаляются безвозвратно ? <br>**

**</b></p>**

1. <p><b>да </b></p>
2. <p><b>нет </b></p>

Ответ: 1

Комментарий: SD005113

**<p><b> "DELETE" переводится с английского как ? </b></p>**

1. <p><b>Удалить </b></p>
2. <p><b>Сохранить</b></p>
3. <p><b>Добавить</b></p>

Ответ: 1

Комментарий: SD005114

**<p><b> что значит выражение “DELETE FROM developers”?</b></p>**

1. <p><b>хотим удалить все данные из таблицы.</b></p>
2. <p><b> хотим добавить все данные из таблицы.</b></p>

Ответ: 1

Комментарий: SD005115

**<p><b> Можно ли удалять информацию в файле без дополнительных библиотек? </b></p**>

1. <p><b>да </b></p>
2. <p><b>нет </b></p>

Ответ: 1

Комментарий: SD005116

**<p><b> Через этот цикл будет удаляться информация ? <br>**

**for(i=0;i<=10;i++)<br>**

**{<br>**

**k=strcmp(buff1,p[i].FIO); <br>**

**if(k==0) <br>**

**{<br>**

**memset(&p[i], 0, sizeof(p[i])); <br>**

**}<br>**

**}<br>**

**for(int j=0;j<=10;j++)<br>**

**{if(p[j].FIO!=NULL&&p[j].number!=NULL&&p[j].pasport!=NULL&&p[j].punkt!=NULL&&p[j].ticket!=NULL) <br>**

**{ <br>**

**fprintf(t,"%s:%s\n","ФИО",p[j].FIO); <br>**

**fprintf(t,"%s:%s\n","Паспорт",p[j].pasport); <br>**

**fprintf(t,"%s:%s\n","Пункт",p[j].punkt); <br>**

**fprintf(t,"%s:%d\n","Номер",p[j].number); <br>**

**fprintf(t,"%s:%d\n","Билеты :",p[j].ticket); <br>**

**}<br>**

**else continue; <br>**

**} <br></b></p>**

1. <p><b>да</b></p>
2. <p><b>нет </b></p>

Ответ: 1

Комментарий: SD005117

**<p><b> Как удалить записи из файла в C++?</b></p>**

1. <p><b> Функция remove() удаляет **файлы**. Она имеет следующий прототип: int remove(const char \*имя\_файла); В случае удачного выполнения она возвращает ноль, а в случае неудачного - не ноль.</b></p>
2. <p><b>Функция used </b></p>

Ответ: 1

Комментарий: SD005118

**<p><b> Что делает программа remove(file) == 0 в коде? <br>**

**#include <stdio.h><br>**

**int main(void) <br>**

**{<br>**

**char file[80]; <br>**

**/\* запросить имя удаляемого файла \*/<br>**

**printf("Введите имя удаляемого файла");<br>**

**gets(file); <br>**

**if(remove(file) == 0) <br>**

**printf("Файл %s удален.\n");<br>**

**else<br>**

**perror("remove");<br>**

**return 0; <br>**

**}<br></b></p>**

1. <p><b> удалить файл.</b></p>
2. <p><b> сохранить файл </b></p>

Ответ: 1

Комментарий: SD005119

**<p><b> Что делает программа remove(file) == 0 в коде? <br>**

**int \_\_fastcall TForm1::DelListDirItems(char \*dir, char \*mask) <br>**

**{<br>**

**WIN32\_FIND\_DATA DIR={0};<br>**

**HANDLE hFile=0; <br>**

**BOOL done=TRUE; <br>**

**char dirname[128]={0};<br>**

**TCHAR LevelUp='.'; <br>**

**int ItemCount=0; <br>**

**char fullpath[256]={0};<br>**

**AnsiString TempString; <br>**

**char DelStr[256]={0};");<br>**

**}<br></b></p>**

1. <p><b> функция удаления файлов по маске mask из директории dir.</b></p>
2. <p><b> сохранить файл </b></p>

Ответ: 1

Комментарий: SD005120

**<p><b> Небольшая часть элемента, которая используется для управления сортировкой.</b></p**

1. <p> <b>Якорь </b> </p>
2. <p> <b> Ключь </b> </p>
3. <p> <b> id </b> </p>

Ответ: 2

Комментарий: SD005121

**<p><b> Конкретные характеристики ключей и элементов в разных приложениях могут существенно отличаться друг от друга, однако абстрактное понятие размещения ключей и связанной с ними информации в определенном порядке и представляет собой суть задачи …. </b></p>**

1. <p><b> Взаимодействия </b></p>
2. <p><b>Компоновки </b></p>
3. <p><b> Компенсации </b></p>
4. <p><b> Сортировки </b></p>

Ответ: 4

Комментарий: SD005122

**<p><b>Метод сортировки, при котором сортируемые файлы полностью помещаются в оперативной памяти, называется </b></p>**

1. <p><b> Прямым </b></p>
2. <p><b> Внутренним </b></p>
3. <p><b> Полноценным</b></p>
4. <p><b> Конкретным </b></p>

Ответ: 2

Комментарий: SD005123

**<p><b> Сортировка файлов, хранящихся на магнитной ленте или диске, называется </b></p>**

1. <p><b> Обратным </b></p>
2. <p><b> Ограниченным </b></p>
3. <p><b> Вложенным </b></p>
4. <p><b>Внешней</b></p>

Ответ: 4

Комментарий: SD005124

**<p><b> Функция (вставить слово) представляет собой шаблонную реализацию, которая обращается к сортируемым элементам только через первый аргумент и нескольких простых операций с данными. </b></p>**

1. <p><b> stl </b></p>
2. <p><b> sort </b></p>
3. <p><b> qsort </b></p>
4. <p><b> caller </b></p>

Ответ: 2

Комментарий: SD005125

**<p> <b> Сортировка, в которой последовательность выполняемых операций не зависит от упорядоченности данных, называется</b> </p>**

1. <p> <b> Линейный </b> </p>
2. <p> <b> Неадаптивной </b> </p>
3. <p> <b> Последовательной</b> </p>
4. <p> <b> Нет правильного ответа </b> </p>

Ответ: 2

Комментарий: SD005126

**<p> <b> Сортировка, в которой последовательность выполняемых операций зависит от упорядоченности данных, называется </b> </p>**

1. <p><b> Адаптивной</b></p>
2. <p><b>Обратной</b></p>
3. <p><b> Непоследовательной </b></p>
4. <p><b> Абстрактной </b></p>

Ответ: 1

Комментарий: SD005127

**<p> <b>** **Если метод сортировки сохраняет относительный порядок размещения в файле элементов с одинаковыми ключами, значит этот метод … </b> </p>**

1. <p><b> Устойчивый </b></p>
2. <p><b> Сбалансированный </b></p>
3. <p><b > Упорядоченный </b></p>
4. <p><b> Одинарный </b></p>

Ответ: 1

Комментарий: SD0005128

**<p> <b> Если переупорядочиваются не сами элементы, а массив указателей так, что первый указатель указывает на наименьший элемент, следующий — на наименьший из оставшихся и т.д. эта сортировка является </b> </p>**

1. <p><b> Прямой </b></p>
2. <p><b> Обрытный </b></p>
3. <p><b> Косвенной </b></p>
4. <p><b> Нет такой сортировки </b></p>

Ответ: 3

Комментарий: SD005129

**<p> <b> Метод, работующий по принципу приведенным далее называется? Сначала находится наименьший элемент массива и меняется местами с элементом, стоящим первым в сортируемом массиве. Потом находится второй наименьший элемент и меняется местами с элементом, стоящим вторым в исходном массиве. Этот процесс продолжается до тех пор, пока весь массив не будет отсортирован. </b> </p>**

1. <p><b> Сортировкой выбором </b></p>
2. <p><b> Пузырьковая сортировка</b></p>
3. <p><b> Сортировка расчёской</b></p>
4. <p><b> Сортировка перемешиванием</b></p>

Ответ: 1

Комментарий: SD005130

**<p> <b> При каждом проходе алгоритма по внутреннему циклу, очередной наибольший элемент массива ставится на своё место в конце массива рядом с предыдущим «наибольшим элементом», а наименьший элемент перемещается на одну позицию к началу массива. Такой метод сортировки называется </b> </p>**

1. <p><b> Чётно-нечётная сортировка </b></p>
2. <p><b> Шейкерная сортировка </b></p>
3. <p><b> Глупая сортировка </b></p>
4. <p><b> Сортировка пузырьком </b></p>

Ответ: 4

Комментарий: SD005131

**<p> <b> Является ли сортировка пузырьком устойчивой </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Она может быть как устойчивой, так и неустойчивой </b></p>

Ответ: 1

Комментарий: SD005132

**<p> <b> Сортировка выбором выполняет порядка … сравнений и … обменов элементов. (Вставить правильный ответ)</b> </p>**

1. <p><b> N2/ 4 сравнений, N / 2 обменов </b></p>
2. <p><b> N2/ 2 сравнений, N обменов </b></p>
3. <p><b> N2/ 4 сравнений, N обменов </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD0050133

**<p> <b> Сортировка вставками выполняет в среднем порядка …  сравнений и … полуобменов (перемещений). (Вставить правильный ответ)</b> </p>**

1. <p><b> N2/ 4 сравнений, N2/ 4 полуобменов </b></p>
2. <p><b> N2/ 4 сравнений, N2/ 2 полуобменов </b></p>
3. <p><b> N2/ 2 сравнений, N2/ 4 полуобменов </b></p>
4. <p><b> Нет правильного ответа</b></p>

Ответ: 1

Комментарий: SD005134

**<p> <b> Пузырьковая сортировка выполняет порядка …  сравнений и … обменов. (Вставить правильный ответ) </b> </p>**

1. <p><b> N2/ 4 сравнений, N2/ 2 полуобменов </b></p>
2. <p><b> N2/ 4 сравнений, N2/ 4 полуобменов </b></p>
3. <p><b> N2/ 2 сравнений, N2/ 2 обменов</b></p>
4. <p><b> Нет правильного ответа</b></p>

Ответ: 3

Комментарий: SD005135

**<p> <b>  Пара ключей, которые нарушают порядок в файле, называются </b> </p>**

1. <p><b> Тождественное преобразование</b></p>
2. <p><b> Инверсией </b></p>
3. <p><b> Нет правильных ответов </b></p>

Ответ: 2

Комментарий: SD005136

**<p> <b> Если отсортированному файлу добавляются нескольких элементов, либо в сортированном файле изменены ключи нескольких элементов, то для таких файлов наиболее эффективна сортировка является </b> </p>**

1. <p><b> Сортировка выбором </b></p>
2. <p><b> Сортировка вставками </b></p>
3. <p><b> Пузырьковая сортировка</b></p>
4. <p><b>Все эти сортировки одинаково хороши </b></p>

Ответ: 2

Комментарий: SD005137

**<p> <b> Время выполнения сортировки вставками зависит от </b> </p>**

1. <p><b> Общего числа инверсий </b></p>
2. <p><b>Характера распределения инверсий </b></p>
3. <p><b> Как от числа, так и от распределения инверсий </b></p>
4. <p><b> Не зависит от инверсий </b></p>

Ответ: 1

Комментарий: SD005138

**<p> <b> Время выполнения сортировки выбором линейно для файлов с  </b> </p>**

1. <p><b> Большими элементами и малыми ключами </b></p>
2. <p><b> Малыми элементами и большими ключами </b></p>
3. <p><b> Большими элементами и большими ключами </b></p>
4. <p><b> Малыми элементами и малыми ключами </b></p>

Ответ: 1

Комментарий: SD005139

**<p> <b> Когда к уже отсортированному файлу добавляются нескольких элементов, либо в сортированном файле изменены ключи нескольких элементов, то такие файлы называются </b> </p>**

1. <p><b> Нарушенными </b></p>
2. <p><b> Изменчивые </b></p>
3. <p><b> Частично упорядоченные </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005140

**<p> <b> Для файлов небольших размеров сортировка вставками и сортировка выбором работают примерно в два раза быстрее пузырьковой сортировки. Утверждение является правдой? </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Не хватает данных для ответа </b></p>

Ответ: 1

Комментарий: SD005141

**<p> <b> Для подсчета количества инверсий в файле необходимо для каждого элемента просуммировать число элементов слева, которые больше его. Так ли это? </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Нет ответа </b></p>

Ответ: 1

Комментарий: SD005142

**<p> <b>  Сортировка Шелла представляет собой простое расширение метода вставок, быстродействие которого достигается за счет возможности … </b> </p>**

1. <p><b> Сужать диапазон поиска элементов </b></p>
2. <p><b> Обмена далеко отстоящих друг от друга элементов </b></p>
3. <p><b> Обмена соседних элементов </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD0005143

**<p> <b> Устойчива ли сортировка Шелла? </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Может быть как устойчивой, так и неустойчивой </b></p>

Ответ: 2

Комментарий: SD005144

**<p> <b> Лучшее время работы сортировки Шелла </b> </p>**

1. <p><b> O(nlogn) </b></p>
2. <p><b> O(n log2 n) </b></p>
3. <p><b> O(n) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005145

**<p> <b> Худшее время работы сортировки Шелла </b> </p>**

1. <p><b> O(n^2) </b></p>
2. <p><b> O(nlogn) </b></p>
3. <p><b> O(nlgn) </b></p>
4. <p><b> O(nloglogn) </b></p>

Ответ: 1

Комментарий: SD005146

**<p> <b> Худшее время работы сортировки выбором </b> </p>**

1. <p><b> О(n^2) </b></p>
2. <p><b> O(nlgn) </b></p>
3. <p><b> O(n log2 n) </b></p>
4. <p><b> O(nloglogn) </b></p>

Ответ: 1

Комментарий: SD005147

**<p> <b> Лучшее время работы сортировки выбором </b> </p>**

1. <p><b> O(nlogn) </b></p>
2. <p><b> O(n) </b></p>
3. <p><b> О(n^2) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005148

**<p> <b> Лучшее время работы сортировки вставками </b> </p>**

1. <p><b> O(nlogn) </b></p>
2. <p><b> O(n) </b></p>
3. <p><b> О(n^2) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005149

**<p> <b> Худшее время работы сортировки вставками </b> </p>**

1. <p><b> O(nlgn) </b></p>
2. <p><b> O(nloglogn) </b></p>
3. <p><b> O(n^2) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005150

**<p><b> Получение конкретного фрагмента или фрагментов информации из больших объемов ранее сохраненных данных это? </b></p>**

1. <p><b> Анализ </b></p>
2. <p><b> Поиск </b></p>
3. <p><b> Выборка </b></p>
4. <p><b> Извлечение </b></p>

Ответ: 2

Комментарий: SD005151

**<p><b>Структура данных элементов с ключами, которая поддерживает две базовые операции: вставку нового элемента и возврат элемента с заданным ключом, называется?</b></p>**

1. <p><b> Таблица символов</b></p>
2. <p><b> Бинарный код </b></p>
3. <p><b> Массив </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: SD005152

**<p><b> Язык С++ позволяет создавать типы данных, которые ведут себя аналогично базовым типам языка Си. Такие типы обычно называют </b></p>**

1. <p><b>Абстрактный тип данных</b></p>
2. <p><b> Примитивный тип данных </b></p>
3. <p><b> Взаимный тип данных </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: SD005153

**<p><b>  Если значения ключей - положительные целые числа, меньшие M, и элементы имеют различные ключи, то тип данных таблицы символов может быть реализован с помощью индексированных значениями ключей массивов так, что для выполнения операций вставить, найти и удалить потребуется </b></p>**

1. <p><b> Постоянное время </b></p>
2. <p><b> Всремя не зависит от значения ключа </b></p>
3. <p><b> N/2 </b></p>
4. <p><b> Мало данных для ответа </b></p>

Ответ: 1

Комментарий: SD005154

**<p><b>  Последовательный поиск в таблице символов с N элементами требует выполнения порядка … сравнений при успешном поиске (в среднем). </b></p>**

1. <p><b> N </b></p>
2. <p><b>N/4 </b></p>
3. <p><b>2N/3 </b></p>
4. <p><b> N/2 </b></p>

Ответ: 4

Комментарий: SD005155

**<p><b> Последовательный поиск в таблице символов с N неупорядоченными элементами требует … количества шагов для выполнения вставок и … сравнений при неудачном поиске (всегда).</b></p>**

1. <p><b> постоянного кол-во шагов, N/2 сравнений </b></p>
2. <p><b> постоянного кол-во шагов, N сравнений </b></p>
3. <p><b> N кол-во шагов, N/2 сравнений </b></p>
4. <p><b> N/2 кол-во шагов, N сравнений </b></p>

Ответ: 2

Комментарий: SD005156

**<p><b> Последовательный поиск в таблице символов из N упорядоченных элементов требует порядка … операций для вставки, успешного поиска и неудачного поиска (в среднем). </b></p>**

1. <p><b> N </b></p>
2. <p><b> N/2 </b></p>
3. <p><b> N^2/2 </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005157

**<p><b> Алгоритм, который изменяет порядок элементов так, чтобы часто запрашиваемые элементы встречались в начале поиска, называется? </b></p>**

1. <p><b> Аглоритм интерполирующего поиск </b></p>
2. <p><b> Алгоритм самоорганизующегося поиска </b></p>
3. <p><b> Алгоритм значемого поиска </b></p>

Ответ: 2

Комментарий: SD005158

**<p><b> Классический алгоритм поиска элемента в отсортированном массиве, использующий дробление массива на половины называется </b></p>**

1. <p><b> Линейный поиск  </b></p>
2. <p><b> Бинарный поиск </b></p>
3. <p><b> Интерполирующий поиск </b></p>
4. <p><b> Поиск подстроки в строке  </b></p>

Ответ: 2

Комментарий: SD005159

**<p><b> При бинарном поиске выполняется не более чем … сравнений (и при успешном, и при неудачном). </b></p>**

1. <p><b> N </b></p>
2. <p><b> N/2 </b></p>
3. <p><b> N - 1 </b></p>
4. <p><b> [lgN] + 1  </b></p>

Ответ: 4

Комментарий: SD005160

**<p><b> Наибольший недостаток использования бинарного поиска </b></p>**

1. <p><b> Потенциально высокие затраты памяти </b></p>
2. <p><b> Для поиска массив должен быть упорядочен </b></p>
3. <p><b> Длительное время работы </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005161

**<p><b> Достоинства бинарного поиска </b></p>**

1. <p><b>  Быстрая работа алгоритма </b></p>
2. <p><b> Прост в реализации </b></p>
3. <p><b> Практичность </b></p>
4. <p><b> Все ответы верны </b></p>

Ответ: 4

Комментарий: SD005162

**<p><b>  Алгоритм который перебирает все элементы в массиве, сравнивая их с заданным ключом называется </b></p>**

1. <p><b> Линейный алгоритм </b></p>
2. <p><b> Бинарный алгоритм </b></p>
3. <p><b> [Интерполирующий поиск С++](https://purecodecpp.com/archives/2503)</b></p>
4. <p><b> Такого алгоритма нету </b></p>

Ответ: 1

Комментарий: SD005163

**<p><b> Иерархическая структура данных, в которой каждый узел имеет значение (оно же является в данном случае и ключом) и ссылки на левого и правого потомка, называется </b></p>**

1. <p><b> Родовая структура </b></p>
2. <p><b> Иерархическое дерево </b></p>
3. <p><b> Бинарное дерево </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005164

**<p><b> Бинарное дерево поиска (binary search tree - BST) — это бинарное дерево, обладающее дополнительными свойствами, какими? </b></p>**

1. <p><b> значение левого потомка меньше значения родителя, а значение правого потомка больше значения родителя для каждого узла дерева. </b></p>
2. <p><b> значение левого потомка больше значения родителя, а значение правого потомка меньше значения родителя для каждого узла дерева. </b></p>
3. <p><b> значение левого потомка меньше значения родителя и значение правого потомка меньше значения родителя для каждого узла дерева. </b></p>
4. <p><b> значение левого потомка меньше значения родителя, а значение правого потомка равно значению родителя для каждого узла дерева. </b></p>

Ответ: 1

Комментарий: SD005165

**<p><b> BST-деревья используются для реализации операций </b></p>**

1. <p><b> Найти </b></p>
2. <p><b> Вставить </b></p>
3. <p><b> Создать </b></p>
4. <p><b> Найти, вставить, создать и подсчитать </b></p>

Ответ: 4

Комментарий: SD005166

**<p><b> Как ещё называют таблицу символов </b></p>**

1. <p><b> Карта </b></p>
2. <p><b> Таблица неясности </b></p>
3. <p><b> Словарь </b></p>

Ответ: 3

Комментарий: SD005167

**<p><b> В дереве бинарного поиска, образованном N случайными ключами, для успешного поиска в среднем требуется около … сравнений </b></p>**

1. <p><b> 2lgN ≈ 1.39lgN </b></p>
2. <p><b> lgN </b></p>
3. <p><b> 2lgN ≈ 1.50lgN </b></p>
4. <p><b> N </b></p>

Ответ: 1

Комментарий: SD005168

**<p><b> В дереве бинарного поиска, образованном N случайными ключами, для вставок и неудачного поиска в среднем требуется около … сравнений. </b></p>**

1. <p><b> 2lgN ≈ 1.50lgN </b></p>
2. <p><b> lgN </b></p>
3. <p><b> 2lgN ≈ 1.39lgN </b></p>
4. <p><b> N </b></p>

Ответ: 3

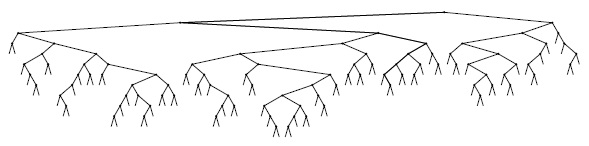
Комментарий: SD005169

**<p><b>  Для поиска в дереве бинарного поиска с N ключами в худшем случае может потребоваться … сравнений. </b></p>**

1. <p><b> lgN </b></p>
2. <p><b> N </b></p>
3. <p><b> 2N </b></p>
4. <p><b> 1/2\*N </b></p>

Ответ: 2

Комментарий: SD005170

**<p><b> Что изображено на рисунке? </b></p>**

1. <p><b> Интерполяционный поиск </b></p>
2. <p><b> Фактографический поиск </b></p>
3. <p><b> Дерево бинарного поиска </b></p>

Ответ: 3

Комментарий: SD005171

**<p><b> Функция, которая вызывает сама себя, называется </b></p>**

1. <p><b> Саморегулируемая функция </b></p>
2. <p><b> Обратная функция </b></p>
3. <p><b> Рекурсивная функция </b></p>
4. <p><b> Функция ротации </b></p>

Ответ: 3

Комментарий: SD005172

**<p><b> Динамическая структура, в которой хранится информация о функциях, выполняющихся в процессоре, назывется </b></p>**

1. <p><b> Стек вызовов </b></p>
2. <p><b> Стек памяти </b></p>
3. <p><b> Стек динамической функции </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: SD005173

**<p><b>  Узел, содержащий ключ S, перемещается в дереве вниз и становится правым дочерним узлом своего прежнего левого дочернего узла. Это?</b></p>**

1. <p><b> Ротация влево в BST-дереве </b></p>
2. <p><b> Ротация вправо в BST-дереве </b></p>
3. <p><b> Без направленная ротация в BST-дереве </b></p>
4. <p><b> Правый дочерний узел </b></p>

Ответ: 2

Комментарий: SD005174

**<p><b> Узел, содержащий ключ A, перемещается в дереве вниз и становится левым дочерним узлом своего прежнего правого дочернего узла. Это? </b></p>**

1. <p><b> Ротация влево в BST- </b></p>
2. <p><b> Ротация вправо в BST-дереве </b></p>
3. 3. <p><b> Левый дочерний узел </b></p>

Ответ: 1

Комментарий: SD005175

**<p><b> Локальное изменение позволяющие перемещать узлы по деревьям без изменения глобальных свойств упорядоченности, которые и делают BST-дерево полезной для поиска структурой, это? </b></p>**

1. <p><b> Аменция </b></p>
2. <p><b> Проекция </b></p>
3. <p><b> Ротация </b></p>
4. <p><b> Абсолюция </b></p>

Ответ: 3

Комментарий: SD005176

**<p><b> Ротации применяются для </b></p>**

1. <p><b> Удаления определенных узлов дерева, для последующего использования. </b></p>
2. <p><b> Перемещения конкретных узлов по дереву и предотвращения разбалансировки деревьев. </b></p>
3. <p><b> Выбора и последующего перемещения для разбалансировки дерева. </b></p>

Ответ: 2

Комментарий: SD005177

**<p><b> Что такое вес бинарного дерева? </b></p>**

1. <p><b> Число листьев </b></p>
2. <p><b> Число узлов </b></p>
3. <p><b> Число узлов, ссылающихся на нулевой элемент </b></p>
4. <p><b> Число дуг дерева </b></p>

Ответ: 1

Комментарий: SD005178

**<p><b> Выберите утверждения, верные для идеально сбалансированного дерева.**

**</b></p>**

1. <p><b> Выберите утверждения, верные для идеально сбалансированного дерева. </b></p>
2. <p><b> Каждый уровень содержит не более узлов, где корень - нулевой узел. </b></p>
3. <p><b> Для каждой его вершины высота ее двух поддеревьев различается не более, чем на 1 </b></p>
4. <p><b> Для поиска любого элемента перебирается не более log2n вершин, где n - число вершин в дереве </b></p>

Ответ: 1, 3, 4

Комментарий: SD005179

**<p><b> Сбалансированным бинарным деревом называется... </b></p>**

1. <p><b> Дерево, для каждой вершины которого высота двух его поддеревьев различается не более чем на единицу</b></p>
2. <p><b> Дерево, элементы которого распределены таким образом, что слева от корня находятся элементы с меньшим индексом, а справа - с большим</b></p>
3. <p><b> Дерево, в котором у каждого поддерева имеется пара потомков </b></p>
4. <p><b> Дерево, индекс корня которого равен среднему арифметическому всех остальных индексов </b></p>

Ответ: 1

Комментарий: SD005180

**<p><b>Алгоритмы, в соответствии с которыми решение поставленных задач сводится к арифметическим действиям, называются</b></p>**

1. <p><b>разветвляющимися</b></p>

2. <p><b>линейными</b></p>

3. <p><b>численными</b></p>

Ответ: 3

Комментарий: AP00101

**<p><b>Система четких однозначных указаний, которая определяет последовательность действий над некоторыми объектами и после конечного числа шагов приводит к получению требуемого результата – это</b></p>**

1. <p><b>алгоритм</b></p>

2. <p><b>схема алгоритма</b></p>

3. <p><b>тело функции</b></p>

4. <p><b>программа</b></p>

Ответ: 1

Комментарий: AP00102

**<p><b>Какое свойство алгоритма означает, что исполнение алгоритма должно закончиться за конечное число шагов?</b></p>**

1. <p><b>точность</b></p>

2. <p><b>результативность</b></p>

3. <p><b>дискретность</b></p>

4. <p><b>массовость</b></p>

Ответ: 2

Комментарий: AP00103

**<p><b>Всякий алгоритм может быть</b></p>**

1. <p><b>записан на естественном языке </b></p>

2. <p><b>изображен в виде рисунка</b></p>

3. <p><b>записан на русском языке</b></p>

Ответ: 1

Комментарий: AP00104

<p><b>Очередь с приоритетом - …</b></p>

1.<p><b>абстрактный контейнер</b></p>

2.<p><b>многомерный массив</b></p>

3.<p><b>двусвязный список</b></p>

4.<p><b>абстрактный массив</b></p>

Ответ:1

Комментарий:AP00601

<p><b> Каждому элементу очереди с приоритетами сопоставлено … </b></p>

1.<p><b> некоторое значение, именуемое идентификатором этого элемента </b></p>

2.<p><b> некоторое значение, именуемое номером этого элемента </b></p>

3.<p><b> некоторое значение, именуемое приоритетом этого элемента </b></p>

4.<p><b> некоторое значение, именуемое адресом этого элемента</b></p>

Ответ:3

Комментарий:AP00602

<p><b> Приоритеты допускают …</b></p>

1.<p><b> сравнение друг с другом </b></p>

2.<p><b> ошибку</b></p>

3.<p><b> изменения в коде</b></p>

4.<p><b> различные комбинации</b></p>

Ответ:1

Комментарий:AP00603

<p><b>Класс std::priority\_queve – это …</b></p>

1.<p><b>контейнерный адаптер, который реализует очередь, из которой элементы читаются в соответствии с их приоритетом</b></p>

2.<p><b> контейнерный адаптер, который реализует связной список, из которой элементы читаются в соответствии с их приоритетом </b></p>

3.<p><b> контейнерный адаптер, который реализует очередь, из которой элементы читаются в соответствии с их номером в очереди</b></p>

4.<p><b> контейнерный адаптер, который реализует массив, из которого элементы читаются в соответствии с нужным нам элементом </b></p>

Ответ:1

Комментарий:AP00604

<p><b>Для чего используется функция push?</b></p>

1.<p><b>для очистки очереди</b></p>

2.<p><b>для доступа к верхнему элементу</b></p>

3.<p><b>Вставляет новый элемент</b></p>

4.<p><b>для вывода очереди</b></p>

Ответ:3

Комментарий:AP00605

<p><b>Для чего используется функция top?</b></p>

1.<p><b>для очистки очереди</b></p>

2.<p><b>для доступа к верхнему элементу</b></p>

3.<p><b>Вставляет новый элемент</b></p>

4.<p><b>для вывода очереди</b></p>

Ответ:4

Комментарий:AP00606

<p><b>Очереди с приоритетом можно реализовать с помощью следующих структур данных: </b></p>

1.<p><b> массив, связный список, куча </b></p>

2.<p><b> массив, куча и двоичное дерево поиска </b></p>

3.<p><b> только через массив</b></p>

4.<p><b> массив, связный список, куча и двоичное дерево поиска </b></p>

Ответ:4

Комментарий:AP00607

<p><b> Функция извлечения из очереди с приоритетами возвращает тот элемент, …</b></p>

1.<p><b> приоритет которого является минимальным </b></p>

2.<p><b> приоритет которого является максимальным </b></p>

3.<p><b> который мы выберем </b></p>

4.<p><b> приоритет которого является средним </b></p>

Ответ:2

Комментарий:AP00608

<p><b> Интерфейс очереди с приоритетами в целом аналогичен …</b></p>

1.<p><b> интерфейсу обычной очереди </b></p>

2.<p><b> интерфейсу одномерного списка </b></p>

3.<p><b> интерфейсу обычного массива</b></p>

4.<p><b> интерфейсу многомерного массива</b></p>

Ответ:1

Комментарий:AP00609

<p><b>Чем отличается очередь с приоритетом от обычной очереди?</b></p>

1.<p><b> в операции очищения появляется второй аргумент, а операция удаления возвращает элемент с наибольшим приоритетом </b></p>

2.<p><b> в операции добавления изменяется первый аргумент, а операция удаления возвращает элемент с наибольшим приоритетом </b></p>

3.<p><b> в операции добавления появляется второй аргумент, а операция удаления возвращает элемент с наибольшим приоритетом </b></p>

4.<p><b> в операции добавления появляется второй аргумент, а операция удаления не возвращает элемент с наибольшим приоритетом </b></p>

Ответ:3

Комментарий:AP00610

<p><b>Что осуществляет данный код?</b></p>

<p><b>

**bool isNull(void)** <br>

**{ return (head == NULL);** <br>

**}**<br>

1.<p><b>Проверку на пустоту</b></p>

2.<p><b> Извлечение элемента из начала </b></p>

3.<p><b> Получение элемента из начала </b></p>

4.<p><b> Добавление элемента в очередь </b></p>

Ответ:1

Комментарий:AP00611

<p><b> Что осуществляет данный код?</b></p>

<p><b>

**void getFromHead()** <br>

**{ if (isNull())** <br>

**cout << "Очередь пуста" << endl;** <br>

**else** <br>

**cout << "Начало = " << head->data << endl;** <br>

**}**<br>

</b></p>

1.<p><b>Проверку на пустоту</b></p>

2.<p><b> Извлечение элемента из начала </b></p>

3.<p><b> Получение элемента из начала </b></p>

4.<p><b> Добавление элемента в очередь </b></p>

Ответ:2

Комментарий:AP00612

<p><b> Что осуществляет данный код?</b></p>

<p><b>

**void insertToQueue(int x)** <br>

**{ Item \*p = new Item;** <br>

**p->data = x;** <br>

**p->next = NULL;** <br>

**Item \*v = new Item;**<br>

**Item \*p1 = new Item;** <br>

**Item \*p2 = new Item;** <br>

**int i = 0;** <br>

**if (isNull())** <br>

**head = tail = p;** <br>

**else**<br>

**{ p2 = head; p1 = head;** <br>

**while (p1 != NULL)** <br>

**{ if (i == 1)** <br>

**{ if (x <= p1->data)** <br>

**{ v->data = x;** <br>

**v->next = p1;** <br>

**p2->next = v;** <br>

**return;** <br>

**}**<br>

**p2 = p2->next;** <br>

**}**<br>

**else**<br>

**{ if (x <= p1->data)** <br>

**{ v->data = x;** <br>

**v->next = p1;** <br>

**head = v;** <br>

**return;** <br>

**}**<br>

**}**<br>

**p1 = p1->next;** <br>

**i = 1;** <br>

**}**<br>

**if (p1 == NULL)** <br>

**{ tail->next = p;** <br>

**tail = p;** <br>

**}**<br>

**}**<br>

**}**<br>

</b></p>

1.<p><b>Проверку на пустоту</b></p>

2.<p><b> Извлечение элемента из начала </b></p>

3.<p><b> Получение элемента из начала </b></p>

4.<p><b> Добавление элемента в очередь </b></p>

Ответ:4

Комментарий:AP00613

<p><b> Что осуществляет данный код?</b></p>

<p><b>

**void deletFirst()**<br>

**{ if (isNull())**<br>

**cout << "Очередь пуста" << endl;** <br>

**else**<br>

**{ Item \*p = head;** <br>

**head = head->next;** <br>

**delete p;** <br>

**}**<br>

**}**<br>

</b></p>

1.<p><b>Проверку на пустоту</b></p>

2.<p><b> Извлечение элемента из начала </b></p>

3.<p><b> Получение элемента из начала </b></p>

4.<p><b> Добавление элемента в очередь </b></p>

Ответ:2

Комментарий:AP00614

<p><b> Что осуществляет данный код?</b></p>

<p><b>

**void printQueue()** <br>

**{ Item \*p = new Item;** <br>

**if (isNull())**<br>

**cout << "Очередь пуста" << endl;** <br>

**else**<br>

**{ cout << "Очередь = ";**<br>

**p = head;** <br>

**while (!isNull())**<br>

**{ if (p != NULL)** <br>

**{ cout << p->data << " "; cout << "->";** <br>

**p = p->next;** <br>

**}**<br>

**else** <br>

**{ cout << "NULL" << endl;** <br>

**return;** <br>

**}**<br>

**}**<br>

**}**<br>

**}**

</b></p>

1.<p><b>Проверку на пустоту</b></p>

2.<p><b> Извлечение элемента из начала </b></p>

3.<p><b> Очистка очереди </b></p>

4.<p><b> Вывод очереди </b></p>

Ответ:4

Комментарий:AP00615

<p><b> Что осуществляет данный код?</b></p>

<p><b>

**void clrQueue()** <br>

**{ while (!isNull()) deletFirst();**<br>

**}**<br>

</b></p>

1.<p><b>Проверку на пустоту</b></p>

2.<p><b> Извлечение элемента из начала </b></p>

3.<p><b> Очистка очереди </b></p>

4.<p><b> Вывод очереди </b></p>

Ответ:3

Комментарий:AP00616

<p><b> Очередь с приоритетами, построенная на обычном массиве или списке, позволяет …</b></p>

1.<p><b>нет правильного варианта ответа</b></p>

2.<p><b> не выполнять вставку за O(1), но поиск и удаление элемента с максимальным приоритетом будут иметь сложность O(N) </b></p>

3.<p><b> выполнять вставку за O(1), но поиск и удаление элемента с максимальным приоритетом не будут иметь сложность O(N) </b></p>

4.<p><b> выполнять вставку за O(1), но поиск и удаление элемента с максимальным приоритетом будут иметь сложность O(N) </b></p>

Ответ:4

Комментарий:AP00617

<p><b> Существует несколько видов приоритетных очередей:</b></p>

1.<p><b> Очередь с приоритетным включением</b></p>

2.<p><b> Очереди с приоритетным исключением</b></p>

3.<p><b> Очереди с одномерным исключением</b></p>

4.<p><b> Очереди с многомерным исключением</b></p>

Ответ:1,2

Комментарий:AP00618

<p><b> Очередь с приоритетным включением - …</b></p>

1.<p><b> последовательность элементов очереди не является строго упорядоченной </b></p>

2.<p><b> последовательность элементов очереди является строго упорядоченной </b></p>

3.<p><b> элемент не добавляется в конец очереди, а при извлечении осуществляется самого приоритетного элемента, который впоследствии удаляется из очереди </b></p>

4.<p><b> элемент добавляется в конец очереди, а при извлечении осуществляется самого приоритетного элемента, который впоследствии удаляется из очереди </b></p>

Ответ:2

Комментарий:AP00619

<p><b> Очередь с приоритетным исключением - …</b></p>

1.<p><b> последовательность элементов очереди не является строго упорядоченной </b></p>

2.<p><b> последовательность элементов очереди является строго упорядоченной </b></p>

3.<p><b> элемент не добавляется в конец очереди, а при извлечении осуществляется самого приоритетного элемента, который впоследствии удаляется из очереди </b></p>

4.<p><b> элемент добавляется в конец очереди, а при извлечении осуществляется самого приоритетного элемента, который впоследствии удаляется из очереди </b></p>

Ответ:4

Комментарий:AP00620

<p><b>Отличие очереди с приоритетом от кучи:</b></p>

1.<p><b>Очередь с приоритетом позволяет поддерживать множество элементов, находить и извлекать минимум из него, тогда как куча - конкретная структура данных, основанная на полном бинарном дереве</b></p>

2.<p><b> Очередь с приоритетом не позволяет поддерживать множество элементов, находить и извлекать минимум из него, тогда как куча - позаоляет </b></p>

3.<p><b> Очередь с приоритетом не позволяет поддерживать множество элементов, находить и извлекать минимум из него, тогда как куча - конкретная структура данных, основанная на полном бинарном дереве </b></p>

4.<p><b> Очередь с приоритетом позволяет поддерживать множество элементов, находить и извлекать минимум из него, тогда как куча - конкретная структура данных, основанная на полном дереве </b></p>

Ответ:1

Комментарий:AP00621

<p><b>Выберете правильное утверждение</b></p>

1.<p><b> Любой односвязный список является очередью с приоритетом, но не любая очередь с приоритетом (в теории) является односвязным списком (на практике почти что любая) </b></p>

2.<p><b> Любая куча не является очередью с приоритетом, зато любая очередь с приоритетом (в теории) является кучей (на практике почти что любая) </b></p>

3.<p><b> Любая куча является очередью с приоритетом, но не любая очередь с приоритетом (в теории) является кучей (на практике почти что любая) </b></p>

4.<p><b>Нет правильного варианта ответа</b></p>

Ответ:3

Комментарий:AP00622

<p><b> swap() — …</b></p>

1.<p><b> меняет элементы двух контейнеров местами </b></p>

2.<p><b> удаляет элемент из очереди</b></p>

3.<p><b> добавляет элемент в очередь</b></p>

4.<p><b> очищает очередь</b></p>

Ответ:1

Комментарий:AP00623

<p><b> emplace() — …</b></p>

1.<p><b>нет правильного ответа</b></p>

2.<p><b> выбирает объект, передавая конструктору указанные через запятую значения, а затем добавляет объект в очередь </b></p>

3.<p><b> удалить объект, передавая конструктору указанные через запятую значения, а затем добавляет объект в очередь </b></p>

4.<p><b> создает объект, передавая конструктору указанные через запятую значения, а затем добавляет объект в очередь </b></p>

Ответ:4

Комментарий:AP00624

<p><b> empty() — …</b></p>

1.<p><b> возвращает значение true, если очередь не содержит элементов, и false — в противном случае </b></p>

2.<p><b> возвращает значение true, если очередь содержит элементов, и false — в противном случае </b></p>

3.<p><b> создаёт очередь</b></p>

4.<p><b> возвращает значение true, если очередь не содержит элементов.</b></p>

Ответ:1

Комментарий:AP00625

<p><b> size() — …</b></p>

1.<p><b> изменяет приоритет</b></p>

2.<p><b> удаляет приоритет</b></p>

3.<p><b> удаляет очередь с приоритетом </b></p>

4.<p><b> возвращает количество элементов в очереди </b></p>

Ответ:4

Комментарий:AP00626

<p><b> Упорядочение элементов в очереди производится по убыванию; в качестве операции сравнения по умолчанию используется оператор … </b></p>

1.<p><b> < </b></p>

2.<p><b> ? </b></p>

3.<p><b> ! </b></p>

4.<p><b> & </b></p>

Ответ:1

Комментарий:AP00627

<p><b>В стандартной библиотеке шаблонов C++ присутствует шаблон priority\_queue<T>. Для возможности его использования требуется подключить заголовочный файл … и пространство имён std</b></p>

1.<p><b> include </b></p>

2.<p><b> queue </b></p>

3.<p><b add.h < b></p>

4.<p><b> stak </b></p>

Ответ:2

Комментарий:AP00628

<p><b>Очередь с приоритетами в STL создаётся на базе шаблона …</b></p>

1.<p><b> хаотичного контейнера с произвольным доступом (vector или deque)</b></p>

2.<p><b> последовательного контейнера с произвольным доступом (vector или deque)</b></p>

3.<p><b> броуновского контейнера с произвольным доступом (vector или deque)</b></p>

4.<p><b> нет правильного варианта ответа</b></p>

Ответ:2

Комментарий:AP00629

<p><b> функциональный объект greater объявлен в заголовочном файле </b></p>

1.<p><b> functional </b></p>

2.<p><b> studio.h </b></p>

3.<p><b> funk.h </b></p>

4.<p><b> windows.h </b></p>

Ответ:1

Комментарий:AP00630

<p><b>Правда ли что добавление нового элемента в массиве может нарушить основное свойство кучи? </b></p>

1. <p><b>Да, всегда</b></p>

2. <p><b>Нет, никогда</b></p>

3. <p><b>Да, но не всегда</b></p>

Ответ: 3

Комментарий: SD001181

<p><b>Дайте определение понятию “Лист, листовой или терминальный узел” </b></p>

1. <p><b>узел, не имеющий дочерних элементов</b></p>

2. <p><b>любой узел дерева, имеющий потомков</b></p>

3. <p><b>одна из вершин, по желанию наблюдателя</b></p>

Ответ: 1

Комментарий: SD001182

<p><b>Дайте определение понятию “Внутренний узел” </b></p>

1. <p><b>самый верхний узел дерева </b></p>

2. <p><b> любой узел дерева, имеющий потомков</b></p>

3. <p><b> узел, не имеющий дочерних элементов</b></p>

Ответ: 2

Комментарий: SD001183

<p><b>Дана двоичная куча в виде одномерного массива, необходимо найти индекс левого дочернего элемента. Какой формулой необходимо воспользоваться для этого зная индекс i родительского элемента?</b></p>

1. <p><b> 2\*i</b></p>

2. <p><b>2\*i+1 </b></p>

3. <p><b>2\*i+2 </b></p>

Ответ: 2

Комментарий: SD001184

<p><b>Даны двоичные max-кучи, определите max-кучу(и) в которой(ых) не нарушено основное свойство кучи<b></p>

1. <p><b>9 8 7 2 5 6 4 3 1</b></p>

2. <p><b>9 7 8 6 4 5 3 2 1 </b></p>

3. <p><b>9 8 7 3 5 6 4 2 1</b></p>

Ответ: 2, 3

Комментарий: SD001185

<p><b>В дереве каждый узел содержит …</b></p>

1. <p><b>указатель на текст слова</b></p>

2. <p><b>счетчик числа встречаемости</b></p>

3. <p><b>указатель на левого и правого потомка</b></p>

Ответ: 1, 2, 3

Комментарий: SD001186

<p><b>Вставьте пропущенное словосочетание “… заключается в посещении текущего узла, обходе левого поддерева, обходе правого поддерева ”</b></p>

1. <p><b>Нисходящий обход</b></p>

2. <p><b>Восходящий обход</b></p>

3. <p><b>Смешанный обход</b></p>

Ответ: 1

Комментарий: SD001187

<p><b> Вставьте пропущенное словосочетание “… заключается в посещении при обходе левого узла поддерева, обходе правого поддерева, посещении текущего узла”</b></p>

1. <p><b>Нисходящий обход</b></p>

2. <p><b>Восходящий обход</b></p>

3. <p><b>Смешанный обход</b></p>

Ответ: 2

Комментарий: SD001188

<p><b>По какой формуле из ниже приведенных можно определить максимальное количество узлов в двоичном дереве с высотой n, где корень — нулевая высота</b></p>

1. <p><b>2n − 1 </b></p>

2. <p><b>2n⁺¹ – 1 </b></p>

3. <p><b>2n⁻¹ + 1 </b></p>

Ответ: 2

Комментарий: SD001189

<p><b> Дайте определение понятию “декартовое дерево”</b></p>

1. <p><b> один из видов самобалансирующихся двоичных деревьев поиска</b></p>

2. <p><b> сбалансированное по высоте двоичное дерево поиска</b></p>

3. <p><b>это структура данных, объединяющая в себе [бинарное дерево поиска](https://neerc.ifmo.ru/wiki/index.php?title=%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_%D0%BF%D0%BE%D0%B8%D1%81%D0%BA%D0%B0,_%D0%BD%D0%B0%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D1%80%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) и [бинарную кучу](https://neerc.ifmo.ru/wiki/index.php?title=%D0%94%D0%B2%D0%BE%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D0%BA%D1%83%D1%87%D0%B0) </b></p>

Ответ: 3

Комментарий: SD001190

<p><b> Может ли в дереве быть цикл?</b></p>

1. <p><b> Да</b></p>

2. <p><b> Нет</b></p>

3. <p><b> Может быть а может и не быть</b></p>

Ответ: 2

Комментарий: SD01191

<p><b> Является ли обязательной ссылка на родительский узел в структуре Node?</b></p>

1. <p><b> Да </b></p>

2. <p><b> Нет </b></p>

3. <p><b> Да, но не всегда</b></p>

Ответ: 2

Комментарий: SD001192

<p><b> что делает данная операция?

if (\*head == NULL) {<br>

    \*head = getFreeNode(value, NULL);<br>

    return;<br>

}

</b></p>

1. <p><b> Проверяет, если дерево пустое, то вставляем корень</b></p>

2. <p><b> Проходит по дереву и ищет место для вставки </b></p>

3. <p><b> Написана ошибочная операция</b></p>

Ответ: 1

Комментарий: SD001193

<p><b> Сколько байт выделится в памяти для данной строки: int \*array = new int[10];</b></p>

1. <p><b>4</b></p>

2. <p><b> 10</b></p>

3. <p><b>40 </b></p>

Ответ: 3

Комментарий: SD001194

<p><b> За что отвечает данная операция inorderTraversal</b></p>

1. <p><b> обход узлов в отсортированном порядке</b></p>

2. <p><b> обход узлов в порядке: вершина, левое поддерево, правое поддерево </b></p>

3. <p><b> обход узлов в порядке: левое поддерево, правое поддерево, вершина</b></p>

Ответ: 1

Комментарий: SD001195

<p><b> За что отвечает данная операция postorderTraversal</b></p>

1. <p><b> обход узлов в отсортированном порядке</b></p>

2. <p><b>обход узлов в порядке: вершина, левое поддерево, правое поддерево,</b></p>

3. <p><b> обход узлов в порядке: левое поддерево, правое поддерево, вершина</b></p>

Ответ: 3

Комментарий: SD001196

<p><b>preorderTraversal </b></p>

1. <p><b> обход узлов в отсортированном порядке</b></p>

2. <p><b>обход узлов в порядке: вершина, левое поддерево, правое поддерево </b></p>

3. <p><b> обход узлов в порядке: левое поддерево, правое поддерево, вершин.</b></p>

Ответ: 2

Комментарий: SD001197

<p><b> Могут ли операции поиска ключа, обхода дерева, поиска минимума/ максимума, поиск следующего/предыдущего изменить структуру рандомизированного бинарного дерева поиска? </b></p>

1. <p><b>Да, всегда </b></p>

2. <p><b>Нет, никогда </b></p>

3. <p><b>Да, но не всегда </b></p>

Ответ: 2

Комментарий: SD001198

<p><b>Можно ли реализовать добавление нового элемента в дерево без рекурсии? </b></p>

1. <p><b> Да</b></p>

2. <p><b> Нет</b></p>

3. <p><b> не знаю ☹</b></p>

Ответ: 1

Комментарий: SD001199

<p><b> Обязателен ли указатель на родительский узел у бинарного дерева?</</b></p>

1. <p><b> Да<b></p>

2. <p><b>Нет </b></p>

3. <p><b>Не знаю ☹ </b></p>

Ответ: 2

Комментарий: SD001200

<p><b> Какой обход совершает данная функция?

void preorder(struct tree \*root)<br>

{<br>

if(!root) return; <br>

if(root->info) printf("%c ", root->info); <br>

preorder(root->left); <br>

preorder(root->right); <br>

}<br></b></p>

1. <p><b> обход дерева в ширину и в глубину</b></p>

2. <p><b> обход дерева в ширину</b></p>

3. <p><b> обход дерева в глубину</b></p>

Ответ: 1

Комментарий: SD001201

<p><b> Как называется поиск в котором мы посещаем каждый узел на уровне прежде чем перейти на следующий уровень </b></p>

1. <p><b> Поиск в глубину и ширину</b></p>

2. <p><b> поиск в глубину</b></p>

3. <p><b>поиск в ширину </b></p>

Ответ: 3

Комментарий: SD001202

<p><b>Может ли быть несколько служебных полей в вершине дерева? </b></p>

1. <p><b>Да </b></p>

2. <p><b>Нет </b></p>

3. <p><b> Не знаю☹</b></p>

Ответ: 1

Комментарий: SD001203

<p><b> разместите в правильном порядке Алгоритм концевого обхода дерева, который заключается в следующем:

1) обойдите правое поддерево;

2) посетите корень дерева.

3) обойдите левое поддерево;

</b></p>

1. <p><b> 132</b></p>

2. <p><b>213 </b></p>

3. <p><b>312 </b></p>

Ответ: 3

Комментарий: SD001204

<p><b>Укажите номера c ошибкой в данных условиях двоичной кучи </b></p>

1. <p><b>Значение в любой вершине не меньше чем значение в потомках </b></p>

2. <p><b> Расстояние до корня отличается более чем на один уровень</b></p>

3. <p><b> Заполняется слева на прав</b></p>

Ответ: 2

Комментарий: SD001205

<p><b>По какой формуле можно определить высоту двоичной кучи? </b></p>

1. <p><b> N\*Log2(N)</b></p>

2. <p><b> Log2(N+1)</b></p>

3. <p><b>Log2(N) </b></p>

Ответ: 1

Комментарий: SD001206

<p><b>Используя формулу из пошлого вопроса определите высоту кучи с 10ю элементами массива </b></p>

1. <p><b> 3</b></p>

2. <p><b> 3,46</b></p>

3. <p><b> 4</b></p>

Ответ: 3

Комментарий: SD001207

<p><b>Дана двоичная куча в виде одномерного массива, необходимо найти индекс правого дочернего элемента. Какой формулой необходимо воспользоваться для этого зная индекс i родительского элемента?</b></p>

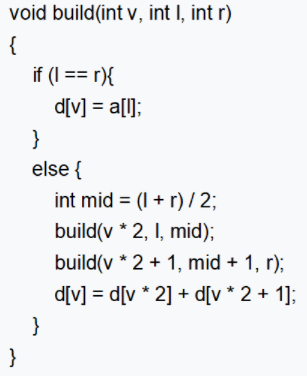
1. <p><b> 2\*i</b></p>

2. <p><b>2\*i+1 </b></p>

3. <p><b>2\*i+2 </b></p>

Ответ: 3

Комментарий: SD001208

<p><b>на рисунке приведено построение … </b></p>

1. <p><b>бинарного дерева</b></p>

2. <p><b>дерева отрезков</b></p>

3. <p><b>двоичного дерева </b></p>

Ответ: 2

Комментарий: SD01209

<p><b> За что отвечает данная строка? Void outHeat()</b></p>

1. <p><b>добавление элемента кучи </b></p>

2. <p><b>вывод элементов кучи в форме кучи </b></p>

3. <p><b> вывод элементов кучи в форме массива </b></p>

Ответ: 2

Комментарий: SD001210

<p><b>Что такое нелинейные структуры данных? </b></p>

1<p><b>Структуры, у которых связи между элементами зависят от выполнения определенных условий. </b></p>

2.<p><b>Структуры, в которых связи между элементами не зависят от выполнения какого-либо условия. </b></p>

3.<p><b>Структуры, состоящие из наборов вершин и ребер. </b></p>

4.<p><b>Линейные структуры, в которых операции включения/исключения элемента могут выполняться как с одного, так и с другого конца. </b></p>

Ответ:1

Комментарий:SD001211

<p><b>Что относиться к нелинейным структурам данных? </b></p>

1.<p><b>Очередь </b></p>

2.<p><b>Стек </b></p>

3.<p><b>Графы </b></p>

4.<p><b>Дек </b></p>

Ответ:3

Комментарий: SD001212

<p><b>Что используется для представления нелинейных структур данных? </b></p>

1.<p><b>Рекурсии </b></p>

2.<p><b>Массив </b></p>

3.<p><b>Списковые структуры данных </b></p>

4.<p><b>Пирамиды </b></p>

Ответ:1

Комментарий: SD001213

<p><b>Как называют процедуру, которая прямо или косвенно обращается к самой себе? </b></p>

1.<p><b>Циклом </b></p>

2.<p><b>Рекурсивной </b></p>

3.<p><b>Поступающая </b></p>

4.<p><b>Ветвящаяся </b></p>

Ответ:2

Комментарий: SD001214

<p><b>Дерево определяется как… </b></p>

1.<p><b>Иерархическая структура </b></p>

2.<p><b>Дивизиональная структура </b></p>

3.<p><b>Матричная структура </b></p>

4.<p><b>Функционально-линейная структура? </b></p>

Ответ:1

Комментарий: SD001215

<p><b>Какой узел имеется на верхнем уровне иерархии дерева? </b></p>

1.<p><b>Вершина дерева </b></p>

2.<p><b>Корень дерева </b></p>

3.<p><b>Ствол дерева </b></p>

4.<p><b>Главная ветвь дерева </b></p>

Ответ:2

Комментарий: SD001216

<p><b>Как называется наибольшая длина пути корня до листьев дерева? </b></p>

1.<p><b>Уровень дерева </b></p>

2.<p><b>Ширина дерева </b></p>

3.<p><b>Длина дерева </b></p>

4.<p><b>Высота дерева </b></p>

Ответ:4

Комментарий: SD001217

<p><b>Как называется дерево, если на каждом уровне задан порядок следования вершин </b></p>?

1.<p><b>Разветвлённым деревом </b></p>

2.<p><b>Упорядоченным деревом </b></p>

3.<p><b>Бинарным деревом </b></p>

4.<p><b>Тернарное дерево </b></p>

Ответ:2

Комментарий: SD001218

<p><b>Как называется множество, состоящее из некоторого числа непересекающихся деревьев? </b></p>

1.<p><b>Рощей </b></p>

2.<p><b>Лесом </b></p>

3.<p><b>Бор </b></p>

4.<p><b>Купа </b></p>

Ответ:2

Комментарий: SD001219

<p><b>Бинарное дерево идеально сбалансировано, если для каждого его узла количество узлов в левом и правом поддеревьях различается не более чем на… </b></p>

1.<p><b>3 </b></p>

2.<p><b>2 </b></p>

3.<p><b>5 </b></p>

4.<p><b>1 </b></p>

Ответ:4

Комментарий: SD001220

<p><b>На что возвращает функция tree? </b></p>

1.<p><b>На корень построенного дерева </b></p>

1.<p><b>На листву дерева </b></p>

2.<p><b>На идеально сбалансированное дерево </b></p>

3.<p><b>На бинарное дерево </b></p>

Ответ:1

Комментарий: SD001221

<p><b>Длина поиска в двоичном дереве поиска определяется… </b></p>

1.<p><b>Шириной самого дерева </b></p>

2.<p><b>Корнем самого дерева </b></p>

3.<p><b>Высотой самого дерева </b></p>

4.<p><b>Листвой самого дерева </b></p>

Ответ:3

Комментарий: SD001222

<p><b>Что называется АВЛ – деревом? </b></p>

1.<p><b>Дерево, у которого высота поддеревьев для каждой вершины различается не более чем на 1 </b></p>

2.<p><b>Дерево, у которого высота поддеревьев для каждой вершины различается не более чем на 3 </b></p>

3.<p><b>Дерево, у которого ширина поддеревьев для каждой вершины различается не более чем на 1 </b></p>

4.<p><b>Дерево, у которого ширина поддеревьев для каждой вершины различается не более чем на 2 </b></p>

Ответ:1

Комментарий: AP001223

<p><b>Какие операции можно проводить над деревьями? </b></p>

1.<p><b>Поиск узла с заданным ключом </b></p>

2.<p><b>Добавление нового узла </b></p>

3.<p><b>Удаление узла (поддерева) </b></p>

4.<p><b>Все варианты верны </b></p>

Ответ:4

Комментарий: SD001224

<p><b>Что такое граф? </b></p>

1.<p><b>Это сложная нелинейная многосвязная динамическая структура, отображающая свойства и связи сложного объекта </b></p>

2.<p><b>Это сложная линейная многосвязная динамическая структура, отображающая свойства и связи сложного объекта </b></p>

3.<p><b>Это сложная нелинейная многосвязная статическая структура, отображающая свойства и связи сложного объекта </b></p>

4.<p><b>Линейные структуры, в которых операции включения/исключения элемента могут выполняться как с одного, так и с другого конца. </b></p>

Ответ:1

Комментарий: SD001225

<p><b>Граф, все связи которого ориентированные, называется… </b></p>

1.<p><b>Нуль-графом </b></p>

2.<p><b>Ориентированным </b></p>

3.<p><b>Неориентированным </b></p>

4.<p><b>Смешанным </b></p>

Ответ:2

Комментарий: SD001226

<p><b>Как называется число ребер, для ориентированного графа, входящих в узел? </b></p>

1.<p><b>Полустепенью захода узла </b></p>

2.<p><b>Полустепенью исхода узла </b></p>

3.<p><b>Полустепенью восхода узла </b></p>

4.<p><b>Полустепенью узла </b></p>

Ответ:1

Комментарий: SD001227

<p><b>Как называется граф со связями обоих типов (ориентированной и неориентированной) </b></p>

1.<p><b>Мультиграфом </b></p>

2.<p><b>Простым </b></p>

3.<p><b>Смешанным </b></p>

4.<p><b>Сложным </b></p>

Ответ:3

Комментарий: SD001228

<p><b>Как называется граф без ребер? </b></p>

1.<p><b>Простым </b></p>

2.<p><b>Обычным </b></p>

3.<p><b>Нуль-графом </b></p>

4.<p><b>Безрёберным </b></p>

Ответ:3

Комментарий: SD001229

<p><b>Как называется граф имеющий параллельные ребра? </b></p>

1.<p><b>Нуль-графом </b></p>

2.<p><b>Мультиграфом </b></p>

3.<p><b>Смешанным </b></p>

4.<p><b>Скрещенным </b></p>

Ответ:2

Комментарий: SD001230

<p><b>Что такое сплетения(многосвязные списк)? </b></p>

1.<p><b>Нелинейная структура данных, объединяющая такие понятия как деревья, графы и списковые структуры </b></p>

2.<p><b>Линейная структура данных, объединяющая такие понятия как стек, очередь и дек </b></p>

3.<p><b>Это последовательность, в которой включение и исключение элемента осуществляется с одной стороны этой последовательности. </b></p>

4.<p><b>Линейная структура (последовательность), в которой операции включения/исключения элемента могут выполняться как с одного, так и с другого конца. </b></p>

Ответ:1

Комментарий: SD001231

<p><b>Чем обычно представлены графы? </b></p>

1.<p><b>Весовыми матрицами </b></p>

2.<p><b>Списками смежности </b></p>

3.<p><b>Матрицами смежности </b></p>

4.<p><b>Варианты 2 и 3 </b></p>

Ответ:4

Комментарий: SD001232

<p><b>Что собой представляет список смежности? </b></p>

1.<p><b>Это двумерный (2D) массив V x V вершин </b></p>

2.<p><b>Это граф в виде массива связанного списка </b></p>

3.<p><b>Это граф в виде одномерного массива </b></p>

4.<p><b>Это граф в виде трёхмерного массива </b></p>

Ответ:2

Комментарий: SD001233

<p><b>Как по-другому называют многосвязные списки? </b></p>

1.<p><b>Плексы </b></p>

2.<p><b>Узел </b></p>

3.<p><b>Дек </b></p>

4.<p><b>Сеть </b></p>

Ответ:1

Комментарий: SD001234

<p><b>Что называется B-деревом? </b></p>

1.<p><b>Сильно ветвящееся дерево степени n+1 </b></p>

2.<p><b>Сильно ветвящееся дерево степени 2n+1 </b></p>

3.<p><b>Сильно ветвящееся дерево степени 3n+1 </b></p>

4.<p><b>Сильно ветвящееся дерево степени 4n+1 </b></p>

Ответ:2

Комментарий: SD001235

<p><b>Что такое путь в графе? </b></p>

1.<p><b>Это последовательность узлов, связанных ребрами </b></p>

2.<p><b>Это последовательность ребер, связанных узлами </b></p>

3.<p><b>Путь от корня до узла </b></p>

4.<p><b>Путь от корня до листвы </b></p>

Ответ:1

Комментарий: SD001236

<p><b>Какой путь в графе называется элементарным? </b></p>

1.<p><b>В котором все рёбра одинаковые </b></p>

2.<p><b>В котором все вершины различны </b></p>

3.<p><b>В котором все ребра различные </b></p>

4.<p><b>В котором все вершины одинаковые </b></p>

Ответ:3

Комментарий: SD001237

<p><b>Как называется путь от узла к самому себе? </b></p>

1.<p><b>Циклом </b></p>

2.<p><b>Простым </b></p>

3.<p><b>Инцидентным </b></p>

4.<p><b>Разветвлённым </b></p>

Ответ:1

Комментарий: SD001238

<p><b>Что такое B\*-дерево? </b></p>

1.<p><b>Разновидность [B-дерева](https://ru.wikipedia.org/wiki/B-%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE), в которой каждый узел дерева заполнен не менее чем на ½ </b></p>

2.<p><b>Разновидность [B-дерева](https://ru.wikipedia.org/wiki/B-%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE), в которой каждый узел дерева заполнен не менее чем на 1/3 </b></p>

3.<p><b>Разновидность [B-дерева](https://ru.wikipedia.org/wiki/B-%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE), в которой каждый узел дерева заполнен не менее чем на 2/3 </b></p>

4.<p><b>Разновидность [B-дерева](https://ru.wikipedia.org/wiki/B-%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE), в которой каждый узел дерева заполнен не менее чем на ¼ </b></p>

Ответ:3

Комментарий: SD001239

<p><b>Какой граф называется простым? </b></p>

1.<p><b>Граф, в котором нет петель и кратных рёбер </b></p>

2.<p><b>Граф, в котором есть петли и кратные ребра </b></p>

3.<p><b>Граф со связями одного типа </b></p>

4.<p><b>Граф имеющий параллельные ребра </b></p>

Ответ:1

Комментарий: SD001240

**<p><b>По какому принципу построена Очередь?</b></p>**

1. <p><b> last in – last out </b></p>

2. <p><b>first in – first out</b></p>

3. <p><b>first in – last out</b></p>

Ответ: 1

Комментарий: SD001121

**<p><b>Какая библиотека нужна для работы с очередью?</b></p>**

1. <p><b>queue</b></p>
2. <p><b>qeueu</b></p>
3. <p><b>qeeue</b></p>

Ответ: 1

Комментарий: SD001122

**<p><b>Как удалить второй элемент очереди?</b></p>**

1. <p><b>Чтобы удалить второй элемент очереди нужно сначала удалить первый.</b></p>
2. <p><b>Чтобы удалить второй элемент очереди нужно сначала удалить третий.</b></p>
3. <p><b>Чтобы удалить второй элемент очереди нужно сначала удалить очередь.</b></p>

Ответ: 1

Комментарий: SD001123

**<p><b>Какие значения может принимать функция empty?</b></p>**

1. <p><b> true </b></p>

2. <p><b> false </b></p>

3. <p><b> true & false </b></p>

Ответ: 3

Комментарий: SD001124

**<p><b>Функция для удаления первого элемента очереди?</p></b>**

1.<p><b>puch() </b></p>

2.<p><b> pop()</b></p>

3.<p><b> front()</b></p>

Ответ: 2

Комментарий: SD001125

**<p><b>Конструкция объявления очереди?</b></p>**

1.<p><b> queue <тип данных> <имя> </b></p>

2.<p><b> qeueu <тип данных> <имя> </b></p>

3.<p><b> queue <имя><тип данных></b></p>

Ответ: 1

Комментарий: SD001126

**<p><b>На основе скольких стеков могут быть реализованы очереди?</b></p>**

1.<p><b>Одного </b></p>

2.<p><b>Трех </b></p>

3.<p><b>Двух </b></p>

Ответ: 3

Комментарий: SD001127

**<p><b>Что такое очередь?</b></p>**

1.<p><b>Очередь это упорядоченный набор элементов, которые могут удаляться с её начала и помещаться в её конец.</b></p>

2. .<p><b> Упорядоченный набор элементов, которые могут удаляться с её конца и помещаться в её конец. </b></p>

3. <p><b> Это динамическая структура данных которая состоит из набора элементов которые размещены хаотично</b></p>

Ответ: 1

Комментарий: SD001128

**<p><b>Что такое STL?</b></p>**

1. <p><b>Стандартная библиотека шаблонов в языке программирования C++ </b></p>
2. <p><b>Стандартная библиотека шаблонов в языке программирования С# </b></p>
3. <p><b>Стандартная библиотека шаблонов в языке программирования JS </b></p>

Ответ: 1

Комментарий: SD001129

**<p><b>Что такое список?</b></p>**

1. <p><b>Линейная последовательность элементов, каждый из которых содержит указатели на своих соседей</b></p>
2. <p><b>Линейная последовательность элементов, каждый из которых содержит указатели на все элементы очереди</b></p>
3. <p><b>Линейная последовательность элементов, каждый из которых содержит указатели на предыдущие элементы очереди </b></p>

Ответ: 1

Комментарий: SD001130

**<p><b>Какие функции нужны для работы с очередью?</b></p>**

1. <p><b>back, front , empty</b></p>
2. <p><b>pop, first, back</b></p>
3. <p><b>last, push, pop </b></p>

Ответ: 1

Комментарий: SD001131

**<p><b>Как обратиться к первому элементу очереди?</b></p>**

1. <p><b> front() </b></p>
2. <p><b> back() </b></p>
3. <p><b> empty() </b></p>

Ответ: 1

Комментарий: SD001132

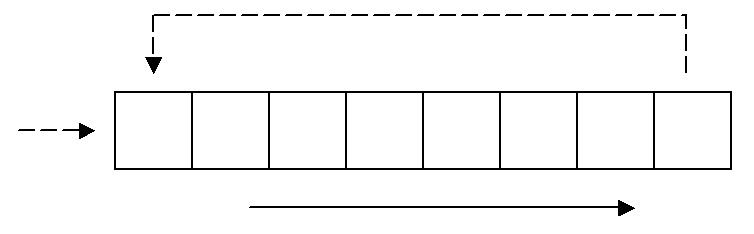
**<p><b>Что такое Дэк?</b></p>**

1. <p><b>Однонаправленный связный список</b></p>
2. <p><b>Двунаправленный связный список </b></p>
3. <p><b>Нет правильного ответа</b></p>

Ответ: 1

Комментарий: SD001133

**<p><b>Какая очредь преставлена на рисунке?</b></p>**



1. <p><b>Кольцевая очередь</b></p>
2. <p><b>Очередь с приоритетом</b></p>
3. <p><b>Простая очередь</b></p>

Ответ: 1

Комментарий: SD001134

**<p><b>Сколько существует способов реализации очереди?</b></p>**

1. <p><b>3</b></p>
2. <p><b>4</b></p>
3. <p><b>5</b></p>

Ответ : 2

Комментарий: SD001135

**<p><b>Что выводит данный код?</b></p>**

queue<float> que1 ({1.1, 2.2, 3.3, 4.4, 5.5});<br>  
cout << que1.empty() << ‘\n‘;<br>  
queue<float> que2;<br>  
cout << que2.empty() << ‘\n‘<br>

1. <p><b> 1 </b></p>
2. <p><b> 3 </b></p>
3. <p><b> 1.1 </b></p>

Ответ: 1

Комментарий: SD001136

**<p><b>Что выводит данный код?</b></p>**

queue<float> que;<br>  
que.push(1.1); que.push(2.2); que.push(3.3); que.push(4.4); que.push(5.5);<br>  
cout << que.size() << ‘\n‘;<br>

1. <p><b> 4 </b></p>
2. <p><b> 7 </b></p>
3. <p><b> 5 </b></p>

Ответ: 3

Комментарий: SD001137

**<p><b>Что выводит данный код?</b></p>**

     int sum = 0;

  queue<int> myqueue;<br>

    myqueue.push(1);<br>

    myqueue.push(8);<br>

    myqueue.push(3);<br>

    myqueue.push(6);<br>

    myqueue.push(2);<br>

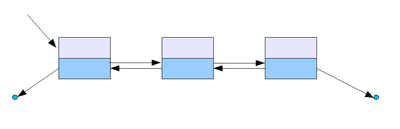
    cout << myqueue.size();<br>

1. <p><b> 6 </b></p>
2. <p><b> 5 </b></p>
3. <p><b> 20 </b></p>

Ответ: 3

Комментарий: SD001138

**<p><b>Какой список изображен на фото?</b></p>**



1. <p><b> Развернутый связный список</b></p>
2. <p><b> Двусвязный список </b></p>
3. <p><b> Список с пропусками </b></p>

Ответ: 2

Комментарий: SD001139

**<p><b>Сколько существует способов реализации очереди?</b></p>**

1. <p><b> 4 </b></p>
2. <p><b> 5 </b></p>
3. <p><b> 3 </b></p>

Ответ: 1

Комментарий: SD001140

**<p><b>Чтобы удалить элемент из очереди реализованной при помощи массива</b></p>**

1. <p><b>уменьшить переменную </b></p>
2. <p><b>pop() </b></p>
3. <p><b> увеличить переменную</b></p>

Ответ: 1

Комментарий: SD001141

**<p><b>Как работает циклическая очередь? </b></p>**

1. <p><b> При достижении конца очереди отсчет начинается сначала</b></p>

1. <p><b> При достижении начало очереди отсчет начинается с конца</b></p>
2. <p><b>При достижении конца цикл заканчивается </b></p>

Ответ: 1

Комментарий: SD001142

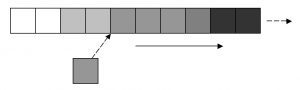
**<p><b>Что делает функция size() </b></p>**

1. <p><b>Проверяет пуста ли очередь</b></p>
2. <p><b>Возвращает количество элементов в контейнере </b></p>
3. <p><b>Возвращает последний элемент </b></p>

Ответ: 2

Комментарий: SD001143

**<p><b>Какая очередь изображена на картинке?</b></p>**

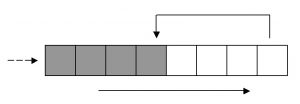


1. <p><b>Очередь с приоритетом</b></p>
2. <p><b>Кольцевая очередь</b></p>
3. <p><b>Простая очередь</b></p>

Ответ: 1

Комментарий: SD001144

**<p><b> Какая очередь изображена на картинке?</b></p>**

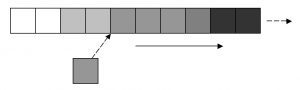


1. <p><b>Кольцевая очередь , когда часть элементов занята</b></p>
2. <p><b>Простая очередь , когда часть элементов занята </b></p>
3. <p><b> Очередь с приоритетом, когда часть элементов занята </b></p>

Ответ: 1

Комментарий: SD001145

**<p><b> Какая очередь изображена на картинке?</b></p>**



1. <p><b> Очередь с приоритетным включением </b></p>
2. <p><b> Очередь с приоритетным исключением </b></p>
3. <p><b> Очередь с приоритетом </b></p>

Ответ: 1

Комментарий: SD001146

**<p><b> Какая очередь изображена на картинке?</b></p>**

Очередь с приоритетным исключением. Элемент добавляется в конец очереди, вытягивается в соответствии с приоритетом

1. <p><b> Очередь с приоритетом </b></p>
2. <p><b> Очередь с приоритетным исключением </b></p>
3. <p><b> Очередь с приоритетным включением </b></p>

Ответ: 2

Комментарий: SD001147

**<p><b>Чтобы удалить элемент из очереди реализованной при помощи массива</b></p>**

1. <p><b>уменьшить переменную </b></p>
2. <p><b>pop() </b></p>
3. <p><b> увеличить переменную</b></p>

Ответ: 1

Комментарий: SD001148

**<p><b>Что такое STL?</b></p>**

1. <p><b>Стандартная библиотека шаблонов в языке программирования C++ </b></p>
2. <p><b>Стандартная библиотека шаблонов в языке программирования С# </b></p>
3. <p><b>Стандартная библиотека шаблонов в языке программирования JS </b></p>

Ответ: 1

Комментарий: SD001149

**<p><b>Какая бибылиотека нужна для работы с очередью?</b></p>**

1. <p><b>queue</b></p>
2. <p><b>qeueu</b></p>
3. <p><b>qeeue</b></p>

Ответ: 1

Комментарий: SD001150

<p><b> Связный список — это … </b></p>

1. <p><b> динамическая структура данных. </b></p>

2. <p><b> статическая структура данных. </b></p>

3. <p><b> список, содержащий введённые нами с клавиатуры данными. </b></p>

4. <p><b> строковый тип данных, в котором каждая строка введённая с клавиатуры, записывается в динамический массив. </b></p>

Ответ: 1

Комментарий: AP00301

<p><b> Какие существуют разновидности связных списков? </b></p>

1. <p><b> односвязный(однонаправленный), двусвязный(двунаправленный) и многосвязный. </b></p>

2. <p><b> простой список и сложный. </b></p>

3. <p><b> статический и динамический. </b></p>

4. <p><b> нет правильного варианта ответа. </b></p>

Ответ: 1

Комментарий: AP00302

<p><b> По способу организации связей (или по архитектуре) список может быть … </b></p>

1. <p><b> линейным, кольцевым (циклическим) и ветвящийся. </b></p>

2. <p><b> только линейным. </b></p>

3. <p><b> только кольцевым и ветвящимся. </b></p>

4. <p><b> линейным, кольцевым (циклическим). </b></p>

Ответ: 1

Комментарий: AP00303

<p><b> По степени упорядоченности хранимых данных списки могут быть … </b></p>

1. <p><b> нет правильного варианта ответа. </b></p>

2. <p><b> указательными и соеденительными. </b></p>

3. <p><b> легко читаемыми и сложно читаемыми. </b></p>

4. <p><b> упорядоченными и неупорядоченными. </b></p>

Ответ: 4

Комментарий: AP00304

<p><b> Для списков, по сравнению с очередями и стеками, имеется значительно … </b></p>

1. <p><b> меньше операций. </b></p>

2. <p><b> больше операций. </b></p>

Ответ: 2

Комментарий: AP00305

<p><b> Односвязный список состоит из … </b></p>

1. <p><b> динамического массива и значений каждого его элемента. </b></p>

2. <p><b> узлов, которые содержат в себе три значения: первое — это тип переменной, второе — это значение переменной, второе — это указатель. </b></p>

3. <p><b> узлов, которые содержат в себе два значения: первое — это какое-либо данное, а второе — это указатель. </b></p>

4. <p><b> Из элементов, которые объединяются в список. </b></p>

Ответ: 3

Комментарий: AP00306

<p><b> Указатель в односвязном связном списке указывает на … </b></p>

1. <p><b> на название списка. </b></p>

2. <p><b> последний узел связанного списка. </b></p>

3. <p><b> предыдущий узел связанного списка. </b></p>

4. <p><b> следующий узел связанного списка. </b></p>

Ответ: 4

Комментарий: AP00307

<p><b> Указатель последнего узла должен … </b></p>

1. <p><b> указывать на первый узел связного списка. </b></p>

2. <p><b> всегда выставляться в NULL. </b></p>

3. <p><b> должен быть пустым. </b></p>

4. <p><b> нет правильного варианта ответа. </b></p>

Ответ: 2

Комментарий: AP00308

<p><b> Односвязный список, в сравнении с динамическим массивом, характеризуется … </b></p>

1. <p><b> менее быстрыми операциями с изменением количества элементов и менее медленной работой при получении данных их этих элементов.

2. <p><b> более быстрыми операциями с изменением количества элементов и более медленной работой при получении данных их этих элементов.

3. <p><b> менее точным сохранением данных, и требует меньше оперативной памяти.

4. <p><b> более точным сохранением данных, и требует больше оперативной памяти

Ответ: 2

Комментарий: AP00309

<p><b> Что необходимо, чтобы удалить узел в односвязном списке? </b></p>

1. <p><b> Присвоить указателю удаляемого узла значение NULL. </b></p>

2. <p><b> Указатель следующего узла заменить на узел, который идёт перед удаляемым узлом. </b></p>

3. <p><b> Присвоить указателю удаляемого узла значение DEL. </b></p>

4. <p><b> Указатель предыдущего узла заменить на узел, который идёт после удаляемого узла. </b></p>

Ответ: 4

Комментарий: AP00310

<p><b> Как осуществить печать односвязного списка? </b></p>

1. <p><b> Операция печати списка заключается в последовательном просмотре всех элементов списка и выводе их значений на экран. Для обработки списка организуется специальная функция, которая считывает все данные содержащиеся в узлах. </b></p>

2. <p><b> Чтобы осуществить печать односвязного списка нам необходимо пройти по всем указателям и вывести их на экран. </b></p>

3. <p><b> Операция печати списка заключается в последовательном просмотре всех элементов списка и выводе их значений на экран. Для обработки списка организуется функция, в которой нужно переставлять указатель на следующий элемент списка до тех пор, пока указатель не станет равен NULL, то есть будет достигнут конец списка. </b></p>

4. <p><b> Чтобы осуществить печать односвязного списка нам необходимо применить функцию, которая выводит на экран указатели. </b></p>

Ответ: 3

Комментарий: AP00311

<p><b> Взаимообмен узлов односвязного списка осуществляется … </b></p>

1. <p><b> путем удаления и добавления соответствующих узлов. </b></p>

2. <p><b> путем переустановки узлов. </b></p>

3. <p><b> путём переустановки данных. </b></p>

4. <p><b> путем переустановки указателей. </b></p>

Ответ: 4

Комментарий: AP00312

<p><b> Что необходимо, чтобы добавить узел в середине односвязного списка? </b></p>

1. <p><b> переустановить указатель узла, предшествующего добавляемому, на добавляемый узел, а указатель добавляемого узла установить на следующий узел. </b></p>

2. <p><b> переустановить указатель узла, добавляемого предшествующему, на добавляемый узел, а указатель следующего узла установить на добавляемый узел. </b></p>

3. <p><b> указатель последнего узла изменить на указатель добавляемого узла, а в добавленном узле указателю присвоить значение NULL. </b></p>

4. <p><b> Нет правильного варианта ответа. </b></p>

Ответ: 1

Комментарий: AP00313

<p><b> Выберете правильное утверждение: </b></p>

1. <p><b> Чтобы получить доступ к какому либо элементу односвязного списка, нужно идти от самого первого элемента к элементу, к которому мы хотим получить доступ. </b></p>

2. <p><b> Чтобы получить доступ к какому либо элементу односвязного списка, нужно идти либо от первого элемента , либо от последнего, в зависимости какой ближе. </b></p>

3. <p><b> Нам необходимо указать номер элемента в списке, к которому мы хотим получить доступ. </b></p>

4. <p><b> Чтобы получить доступ к какому либо элементу односвязного списка, нужно идти о т самого последнего элемента к элементу, к которому мы хотим получить доступ. </b></p>

Ответ: 1

Комментарий: AP00314

<p><b> Выберете преимущество односвязного списка по сравнению с связными динамическими структурами данных: </b></p>

1. <p><b> Быстрый доступ к любому узлу списка. </b></p>

2. <p><b> Меньший расход памяти и простота операций. </b></p>

3. <p><b> Нет правильного варианта ответа. </b></p>

Ответ: 2

Комментарий: AP00315

<p><b> Что означает данный код? </b></p>

<p><b>

struct Link2<br>

{<br>

int data; <br>

Link2\* next, \* prev; <br>

};<br>

</b></p>

1. <p><b> Тип узла односвязного списка. </b></p>

2. <p><b> Тип массива. </b></p>

3. <p><b> Тип переменной. </b></p>

4. <p><b> Тип узла двусвязного списка. </b></p>

Ответ: 1

Комментарий: AP00316

<p><b> Что осуществляет данный код? </b></p>

<p><b>

struct Link1<br>

{<br>

int data; <br>

Link1\* next; <br>

};<br>

void Insert1(Link1\* link, int data) <br>

{<br>

Link1\* q = new Link1; <br>

q->data = data; <br>

q->next = link->next; <br>

link->next = q; <br>

}<br>

</b></p>

1. <p><b> Создание линейного односвязного списка. </b></p>

2. <p><b> Добавление узла в односвязном списке. </b></p>

3. <p><b> Удаление линейного односвязного списка. </b></p>

4. <p><b> Просмотр линейного односвязного списка. </b></p>

Ответ: 2

Комментарий: AP00317

<p><b> Что осуществляет данный код? </b></p>

struct Link1<br>

{<br>

int data; <br>

Link1\* next; <br>

};<br>

void Show(Link1\* link) <br>

{<br>

Link1\* q = link->next; <br>

while (q) <br>

{<br>

cout << q->data << ' '; <br>

q = q->next; <br>

}<br>

cout << endl; <br>

}<br>

1. <p><b> Создание линейного односвязного списка. </b></p>

2. <p><b> Добавление узла в линейный односвязный список. </b></p>

3. <p><b> Удаление линейного односвязного списка. </b></p>

4. <p><b> Просмотр линейного односвязного списка. </b></p>

Ответ: 4

Комментарий: AP00318

<p><b> Что осуществляет данный код? </b></p>

<p><b>

struct Link1<br>

{<br>

int data; <br>

Link1\* next; <br>

};<br>

void Delete1(Link1\* link) <br>

{<br>

Link1\* q; <br>

if (link->next) <br>

{<br>

q = link->next<br>

link->next = q->next; <br>

delete q; <br>

}<br>

}<br>

</b></p>

1. <p><b> Удаление узла в линейном односвязном списке. </b></p>

2. <p><b> Добавление узла в линейный односвязный список. </b></p>

3. <p><b> Удаление узла в линейном двусвязном списке. </b></p>

4. <p><b> Добавление узла в линейный двусвязный список. </b></p>

Ответ: 1

Комментарий: AP00319

<p><b> Сколько указателей содержит каждый узел двусвязного линейного списка? </b></p>

1. <p><b> 1. </b></p>

2. <p><b> 2. </b></p>

3. <p><b> 3. </b></p>

4. <p><b> 4. </b></p>

Ответ: 2

Комментарий: AP00320

<p><b> Преимущества двусвязного списка: </b></p>

1. <p><b> проще выполняются некоторые операции (например, удаление) </b></p>

2. <p><b> есть возможность перестроить поврежденный список и проще выполняются некоторые операции (например, удаление). </b></p>

3. <p><b> есть только возможность перестроить поврежденный список. </b></p>

4. <p><b> есть возможность перестроить поврежденный список ,но сложнее выполняются некоторые операции (например, удаление). </b></p>

Ответ: 2

Комментарий: AP00321

<p><b> Выберете правильное утверждение: </b></p>

1. <p><b> Чтобы получить доступ к какому либо элементу односвязного списка, нужно идти только от самого последнего элемента к элементу, к которому мы хотим получить доступ. </b></p>

2. <p><b> Нам необходимо указать номер элемента в списке, к которому мы хотим получить доступ. </b></p>

3. <p><b> Чтобы получить доступ к какому либо элементу односвязного списка, нужно идти от самого первого элемента к элементу, к которому мы хотим получить доступ. </b></p>

4. <p><b> Чтобы получить доступ к какому либо элементу многосвязного списка, можно идти от самого последнего элемента или от самого первого к элементу, к которому мы хотим получить доступ. </b></p>

Ответ: 4

Комментарий: AP00322

<p><b> Что содержат указатель(и) каждого узла двусвязного списка? </b></p>

1. <p><b> Первый указатель содержит адрес на предыдущий узел, а второй – на следующий узел. </b></p>

2. <p><b> Первый указатель содержит адрес на предыдущий узел, второй – на следующий узел, третий – на последний узел списка. </b></p>

3. <p><b> Первый указатель содержит адрес на предыдущий узел, второй – на следующий узел, третий – на последний узел списка, четвёртый – на первый узел списка. </b></p>

4. <p><b> Указатель содержит адрес предыдущего узла. </b></p>

Ответ: 1

Комментарий: AP00323

<p><b> Что означает данный код? </b></p>

<p><b>

struct Link2<br>

{<br>

int data; <br>

Link2\* next, \* prev; <br>

};<br>

</b></p>

1. <p><b> Тип узла односвязного списка. </b></p>

2. <p><b> Тип массива. </b></p>

3. <p><b> Тип переменной. </b></p>

4. <p><b> Тип узла двусвязного списка. </b></p>

Ответ: 4

Комментарий: AP00324

<p><b> Что осуществляет данный код? </b></p>

<p><b>

void Delete2(Link2\* del) <br>

{<br>

del->prev->next = del->next; <br>

if (del->next) <br>

del->next->prev = del->prev; <br>

delete del; <br>

}<br>

</b></p>

1. <p><b> Добавление узла в линейном двусвязном списке. </b></p>

2. <p><b> Удаление узла в линейном двусвязном списке. </b></p>

3. <p><b> Создание линейного двусвязного списка. </b></p>

4. <p><b> Просмотр линейного двусвязного списка. </b></p>

Ответ: 2

Комментарий: AP00325

<p><b> Что осуществляет данный код? </b></p>

<p><b>

void Insert2(Link2\* St, int data) <br>

{<br>

Link2\* q = new Link2; <br>

q->data = data; <br>

q->next = St->next; <br>

q->prev = St; <br>

St->next = q; <br>

if (q->next) <br>

q->next->prev = q; <br>

}<br>

</b></p>

1. <p><b> Добавление узла в линейном двусвязном списке. </b></p>

2. <p><b> Удаление узла в линейном двусвязном списке. </b></p>

3. <p><b> Создание линейного двусвязного списка. </b></p>

4. <p><b> Просмотр линейного двусвязного списка. </b></p>

Ответ: 1

Комментарий: AP00326

<p><b> Что осуществляет данный код? </b></p>

<p><b>

int Search2(Link2\* Start, Link2\*& Find, int Key) <br>

{<br>

Link2\* Cur = Start->next; <br>

int Success = 0; <br>

while (Cur && !Success) <br>

{<br>

if (Cur->data == Key) <br>

{<br>

Find = Cur; <br>

Success = 1; <br>

break; <br>

}<br>

Cur = Cur->next; <br>

}<br>

return Success; <br>

}<br>

</b></p>

1. <p><b> Удаление узла в линейном двусвязном списке. </b></p>

2. <p><b> Просмотр линейного двусвязного списка. </b></p>

3. <p><b> Создание линейного двусвязного списка. </b></p>

4. <p><b> Поиск в линейном двусвязном списке. </b></p>

Ответ: 4

Комментарий: AP00327

<p><b> Если значение указателя последнего звена линейного односвязного списка заменить с NULL на адрес ведущего звена, то линейный список превратится в … </b></p>

1. <p><b> односвязный ветвящийся список. </b></p>

2. <p><b> двусвязный кольцевой список. </b></p>

3. <p><b> Многосвязный список. </b></p>

4. <p><b> односвязный кольцевой список. </b></p>

Ответ: 4

Комментарий: AP00328

<p><b> Отсутствие "последнего" звена в кольцевом списке приводит к … </b></p>

1. <p><b> еще большему упрощению операций добавления и удаления, по сравнению с 1- и 2-связным линейным списком. </b></p>

2. <p><b> еще большему усложнению операций добавления и удаления, по сравнению с 1- и 2-связным линейным списком. </b></p>

3. <p><b> еще большему усложнению операций поиска, по сравнению с 1- и 2-связным линейным списком. </b></p>

4. <p><b> Нет правильного варианта ответа. </b></p>

Ответ: 1

Комментарий: AP00329

<p><b> Многосвязные списки представляют собой … </b></p>

1. <p><b> совокупность связанных списков. </b></p>

2. <p><b> динамические структуры данных, в основу которых положены 1- или 2-связные списки, в которых имеются дополнительные связи между звеньями. </b></p>

3. <p><b> динамические структуры данных, в основу которых положены только односвязные списки, в которых имеются дополнительные связи между звеньями. </b></p>

4. <p><b> статические структуры данных, в основу которых положены 1- или 2-связные списки, в которых имеются дополнительные связи между звеньями. </b></p>

Ответ: 2

Комментарий: AP00330

**<p><b>С какого числа начинается индекс массива? </p><b>**

1. <p><b>1</p><b>
2. <p><b>0</p><b>
3. <p><b>С любого числа</p><b>
4. <p><b>Нет правильного ответа</p><b>

Ответ:2

Комментарий:SD00101

**<p><b>Какие типы данные мы можем применять для массива? </p><b>**

1. <p><b>int</p><b>
2. <p><b>char</p><b>
3. <p><b>long</p><b>
4. <p><b>Любые типы данных</p><b>

Ответ:4

Комментарий:SD00102

**<p><b>Как элементы массива размещаются в памяти? </p><b>**

1. <p><b>Последовательно </p><b>
2. <p><b>В любом порядке</p><b>
3. <p><b>По возрастанию объема памяти</p><b>
4. <p><b>Нет правильного ответа</p><b>

Ответ:1

Комментарий:SD00103

**<p><b>Какие основные характеристики имеет массив? </p><b>**

1. <p><b>Отдельные объекты данных в массиве называются элементами. Все элементы массива должны иметь одинаковый тип данных. </p><b>
2. <p><b>Все элементы располагаются в памяти компьютера последовательно, и индекс первого элемента равен нулю. </p><b>
3. <p><b>Имя массива является постоянным значением, представляющим собой адрес первого элемента массива. </p><b>
4. <p><b>Все из перечисленных</p><b>

Ответ:4

Комментарий:SD00104

**<p><b>Как объявляется массив? </p><b>**

Ответ: объявление массива начинается с указания его типа, затем следуют его имя и пара квадратных скобок, в которые заключено выражение типа константа, определяющее размер массива.

Комментарий:SD00105

**<p><b>Какими способами обращаться к элементам массива? </p><b>**

1. <p><b>По номеру элемента в массиве (через его индекс); </p><b>
2. <p><b>По указателю. </p><b>
3. <p><b>По имени массива</p><b>
4. <p><b>Все ответы верные</p><b>

Ответ:1,2

Комментарий:SD00106

**<p><b>Как производить инициализацию массивов, содержащих элементы базовых типов? </p><b>**

Ответ: Инициализацию массивов, содержащих элементы базовых типов, можно производить при их объявлении.

Комментарий:SD00107

**<p><b>Способы инициализации массивов? </p><b>**

1. <p><b>По умолчанию во время их создания. Применимо только к глобальным и статическим (static) массивам. </p><b>
2. <p><b>Явно во время создания при помощи констант инициализации. </p><b>
3. <p><b>Во время выполнения программы при присваивании или копировании данных в массив. </p><b>
4. <p><b>Все варианты ответов правильные</p><b>

Ответ:4

Комментарий:SD00108

**<p><b>Какие массивы можно** **инициализировать? </p><b>**

1. <p><b>Внешние</p><b>
2. <p><b>Автоматические</p><b>
3. <p><b>Статические</p><b>
4. <p><b>Регистровые</p><b>

Ответ:1,3

Комментарий:SD00109

**<p><b>Что такое имя массива? </p><b>**

Ответ: Это адрес его первого элемента

Комментарий:SD00110

**<p><b>Что выведет программа на 11 и 12 месяц?**

**#include <stdio.h>**

**int days[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31};**

**main ()**

**{**

**int index;**

**extern int days[]; /\* необязательное описание \*/**

**for (index = 0; index < 12; index++ )**

**printf(" Месяц %d имеет %d дней\n", index + 1, days [index]);**

**}</p><b>**

1. <p><b>Месяц 11 имеет 30 дней, Месяц 12 имеет 31 день</p><b>
2. <p><b>Месяц 11 имеет 0 дней, Месяц 12 имеет 0 дней</p><b>
3. <p><b>Программа выдаст ошибку</p><b>
4. <p><b>Программа ничего не выведет на экран</p><b>

Ответ:2

Комментарий:SD00111

**<p><b>Если в массиве A 132 элемента, каким будет правильное обращение к последнему элементу массива? </p><b>**

1. <p><b>A[132] </p><b>
2. <p><b>A[131] </p><b>
3. <p><b>A[133] </p><b>
4. <p><b>A[-1] </p><b>

Ответ:2

Комментарий:SD00112

**<p><b>Если имеется объявление int a[15], какой фрагмент кода выводит все элементы массива? </p><b>**

1. <p><b>for (int k = 0; k <= 15; k++) cout << a[k]; </p><b>
2. <p><b>for (int k = 1; k < 16; k++) cout << a[k]; </p><b>
3. <p><b>for (int k = 0; k < 15; k++) cout << a[k]; </p><b>

Ответ:3

Комментарий:SD00113

**<p><b>Для получения адреса переменной используется операция</p><b>**

1. <p><b>\*</p><b>
2. <p><b>&</p><b>
3. <p><b>-></p><b>
4. <p><b>Нет правильного ответа</p><b>

Ответ:2

Комментарий:SD00114

**<p><b>Какой массив имеет самый большой размер? </p><b>**

char \*s1= "QWERTY";

char s2[7]= "qwerty";

char \*s3= "127\*27";</p><b>

1. <p><b>Все массивы имеют одинаковый размер</p><b>
2. <p><b>s1</p><b>
3. <p><b>s2</p><b>
4. <p><b>s3</p><b>

Ответ:1

Комментарий:SD00115

**<p><b>В каких местах может размещаться переменная? </p><b>**

1. <p><b>в области данных программы </p><b>
2. <p><b>в стеке</p><b>
3. <p><b>в свободной памяти</p><b>
4. <p><b>Все ответы правильные</p><b>

Ответ:4

Комментарий:SD00116

**<p><b>При помощи каких выражений выделяется динамическая память? </p><b>**

1. <p><b>malloc </p><b>
2. <p><b>calloc</p><b>
3. <p><b>new</p><b>
4. <p><b>Все ответы правильные</p><b>

Ответ:4

Комментарий:SD00117

**<p><b>int mas[22] Что содержит данный массив? </p><b>**

1. <p><b>Массив, содержащий 22 символов</p><b>
2. <p><b>Массив, содержащий 22 целых чисел</p><b>
3. <p><b>Массив, содержащий 22 целых чисел типа long</p><b>
4. <p><b>Нет правильного ответа</p><b>

Ответ:2

Комментарий: SD00118

**<p><b>Выберите все правильные утверждения о массивах в языке C++</p><b>**

1. <p><b>Элементы массива могут быть разных типов</p><b>
2. <p><b>все элементы массива должны быть одного типа</p><b>
3. <p><b>элементы всегда нумеруются с нуля</p><b>
4. <p><b>элементы всегда нумеруются с единицы</p><b>

Ответ:2,3

Комментарий: SD00119

**<p><b>Найдите все верные команды объявления массива Х: </p><b>**

1. <p><b>int X = [1, 2, 5, 8] </p><b>
2. <p><b>int Х[4] = {1, 2, 5, 8}</p><b>
3. <p><b>float X[2] = {2.36, -5}</p><b>
4. <p><b>float X[4] = {2, -5, 8}</p><b>

Ответ:2,3,4

Комментарий: SD00120

**<p><b>Какой индекс имеет последний элемент массива int A[9]? </p><b>**

1. <p><b>8</p><b>
2. <p><b>9</p><b>
3. <p><b>10</p><b>
4. <p><b>7</p><b>

Ответ:1

Комментарий: SD00121

**<p><b>Массив - это ...? </p><b>**

1. <p><b>Массив - это упорядоченные в памяти элементы одного и того же типа, имеющие имя. Доступ к отдельным элементам массива осуществляется по имени массива и индексу</p><b>
2. <p><b>Массив - это упорядоченные в памяти элементы разного типа, имеющие имя. Доступ к отдельным элементам массива осуществляется по имени массива и индексу</p><b>
3. <p><b>Массив - это упорядоченные в памяти элементы одного и того же типа, имеющие имя. Доступ к отдельным элементам массива осуществляется по имени массива и адресу</p><b>

Ответ:1

Комментарий: SD00122

**<p><b>Что программа выведет на экран в результате работы следующего программного кода:**

**#include <iostream>**

**using namespace std;**

**int main() {**

**int A[5] = {4, 56, -5, 8, 2}**

**for (int i = 1; i < 5; i++)**

**cout << A[i] << endl;**

**return 0; </p><b>**

**}**

1. <p><b>4 56 -5 8 2</p><b>
2. <p><b>56 -5 8 2</p><b>
3. <p><b>Ничего, это код для заполнения массива А элементами</p><b>

Ответ:2

Комментарий: SD00123

**<p><b>Выбирите верные N, которые могут опредлять число элементов в массиве**

**float A [N]? </p><b>**

1. <p><b>Int N = 2</p><b>
2. <p><b>Float N = 2.36</p><b>
3. <p><b>Int N = 1 + 1</p><b>
4. <p><b>Int const N = 10</p><b>

Ответ:1,3,4

Комментарий: SD00124

**<p><b>Дайте определение понятию "Двумерный массив"</p><b>**

1. <p><b>это перечисляемый тип данных в виде линейной таблицы</p><b>
2. <p><b>это одномерный массив, тип элементов которого также является массивом (массив массивов) </p><b>
3. <p><b>это структурированный тип данных в виде прямоугольной таблицы</p><b>
4. <p><b>это структурированный комбинированный тип данных, состоящий из фиксированного числа компонент разного типа</p><b>
5. <p><b>это набор данных , структурированных в виде множеств</p><b>

Ответ:3

Комментарий: SD00125

**<p><b>Однородные данные, расположенные в последовательных ячейках оперативной памяти, называются</p><b>**

1. <p><b>Массивы</p><b>
2. <p><b>Терминалы</p><b>
3. <p><b>Стек</p><b>

Ответ:1

Комментарий: SD00126

**<p><b>Двумерные массивы носят название</p><b>**

1. <p><b>Матрицы</p><b>
2. <p><b>Списка</p><b>
3. <p><b>Спецификации</p><b>

Ответ:1

Комментарий: SD00127

**<p><b>Присвоение начальных значений всем элементами массива носит название</p><b>**

1. <p><b>Инициализация</p><b>
2. <p><b>Идентификация</p><b>
3. <p><b>Спецификация</p><b>

Ответ:1

Комментарий: SD00128

**<p><b>Массивы, память под которые выделяется работающей программе по запросам, предусмотренным программистом, называют</p><b>**

1. <p><b>Динамическими</p><b>
2. <p><b>Анализаторными</p><b>
3. <p><b>Модификативными</p><b>

Ответ:1

Комментарий: SD00129

**<p><b>Под какой тип массивов память выделяется при обращении к функции? </p><b>**

1. <p><b>Локальные</p><b>
2. <p><b>Динамические</p><b>
3. <p><b>Статические</p><b>

Ответ:1

Комментарий: SD00130

<p><b>На какие основные категории делятся STL контейнеры(тут несколько вариантов ответа) </b></p>

1. <p><b>Последовательные</b></p>
2. <p><b>Двойные</b></p>
3. <p><b>Параллельные</b></p>
4. <p><b>ассоциативные</b></p>
5. <p><b>адаптеры</b></p>
6. <p><b>соседские</b></p>

Ответ: 1 4 5

Комментарий: SD003001

<p><b>**Класс vector – это**

1. <p><b> это [динамический массив](https://ravesli.com/urok-86-dinamicheskie-massivy/), способный увеличиваться по мере необходимости для содержания всех своих элементов.</b></p>
2. <p><b>это двусторонняя очередь, реализованная в виде динамического массива, который может расти с обоих концов. </b></p>
3. <p><b>это двусвязный список, каждый элемент которого содержит 2 указателя: один указывает на следующий элемент списка.</b></p>
4. <p><b> это контейнерный класс, элементы которого работают по принципу **LIFO** т.е. элементы добавляются (вносятся) в конец контейнера.</b></p>

Ответ: 1

Комментарий: SD003002

<p><b> **Последовательные контейнеры** (или «контейнеры последовательности») — это

1. <p><b> это контейнерные классы, которые автоматически сортируют все свои элементы (в том числе и те, которые добавляете вы)</b></p>
2. <p><b> это специальные предопределенные контейнерные классы, которые адаптированы для выполнения конкретных заданий. </b></p>
3. <p><b> это контейнерные классы, вы можете добавить свой элемент в любое место контейнера. </b></p>
4. <p><b> это контейнерные классы, элементы которых находятся в параллельности. И куда бы вы их не добавили, то они будут параллельны. </b></p>

Ответ: 3

Комментарий: SD003003

<p><b>**Адаптеры** — это </b></p>

1. <p><b> это контейнерные классы, которые автоматически сортируют все свои элементы.</b></p>
2. <p><b> это специальные предопределенные контейнерные классы, которые адаптированы для выполнения конкретных заданий. </b></p>
3. <p><b> это контейнерные классы, элементы которых находятся в последовательности. </b></p>
4. <p><b> это контейнерные классы, элементы которых находятся в параллельности. И куда бы вы их не добавили, то они будут параллельны. </b></p>

Ответ: 2

Комментарий: SD003004

<p><b> **Ассоциативные контейнеры** — это</b></p>

1. <p><b> это это специальные предопределенные контейнерные классы, которые адаптированы для выполнения конкретных заданий. </b></p>
2. <p><b> это контейнерные классы, которые автоматически сортируют все свои элементы.</b></p>
3. <p><b> это контейнерные классы, элементы которых находятся в последовательности. </b></p>
4. <p><b> это контейнерные классы, элементы которых находятся в последовательности. </b></p>

Ответ: 2

Комментарий: SD003005

<p><b> Библиотека STL состоит из следующих частей: </b></p>

1. <p><b> Контейнеры (containers); </b></p>
2. <p><b> Итераторы (iterators); </b></p>
3. <p><b> Алгоритмы (algorithms); </b></p>
4. <p><b> Адаптеры (adaptors); </b></p>
5. <p><b> Байты(bite); </b></p>

Ответ: 1 2 3 4

Комментарий: SD003006

<p><b>Какие из перечисленных вариантов возвращают количество элементов в std::vector? </b></p>

1. <p><b> std::vector::length</b></p>
2. <p><b> std::vector::count</b></p>
3. <p><b> std::vector::capacity</b></p>
4. <p><b> std::vector::size</b></p>

Ответ: 4

Комментарий: SD003007

<p><b> Какие утверждения о предикатах и функторах верны (укажите все подходящие варианты)? </b></p>

1. <p><b> Предикат - частный случай функтора. </b></p>
2. <p><b> Метод operator() функтора может возвращать только значения типа bool. </b></p>
3. <p><b> Предикаты могут использоваться для сортировки элементов в контейнерах. </b></p>

Ответ: 1 3

Комментарий: SD003008

<p><b> Является ли std::string контейнером STL? </b></p>

1. <p><b> ДА</b></p>
2. <p><b>НЕТ</b></p>

Ответ: 2

Комментарий: SD003009

<p><b> Какой код позволяет применить некоторую операцию ко всем элементам последовательности вSTL: </b></p>

1. <p><b> forall</b></p>
2. <p><b> while</b></p>
3. <p><b> for\_each</b></p>
4. <p><b>for\_all</b></p>

Ответ: 3

Комментарий: SD003010

<p><b> Какие контейнеры могут быть реализованы на основе красного-черного дерева: </b></p>

1. <p><b> deque</b></p>
2. <p><b> list </b></p>
3. <p><b> multimap </b></p>
4. <p><b> map </b></p>

Ответ: 3 4

Комментарий: SD003011

<p><b> Какие цели преследовались и каким ценностям отдавалось предпочтение при создании STL? </b></p>

1. <p><b> Максимальная общность</b></p>
2. <p><b> Максимальная эффективность</b></p>
3. <p><b>Максимальная понятность кода STL</b></p>
4. <p><b> Следование семантики работы с передачей по значению</b></p>

Ответ: 1 2 4

Комментарий: SD003012

<p><b> В чём заключается архитектурное свойств STL - гарантии вычислительной сложности? </b></p>

1. <p><b> минимально возможный расход памяти для алгоритма </b></p>
2. <p><b> гарантированное ограничение максимальной сложности работы алгоритма </b></p>
3. <p><b> гарантии достижения минимально-возможной сложности работы алгоритма в каждом конкретном случае </b></p>
4. <p><b> гарантии ограничения максимума расходуемом алгоритмом памяти</b></p>

Ответ: 2

Комментарий: SD003013

<p><b>Какие основные элементы STL инкапсулируют хранение различных значений и объектов? </b></p>

1. <p><b> обобщённые алгоритмы STL </b></p>
2. <p><b> адаптеры STL </b></p>
3. <p><b> итераторы STL </b></p>
4. <p><b> контейнеры STL</b></p>

Ответ: 4

Комментарий: SD003014

<p><b> Kакие основные элементы STL абстрагируют перемещение по коллекциям объектов? </b></p>

1. <p><b> итераторы STL </b></p>
2. <p><b> адаптеры STL </b></p>
3. <p><b> обобщённые алгоритмы STL</b></p>
4. <p><b> контейнеры STL </b></p>

Ответ: 1

Комментарий: SD003015

<p><b> Какие контейнеры STL являются последовательными?(несколько вариантов ответов) </b></p>

1. <p><b> вектор</b></p>
2. <p><b> множество</b></p>
3. <p><b> список</b></p>
4. <p><b> дека </b></p>

Ответ: 1 3 4

Комментарий: SD003016

<p><b> Контейнеры отличаются друг от друга: </b></p>

1. <p><b>временем выполнения операций</b></p>
2. <p><b>интерфейсом</b></p>
3. <p><b>расходом памяти на хранение данных</b></p>
4. <p><b>операциями</b></p>

Ответ: 1

Комментарий: SD003017

<p><b> Какие встроенные типы контейнера являются итераторами? </b></p>

1. <p><b> iterator </b></p>
2. <p><b> pointer</b></p>
3. <p><b> size\_type</b></p>
4. <p><b> const\_reference</b></p>

Ответ: 1 3

Комментарий: SD003018

<p><b>Какие утверждения про итераторы верны? </b></p>

1. <p><b>итераторы - это обобщённые указатели</b></p>
2. <p><b>итераторы предназначены для обхода последовательности объектов в обобщённом контейнере</b></p>
3. <p><b>классы итераторов различаются по быстродействию операций с ними</b></p>
4. <p><b>любой итератор может быть успешно разыменован</b></p>

Ответ: 1 2

Комментарий: SD003019

<p><b> Какому классу итераторов эквивалентны указатели С++?</b></p>

1. <p><b> итераторы входные </b></p>
2. <p><b> итераторы однонаправленные </b></p>
3. <p><b> итераторы двунаправленные </b></p>
4. <p><b> произвольного доступа</b></p>

Ответ: 4

Комментарий: SD003020

<p><b> Самый часто применяемый алгоритм STL</b></p>

1. <p><b> binary\_search</b></p>
2. <p><b> lower\_bound</b></p>
3. <p><b> upper\_bound</b></p>
4. <p><b> sort</b></p>

Ответ: 4

Комментарий: SD003021

<p><b> Kонтейнерный класс, элементы которого работают по принципу **FIFO**</b></p>

1. <p><b> queue</b></p>
2. <p><b> stack</b></p>
3. <p><b> priority\_queue</b></p>
4. <p><b> map</b></p>

Ответ: 1

Комментарий: SD003022

<p><b>Контейнерный класс, элементы которого работают по принципу **LIFO**</b></p>

1. <p><b> queue</b></p>
2. <p><b> stack</b></p>
3. <p><b> priority\_queue</b></p>
4. <p><b> map</b></p>

Ответ: 2

Комментарий: SD003023

<p><b> Тип очереди, в которой все элементы отсортированы </b></p>

1. <p><b> queue</b></p>
2. <p><b> stack </b></p>
3. <p><b> priority\_queue</b></p>
4. <p><b> map</b></p>

Ответ: 3

Комментарий: SD003024

<p><b> Стандартные последовательные контейнеры STL</b></p>

1. <p><b> vector</b></p>
2. <p><b> string</b></p>
3. <p><b> set</b></p>
4. <p><b> multiset</b></p>
5. <p><b>deque</b></p>
6. <p><b> list</b></p>

Ответ: 1 2 5 6

Комментарий: SD003025

<p><b>Cтандартные ассоциативные контейнеры STL</b></p>

1. <p><b> vector</b></p>
2. <p><b> set </b></p>
3. <p><b> multiset</b></p>
4. <p><b> deque</b></p>
5. <p><b>map </b></p>
6. <p><b> multimap</b></p>

Ответ: 2 3 5 6

Комментарий: SD003026

<p><b> Нестандартные ассоциативные контейнеры STL(есть несколько вариантов ответа) </b></p>

1. <p><b> vector </b></p>
2. <p><b> set</b></p>
3. <p><b> hash\_set</b></p>
4. <p><b> hash\_multiset</b></p>
5. <p><b>hash\_map</b></p>
6. <p><b>hash\_multimap</b></p>

Ответ: 3 4 5 6

Комментарий: SD003027

<p><b> Задача: задано число *n*, затем *n* целых чисел, требуется отсортировать их по возрастанию. Поскольку *n* заранее неизвестно, в программе нужно использовать динамические массивы. </b></p>

<p><b> Решит ли код задачу? </b></p>

[#include](https://vk.com/im?sel=221220038&st=%23include) <stdlib.h> <br>

int ReadInt() { <br>

} <br>

int Compare(const void\* a, const void\* b) { <br>

return (\*(int\*)a - \*(int\*)b); <br>

} <br>

int main() { <br>

int n = ReadInt(); <br>

int\* a = (int\*)malloc(sizeof(int) \* n); <br>

for (int i = 0; i < n; ++i) { <br>

a[i] = ReadInt(); <br>

} <br>

qsort(a, n, sizeof(int), Compare); <br>

free(a); <br>

return 0; <br>

1. <p><b> ДА </b></p>
2. <p><b> НЕТ </b></p>

Ответ: 1

Комментарий: SD003028

<p><b> Задача: Пусть дан набор строк, строки поступают на стандартный ввод. Требуется вывести все различные строки в лексикографическом порядке и для каждой указать, сколько раз она встретилась в наборе. </b></p>

<p><b> Решит ли данный код задачу? </b></p>

[#include](https://vk.com/im?sel=221220038&st=%23include) <iostream <br>

int main() { <br>

std::map<std::string, int> words; <br>

std::string word; <br>

while (std::getline(std::cin, word)) { <br>

words[word]++; <br>

} <br>

for (std::map<std::string, int>::const\_iterator it = words.begin(); it != words.end(); it++) { <br>

std::cout « it->first « " - " « it->second « '\n'; <br>

} <br>

return 0; <br>

} <br>

1. <p><b>ДА</b></p>
2. <p><b> НЕТ</b></p>

Ответ: 2

Комментарий: SD003029

<p><b>Как часто std::multimap и std::unordered\_multimap перемешивают записи? </b></p>

1. <p><b> всегда</b></p>
2. <p><b> никогда</b></p>
3. <p><b> часто </b></p>
4. <p><b> когда std::multimap равен нулю </b></p>

Ответ: 2

Комментарий: SD003030

<p><b> Система управления базами данных (СУБД) это…</b></p>

1. <p><b> совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных. </b></p>
2. <p><b> совокупность программных средств только специального назначения, обеспечивающих управление созданием и использованием баз данных. </b></p>
3. <p><b> совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием систем данных. </b></p>
4. <p><b> совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием программ данных. </b></p>

Ответ 1

Комментарий: SD004001

<p><b> СУБД  это…</b></p>

1. <p><b> СУБД — комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными (обновлять, удалять и выбирать).</b></p>
2. <p><b> СУБД — комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными (вставлять, редактировать, удалять и выбирать).</b></p>
3. <p><b> СУБД — комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными (вставлять, обновлять, удалять и выбирать). </b></p>
4. <p><b> СУБД — комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными (вставлять и удалять). </b></p>

Ответ 3

Комментарий: SD004002

<p><b> Основные функции СУБД это….</b></p>

1. <p><b> управление данными во [внешней памяти](https://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B5%D1%88%D0%BD%D1%8F%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) (на дисках) </b></p>
2. <p><b> поддержка языков БД </b></p>
3. <p><b> переработка языков </b></p>
4. <p><b> управление данными в [оперативной памяти](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) с использованием [дискового кэша](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BA%D0%BE%D0%B2%D1%8B%D0%B9_%D0%BA%D1%8D%D1%88) </b></p>

Ответ 1 2 4

Комментарий: SD004003

<p><b> База данных — это средство для </b></p>

1. <p><b> поиска данных </b></p>
2. <p><b> хранения данных </b></p>
3. <p><b> сортировки данных </b></p>
4. <p><b> хранения, поиска и упорядочения данных </b></p>

Ответ 4

Комментарий: SD004004

<p><b> Основные средства СУБД для работы пользователя с базой данных:</b></p>

1. <p><b> алгоритмический язык Паскаль </b></p>
2. <p><b> разрабатываемые пользователем программы </b></p>
3. <p><b> язык запросов </b></p>
4. <p><b> графический интерфейс </b></p>

Ответ 3 4

Комментарий: SD004005

<p><b> СУБД с непосредственной записью </b></p>

1. <p><b> В таких СУБД все изменённые блоки данных незамедлительно записываются во [внешнюю память](https://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B5%D1%88%D0%BD%D1%8F%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) при поступлении сигнала подтверждения любой [транзакции](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D0%BD%D0%B7%D0%B0%D0%BA%D1%86%D0%B8%D0%B8).</b></p>
2. <p><b> В таких СУБД все изменённые блоки данных не записываются моментально</b></p>
3. <p><b> В таких СУБД все изменённые блоки данных незамедлительно записываются во [внешнюю память](https://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B5%D1%88%D0%BD%D1%8F%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) при поступлении сигнала подтверждения любой [транзакции](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D0%BD%D0%B7%D0%B0%D0%BA%D1%86%D0%B8%D0%B8) </b></p>
4. <p><b> В таких СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти </b></p>

Ответ 1

Комментарий: SD004006

<p><b> В таких СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления любого из следующих событий: </b></p>

1. <p><b> [Контрольная точка](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%82%D0%BE%D1%87%D0%BA%D0%B0) </b></p>
2. <p><b> Нехватка оперативной памяти для буферов внешней памяти </b></p>
3. <p><b> Точка с запятой </b></p>
4. <p><b> Точка с черточкой </b></p>

Ответ 1 2

Комментарий: SD004007

<p><b>Сетевая модель данных это…</b></p>

1. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры.</b></p>
2. <p><b>  [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляющая [реляционными базами данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).</b></p>
3. <p><b> логическая [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), являющаяся расширением [иерархического подхода](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), описывающая структурный аспект.</b></p>
4. <p><b> [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), основанная на [объектной модели данных](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1).</b></p>

Ответ 3

Комментарий: SD004008

<p><b> Иерархическая модель данных это…</b></p>

1. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры.</b></p>
2. <p><b>  [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляющая [реляционными базами данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).</b></p>
3. <p><b> логическая [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), являющаяся расширением [иерархического подхода](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), строгая математическая теория. </b></p>
4. <p><b> [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), основанная на [объектной модели данных](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1).</b></p>

Ответ 1

Комментарий: SD004009

<p><b> Объектно-ориентированная классификация это…</b></p>

1. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры. </b></p>
2. <p><b>  [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляющая [реляционными базами данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).</b></p>
3. <p><b> логическая [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), являющаяся расширением [иерархического подхода](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), строгая математическая теория.</b></p>
4. <p><b> [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), основанная на [объектной модели данных](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1).</b></p>

Ответ 4

Комментарий: SD004010

<p><b> Cетевые классификации это…</b></p>

1. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры. </b></p>
2. <p><b>  [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляющая [реляционными базами данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).</b></p>
3. <p><b> логическая [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), являющаяся расширением [иерархического подхода](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), строгая математическая теория.</b></p>
4. <p><b> [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), основанная на [объектной модели данных](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1).</b></p>

Ответ 2

Комментарий: SD004011

<p><b> Реляционные классификации это…</b></p>

1. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры. </b></p>
2. <p><b>  [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), управляющая [реляционными базами данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). Практически все разработчики современных приложений. </b></p>
3. <p><b> логическая [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), являющаяся расширением [иерархического подхода](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), строгая математическая теория. </b></p>
4. <p><b> [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), основанная на [объектной модели данных](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1).</b></p>

Ответ 3

Комментарий: SD004012

<p><b> [Объектно-реляционные](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94) классификации это…</b></p>

1. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры. </b></p>
2. <p><b> [реляционная СУБД](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94), поддерживающая некоторые технологии, присущие [объектно-ориентированным СУБД](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94). </b></p>
3. <p><b> логическая [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), являющаяся расширением [иерархического подхода](https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), строгая математическая теория.</b></p>
4. <p><b> [система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), основанная на [объектной модели данных](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1).</b></p>

Ответ 2

Комментарий: SD004013

1. <p><b>Что такое файл? </b></p>  
   <p><b>Именованный набор байтов, который может быть сохранен на некотором накопителе.</b></p>  
   <p><b>Набор байтов, который может быть сохранен на некотором накопителе.</b></p>  
   <p><b>Именованный набор нулей и единиц, который может быть сохранен на некотором накопителе.</b></p>
2. <p><b> Набор числовых единиц и байтов</b></p>  
   Ответ: 1  
   Комментарий: SD004014

<p><b> СУБД с непосредственной записью это…</b></p>

1. <p><b>СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления нехватки пространства во внешней памяти</b></p>
2. <p><b>СУБД все изменённые блоки данных незамедлительно записываются во [внешнюю память](https://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B5%D1%88%D0%BD%D1%8F%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) при поступлении сигнала подтверждения любой [транзакции](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D0%BD%D0%B7%D0%B0%D0%BA%D1%86%D0%B8%D0%B8). Такая стратегия используется только при высокой эффективности внешней памяти</b></p>
3. <p><b>СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления останов</b></p>
4. <p><b>СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления К[онтрольной точки](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%82%D0%BE%D1%87%D0%BA%D0%B0)</b></p>

Ответ: 2  
Комментарий: SD004015

<p><b> СУБД с отложенной записью это…</b></p>

1. <p><b>СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления нехватки пространства во внешней памяти</b></p>
2. <p><b>СУБД все изменённые блоки данных незамедлительно записываются во [внешнюю память](https://ru.wikipedia.org/wiki/%D0%92%D0%BD%D0%B5%D1%88%D0%BD%D1%8F%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C) при поступлении сигнала</b></p>
3. <p><b>СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления останов</b></p>
4. <p><b>СУБД изменения аккумулируются в [буферах](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D1%84%D0%B5%D1%80_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) внешней памяти до наступления к[онтрольной точки](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%82%D0%BE%D1%87%D0%BA%D0%B0)</b></p>

Ответ: 1 3 4  
Комментарий: SD004016

<p><b> Наибольший недостаток использования бинарного поиска </b></p>

1. <p><b> Потенциально высокие затраты памяти </b></p>
2. <p><b> Для поиска массив должен быть упорядочен </b></p>
3. <p><b> Длительное время работы </b></p>
4. <p><b> Нет правильного ответа </b></p>  
    Ответ: 2  
    Комментарий: SD004017

<p><b> Как ещё называют таблицу символов </b></p>

1. <p><b> Карта </b></p>
2. <p><b> Таблица неясности </b></p>
3. <p><b> Cхема символов</b></p>
4. <p><b> Словарь </b></p>

Ответ: 4

Комментарий: SD004018

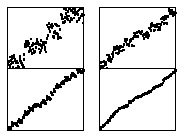
<p><b> База данных это</b></p>

1. <p><b> произвольный тип информации </b></p>
2. <p><b> совокупность приложений, хранимых в соответствии со [схемой данных](https://ru.wikipedia.org/wiki/%D0%A1%D1%85%D0%B5%D0%BC%D0%B0_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85).</b></p>
3. <p><b> совокупность [данных](https://ru.wikipedia.org/wiki/%D0%94%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5), манипулирование которых выполняют с правилами средств [моделирования данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) </b></p>
4. <p><b> это [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), где используется представление [базы данных](https://ru.wikipedia.org/wiki/%D0%91%D0%B0%D0%B7%D0%B0_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в виде [древовидной](https://ru.wikipedia.org/wiki/%D0%94%D1%80%D0%B5%D0%B2%D0%BE%D0%B2%D0%B8%D0%B4%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) структуры. </b></p>

Ответ: 3

Комментарий: SD004019

<p><b>Что изображено на рисунке?</b></p>

<p><b></b></p>

1. <p><b>бинарная сортировка</b></p>
2. <p><b> дерево разбиения </b></p>
3. <p><b> быстрая сортировка </b></p>
4. <p><b> MSD-сортировки </b></p>

Ответ: 4

Комментарий: SD004020

<p><b> Как удалить запись из БД </b></p>

1. <p><b> Where </b></p>
2. <p><b> DELETE </b></p>
3. <p><b> оба варианта верны </b></p>
4. <p><b> оба неверны </b></p>

Ответ: 2

Комментарий: SD004021

<p><b> Как редактировать запись из БД </b></p>

1. <p><b> Where </b></p>
2. <p><b> REDACT </b></p>
3. <p><b> оба варианта верны </b></p>
4. <p><b> UPDATE </b></p>

Ответ: 4

Комментарий: SD004022

<p><b> плюсы бинарного поиска </b></p>

1. <p><b> Быстрая работа алгоритма </b></p>
2. <p><b> Прост в реализации </b></p>
3. <p><b> Все ответы верны </b></p>
4. <p><b> практичность</b></p>

Ответ: 3  
Комментарий: SD004023

<p><b>Что представляет собой объект?</b></p>

1. <p><b>это понятие, сочетающее в себе совокупность данных и действий над ними.</b></p>
2. <p><b>это экземпляр класса</b></p>
3. <p><b> это понятие, сочетающее в себе только совокупность данных </b></p>
4. <p><b> переменная, тип которой задается классом </b></p>

Ответ: 1, 2, 4  
Комментарий: SD004024

<p><b> Записи БД…. </b></p>

1. <p><b>можно просматривать в виде таблицы.</b></p>
2. <p><b>можно просматривать и редактировать в виде таблицы или в виде формы</b></p>
3. <p><b> это понятие, сочетающее в себе только совокупность данных </b></p>
4. <p><b> переменная, тип которой задается классом </b></p>

Ответ: 2  
Комментарий: SD004025

<p><b>Cоздать форму можно с использованием </b></p>

1. <p><b>Конструктора.</b></p>
2. <p><b>Оба варианта верны </b></p>
3. <p><b> мастера формы </b></p>
4. <p><b> Оба варианты неверны</b></p>

Ответ: 1, 2, 4  
Комментарий: SD004026

<p><b>  Какая наименьшая единица хранения данных е БД?</b></p>

1. <p><b>хранимое поле</b></p>
2. <p><b>хранимый файл </b></p>
3. <p><b> хранимая запись</b></p>
4. <p><b>хранимый байт </b></p>

Ответ: 1  
Комментарий: SD004027

<p><b> Какие средства используются для синхронизации?</b></p>

1. <p><b>блокировки.</b></p>
2. <p><b>транзакции</b></p>
3. <p><b> Пароли </b></p>
4. <p><b>восстановление данных после сбоев</b></p>

Ответ: 1, 2, 4  
Комментарий: SD004028

<p><b> Основные требования, предъявляемые к базе данных? </b></p>

1. <p><b> адаптивность и расширяемость </b></p>
2. <p><b> восстановление данных после сбоев </b></p>
3. <p><b> все варианты верны </b></p>
4. <p><b> описание полномочий </b></p>

Ответ: 3  
Комментарий: SD004029

<p><b> Какие задачи можно выполнять с помощью базы данных?</b></p>

1. <p><b>выбор шаблона</b></p>
2. <p><b>добавление таблицы</b></p>
3. <p><b> все варианты верны </b></p>
4. <p><b>копирование и вставка данных </b></p>

Ответ: 3  
Комментарий: SD004030

<p><b>Задачи редактирования файлов в БД?</b></p>

1. <p><b>Исключение из файла фрагмента текста, получение информации о количестве символов текста.</b></p>
2. <p><b> Вставка в файл фрагмента текста, исключение из файла фрагмента текста, упорядочивание элементов файла по ключу.</b></p>
3. <p><b>Упорядочивание элементов файла по определенному ключу.</b></p>

Ответ: 2

Комментарий: SD005063

<p><b>Операцию замены можно выполнить с помощью</b></p>

1. <p><b>исключение из файла фрагмента текста и вставки в файл фрагмента текста.</b></p>
2. <p><b>упорядочивание элементов файла по определенному ключу и вставки в файл фрагмента текста.</b></p>
3. <p><b>исключение из файла фрагмента текста и замена фрагмента текста файла на другой фрагмент.</b></p>

Ответ: 1

Комментарий: SD005064

<p><b>Во время добавления данных в БД происходит аварийное закрытия файла. Но данные были сохраненный и не испорчены. По какой причине данные сохранились?</b></p>

1. <p><b>Утверждение не является верным. Данные были испорчены.</b></p>
2. <p><b>Пользователь успел применить сочетание клавиш ctrl+s.</b></p>
3. <p><b> в конце файла был записан *управляющий символ* конца файла.</b></p>

Ответ: 3

Комментарий: SD005065

<p><b>Работа с файлом осуществляется через (работает осуществляется при помощи консоли) </b></p>

1. <p><b>таблицы</b></p>
2. <p><b>периферийное устройство</b></p>
3. <p><b>буфер</b></p>

Ответ: 3

Комментарий: SD005066

<p><b>Что произойдет после записи символа в требуемом месте?</b></p>

1. <p><b>Файла больше считываться не будет.</b></p>
2. <p><b>Файл считается до конца.</b></p>
3. <p><b>Файл закроется (символ не добавиться).</b></p>

Ответ: 1

Комментарий: SD005067

<p><b>Таким образом, для выполнения операций исключения, вставки и замены фрагментов текста в файлах БД можно использовать?</b></p>

1. <p><b>Два варианта верные.</b></p>
2. <p><b>Использовать вспомогательный файл.</b></p>
3. <p><b>Пользоваться массивами указателей.</b></p>

Ответ: 1

Комментарий: SD005068

<p><b>Если *файл* целиком нельзя разместить в памяти программы то</b></p>

1. <p><b>файл приходится считывать *по* частям и осуществлять перенос информации из одной части в другую.</b></p>
2. <p><b>файл приходится считывать полностью и делить его на части</b></p>
3. <p><b>файл самостоятельно уменьшиться в размере и считается</b></p>

Ответ: 1

Комментарий: SD005069

<p><b>Чтобы вставить строку в середину файла, используя вспомогательный файл, необходимо проделать следующий алгоритм. Является ли алгоритм верным?

1. Открыть входной файл в режиме чтения и вспомогательный файл<br> в режиме записи (этот файл будет создан автоматически). <br>

2.Прочесть из входного файла текст до места вставки. <br>

3. Записать прочитанный фрагмент во вспомогательный файл. <br>

4 Записать во вспомогательный файл вставляемый фрагмент. <br>

5. Прочесть из входного файла остаточный фрагмент и записать его<br> во вспомогательный файл. <br>

6. Закрыть оба файла<br></b></p>

1. <p><b>нет</b></p>
2. <p><b>частично</b></p>
3. <p><b>да</b></p>

Ответ: 3

Комментарий: SD005070

<p><b>Редактирование файла на программном уровне это</b></p>

1. <p><b>изменения, производимые с элементами файла после завершения программы </b></p>
2. <p><b> изменения, производимые с элементами файла в процессе работы</b></p>
3. <p><b>изменения, производимые с файлом в процессе работы</b></p>

Ответ: 2

Комментарий: SD005071

<p><b>Как поможет Метод seekg() в редактировании файл?</b></p>

1. <p><b>Произведёт установку текущей позиции в нужную.</b></p>
2. <p><b>Проверяет не достигнут ли конец файла.</b></p>
3. <p><b>Закрывает файл.</b></p>

Ответ: 1

Комментарий: SD005072

<p><b>Что значит строчка file.seekg(30,ios\_base::beg);</b></p>

1. <p><b>Стать на 31-й байт</b></p>
2. <p><b>Стать на 31 байт с конца</b></p>
3. <p><b>Перепрыгнуть через 3 байта</b></p>

Ответ: 1

Комментарий: SD005073

<p><b>Выберите простой способ форматирования данных</b></p>

1. <p><b>cin,cout</b></p>
2. <p><b>printf scanf</b></p>
3. <p><b>оба варианта верны</b></p>

Ответ: 3

Комментарий: SD005074

<p><b>Считаются ли флаги форматированием частью редактирования файла?</b></p>

1. <p><b>да</b></p>
2. <p><b>нет</b></p>
3. <p><b>зависит от использования</b></p>

Ответ: 3

Комментарий: SD005075

<p><b>Флаги форматирования класса ios dec выполняет функцию</b></p>

1. <p><b>перевода значения в десятичную форму</b></p>
2. <p><b>перевода значения в восьмеричную форму</b></p>
3. <p><b>перевода значения в шестнадцатеричную форму</b></p>

Ответ: 1

Комментарий: SD005076

<p><b>Флаги форматирования класса ios right выполняет функцию</b></p>

1. <p><b>выводит первое слово</b></p>
2. <p><b>выравнивание по правому краю</b></p>
3. <p><b>выводит последние слово</b></p>

Ответ: 2

Комментарий: SD005077

<p><b>Флаги форматирования класса ios fixed выполняет функцию</b></p>

1. <p><b>Экспоненциальный вывод чисел с плавающей запятой</b></p>
2. <p><b>Фиксированный вывод чисел с плавающей запятой</b></p>
3. <p><b>нету верного утверждения</b></p>

Ответ: 2

Комментарий: SD005078

<p><b>Флаги форматирования класса ios skipws выполняет функцию</b></p>

1. <p><b>перевод в восьмеричную форму</b></p>
2. <p><b>показывать десятичную точку при выводе</b></p>
3. <p><b> пропуск пробелов при вводе</b></p>

Ответ: 3

Комментарий: SD005079

<p><b>Инструкции редактирования, которые вставляются прямо в поток.</b></p>

1. <p><b>Манипуляторы</b></p>
2. <p><b>Структуры</b></p>
3. <p><b>Классы</b></p>

Ответ: 1

Комментарий: SD005080

<p><b>Манипуляторы бывают двух видов:(укажите не верный вариант)</b></p>

1. <p><b>константным</b></p>
2. <p><b>с аргументом</b></p>
3. <p><b>без аргумента</b></p>

Ответ: 1

Комментарий: SD005081

<p><b>Манипуляторы ios setw() используется для</b></p>

1. <p><b>Устанавливает ширину поля для вывода данных</b></p>
2. <p><b>Устанавливает указанные флаги форматирования</b></p>
3. <p><b>Устанавливает указанные флаги форматирования</b></p>

Ответ: 1

Комментарий: SD005082

<p><b>Какая функция ostream устанавливает позицию в байтах файлового указателя относительно указанного места в файле?</b></p>

1. <p><b>seekp(position, seek\_dir)</b></p>
2. <p><b>tellp()</b></p>
3. <p><b>write()</b></p>

Ответ: 1

Комментарий: SD005083

<p><b>Какая функция ostream устанавливает SIZE символов из массива str в файл</b></p>

1. <p><b> seekp(position, seek\_dir)</b></p>
2. <p><b>write()</b></p>
3. <p><b>flush()</b></p>

Ответ: 2

Комментарий: SD005084

<p><b>Для чего создается массив с именем buff</b></p>

1. <p><b>для хранение измененных данных</b></p>
2. <p><b>для хранения всех данных</b></p>
3. <p><b>для передачи данных в другом порядке</b></p>

Ответ: 1

Комментарий: SD005085

<p><b>Для манипулирования данными, хранящимися в БД, используется группа операторов</b></p>

1. <p><b>sql</b></p>
2. <p><b>oracle</b></p>
3. <p><b>два ответа являются верными</b></p>

Ответ: 3

Комментарий: SD005086

<p><b>Файл test1.txt заполнены цифрами, все четные цифры записываться в test2.txt, а оставшиеся в test3.txt. Запишите строчку с ошибкой.<br>

void main(){<br>

ifstream ifs;<br>

ofstream ofs1; <br>

ofstream ofs2; <br>

string first\_file\_name = "C:\\programming\\test1.txt";<br>

string second\_file\_name = "C:\\programming\\test2.txt";<br>

string third\_file\_name = "C:\\programming\\test3.txt";<br>

ifs.open(first\_file\_name.c\_str());<br>

ofs1.open(second\_file\_name.c\_str());<br>

ofs2.open(third\_file\_name.c\_str());<br>

int buf; <br>

while (ifs >> buf) {<br>

if (buf % 2 != 0) <br>

ofs1 << buf; <br>

else<br>

ofs2 << buf; }<br>

ifs.close();<br>

ofs1.close();<br>

ofs2.close();}<br></b></p>

1. p><b>while (ifs >> buf)</b></p>
2. <p><b>if (buf % 2 != 0)</b></p>
3. <p><b>ifs.open(first\_file\_name.c\_str())</b></p>

Ответ: 2

Комментарий: SD005087

<p><b>Создан файл, содержащий десять цифр. цифры, отсортированные по возрастанию записываться в другой файл. Укажите строчку в которой есть ошибка. <br>

int main(){<br>

std::ifstream fin("number.txt");<br>

int m[10]; <br>

for (int i = 0; i < 10; i++) {<br>

fin >> m[i]; }<br>

bool flag = false; // сортировка пузырьком<br>

while (!flag) {<br>

flag = true; <br>

for (int i = 0; i < 10; i++) {<br>

if (m[i] > m[i + 1]) {<br>

int tmp = m[i]; <br>

m[i] = m[i + 1]; <br>

m[i + 1] = tmp; <br>

flag = false;}}} <br>

std::ofstream fout("result.txt"); <br>

for (int i = 0; i < 10; i++) {<br>

fout << m[i] << " ";}<br>

return 0; }<br></b></p>

1. <p><b>flag = true; </b></p>
2. <p><b>if (m[i] > m[i + 1]) </b></p>
3. <p><b>bool flag = false; </b></p>

Ответ: 3

Комментарий: SD005088

<p><b>Пример чтения из одного файла и записи в другой с переводом символов в верхний регистр. Укажите строчку, в которой есть ошибка. <br>

int main() {<br>

ifstream fin; <br>

ofstream fout; <br>

string fin\_name("input.txt"), fout\_name("output.txt"), content("");<br>

fin.open(fin\_name); <br>

fout.open(fout\_name); <br>

if (fin.is\_open() || fout.is\_open()) {<br>

getline(fin, content, '\0'); <br>

fin.close();<br>

transform(content.begin(), content.end(), content.begin(), ::toupper); <br>

fout << content; <br>

fout.close();}<br>

return 0;}<br></b></p>

1. <p><b>string fin\_name("input.txt"), fout\_name("output.txt"), …</b></p>
2. <p><b> if (fin.is\_open() || fout.is\_open())</b></p>
3. <p><b>fout.close();</b></p>

Ответ: 2

Комментарий: SD005089

<p><b>Дан файл f, компоненты которого являются целыми числами. Файл f содержит столько же отрицательных чисел, сколько и положительных. Используя вспомогательный файл h, переписать компоненты файла f в файл g так, чтобы в файле g:сначала шли положительные, потом отрицательные числа. Укажите строчку, в которой есть ошибка. <br>

int main(){<br>

int a = 0; <br>

ifstream fin; <br>

ofstream fout; <br>

fout.open("g.txt");<br>

fin.open("f.txt"); <br>

if (!(fin.is\_open())){<br>

Sleep(5000); <br>

return -1;} <br>

const int kol = 10; <br>

int A[kol]; <br>

for (int i(0); i < 0; i++)<br>

{<br>

a = 0; <br>

fin >> a; <br>

A[i] = a; <br>

cout << A[i] << " ";}<br>

int i = 0; <br>

for (int i(0); i < kol; i++){<br>

while (A[i] > 0) {<br>

cout << A[i] << " ";<br>

i++;<br>

if (i > 2) <br>

break;} <br>

while (A[i] < 0) {<br>

cout << A[i] << " ";<br>

i++;<br>

if (i > 2) break;}} <br></b></p>

1. <p><b>fout.open("g.txt”)</b></p>
2. <p><b>const int kol = 10;</b></p>
3. <p><b>for (int i(0); i < 0; i++)</b></p>

Ответ: 3

Комментарий: SD005090

<p><b> База данных - это:</b></p>

1.<p><b> специальным образом организованная и хранящаяся на внешнем носителе совокупность взаимосвязанных данных о некотором объекте</b></p>

2.<p><b> произвольный набор информации</b></p>

3.<p><b> совокупность программ для хранения и обработки больших массивов информации </b></p>

4.<p><b> интерфейс, поддерживающий наполнение и манипулирование данными </b></p>

Ответ: 1

Комментарий: SD005001

<p><b>Какой функционал предоставляет хедер fstream? </b></p>

1.<p><b>Фильтрация файлов. </b></p>

2.<p><b>Считывание данных из файлов и запись в файл. </b></p>

3.<p><b>Редактирование файлов. </b></p>

4.<p><b>Поиск файлов. </b></p>

Ответ:2

Комментарий: SD005002

<p><b>Какая функция считывает символы из stdin и помещает их в массив символов? </b></p>

1.<p><b>tellg()</b></p>

2.<p><b>is\_open()</b></p>

3.<p><b>seekg()</b></p>

4.<p><b>gets()</b></p>

Ответ:4

Комментарий: SD005003

<p><b>Какой класс предоставляет возможности для чтения ? </b></p>

1.<p><b>ifstream</b></p>

2.<p><b>olstream</b></p>

3.<p><b>sestream</b></p>

4.<p><b>usstream</b></p>

Ответ:1

Комментарий: SD005004

<p><b>Какой класс предоставляет возможности для записи? </b></p>

1.<p><b>ifstream</b></p>

2.<p><b>olstream</b></p>

3.<p><b>ofstream</b></p>

4.<p><b>usstream</b></p>

Ответ:3

Комментарий: SD005005

<p><b>Какими способами можно открыть БД? </b></p>

1.<p><b>open()</b></p>

2.<p><b>is\_open()</b></p>

3.<p><b>Указать путь к нему в конструкторе. </b></p>

4.<p><b>Варианты 1,3. </b></p>

Ответ:4

Комментарий: SD005006

<p><b>Если файл не открылся для проверки…</b></p>

1.<p><b>Использовать is\_open()</b></p>

2.<p><b>Проверить переменную файла в логическом выражении</b></p>

3.<p><b>Использовать open()</b></p>

4.<p><b>Вариант 1,2. </b></p>

Ответ.4

Комментарий: SD005007

<p><b>С помощью какого оператора можно организовать считывание</b></p>

1.<p><b>\*</b></p>

2.<p><b> +- </b></p>

3.<p><b> >> </b></p>

4.<p><b>\*</b></p>

Ответ:3

Комментарий: SD005008

<p><b>Какая функция возвращает текущую позицию для чтения? </b></p>

1.<p><b>tellg()</b></p>

2.<p><b>seekg()</b></p>

3.<p><b>getline()</b></p>

4.<p><b>eof()</b></p>

Ответ:1

Комментарий: SD005009

<p><b>Какая функция проверяет не достигнут ли конец БД? Т.е. можно ли из него продолжить чтение. </b></p>

1.<p><b>seekg()</b></p>

2.<p><b>eof()</b></p>

3.<p><b>getline()</b></p>

4.<p><b>tellg()</b></p>

Ответ:2

Комментарий: SD005010

<p><b>Какая функция считывает не одно слово, а целую строку до встречи с первым переводом на новую строку? </b></p>

1.<p><b>getline()</b></p>

2.<p><b>cin()</b></p>

3.<p><b>tolower()</b></p>

4.<p><b>tellg()</b></p>

Ответ:1

Комментарий: SD005011

<p><b>Какая функция используется для закрытия файла? </b></p>

1.<p><b>open()</b></p>

2.<p><b>tellg()</b></p>

3.<p><b>close()</b></p>

4.<p><b>eof()</b></p>

Ответ:3

Комментарий: SD005012

<p><b>Класс ,для установки режима открытия файлов для чтения. </b></p>

1.<p><b>**ios\_base::in**</b></p>

2.<p><b>**ios\_base::out**</b></p>

3.<p><b>**ios\_base::app**</b></p>

4.<p><b>**ios\_base::trunc**</b></p>

Ответ:1

Комментарий: SD005013

<p><b>Класс ,для установки режима открытия файлов для записи. </b></p>

1.<p><b>**ios\_base::in**</b></p>

2.<p><b>**ios\_base::out**</b></p>

3.<p><b>**ios\_base::app**</b></p>

4.<p><b>**ios\_base::trunc**</b></p>

Ответ:2

Комментарий: SD005014

<p><b>Класс ,для установки режима открытия файлов для записи в конце файла. </b></p>

1.<p><b>**ios\_base::in**</b></p>

2.<p><b>**ios\_base::out**</b></p>

3.<p><b>**ios\_base::app**</b></p>

4.<p><b>**ios\_base::trunc**</b></p>

Ответ:3

Комментарий: SD005016

<p><b> Какой из вариантов не является функцией СУБД?</b></p>

1.<p><b> реализация языков определения и манипулирования данными</b></p>

2. <p><b>. обеспечение пользователя языковыми средствами манипулирования данными </b></p>

3. <p><b> защита и целостность данных </b></p>

4. <p><b> координация проектирования, реализации и ведения БД </b></p>

Ответ: 4

Комментарий: SD005017

</b></p>Какой метод производит установку текущей позиции в нужную, указываемую числом? </b></p>

</b></p>**eof()**</b></p>

</b></p>**get()**</b></p>

</b></p>**tellp()**</b></p>

</b></p>**seekg()**</b></p>

Ответ:4

Комментарий: SD005018

<p><b> Как называется набор хранимых записей одного типа?</b></p>

1. <p><b> хранимый файл </b></p>

2. <p><b> представление базы данных </b></p>

3. <p><b> логическая таблица базы данных </b></p>

4. <p><b> физическая таблица базы данных </b></p>

Ответ: 1

Комментарий: SD005019

</b></p>Как отсчитать новую позицию с конца файла с помощью метода seekg()?</b></p>

1.</b></p>ios\_base::end</b></p>

2.</b></p>ios\_base::beg</b></p>

3.</b></p>ios\_base::start</b></p>

4.</b></p>ios\_base::cur</b></p>

Ответ:1

Комментарий: SD005020

</b></p>Функция считывает count байт из файла…</b></p>

1.</b></p>read()</b></p>

2.</b></p>write()</b></p>

3.</b></p>gets()</b></p>

4.</b></p>tellg()</b></p>

Ответ:1

Комментарий: SD005021

<p><b> К чему приведет отсутствие логической и физической независимости данных?</b></p>

1. <p><b> к необходимости изменения прикладных программ при изменении физического представления базы данных</b></p>

2. <p><b> к большей достоверности данных</b></p>

3. <p><b> к возможному изменению физического представления данных при изменении прикладных программ </b></p>

4. <p><b> Вариант 1 3. </b></p>

Ответ:4

Комментарий: SD005022

</b></p>Какая функция переписывает count байт из буфера? </b></p>

1.</b></p>write()</b></p>

2.</b></p>read()</b></p>

3.</b></p>**tellp()**</b></p>

4.</b></p>tellg()</b></p>

Ответ:1

Комментарий: SD005023

</b></p>Класс ,для установки режима открытия файлов в бинарном. </b></p>

1.</b></p>**ios\_base::in**</b></p>

2.</b></p>**ios\_base::out**</b></p>

3.</b></p>**ios\_base::binary**</b></p>

4.</b></p>**ios\_base::trunc**</b></p>

**Ответ:3**

Комментарий: SD005024

<p><b> Какая наименьшая единица хранения данных в БД?</b></p>

1. <p><b> хранимое поле </b></p>

2. <p><b> хранимый файл</b></p>

3. <p><b> хранимая запись </b></p>

4. <p><b> хранимый байт </b></p>

Ответ: 1

Комментарий: SD005025

<p><b> Верно ли, что для считывания данных из БД, необходимо: описать переменную типа **ofstream**. </b></p>

1.<p><b> Да</b></p>

2.<p><b> Нет</b></p>

Ответ:2

Комментарий: SD005026

<p><b> Верно ли, что для записи данных из БД, необходимо:описать переменную типа **ofstream**. </b></p>

1.<p><b> Да</b></p>

2.<p><b> Нет</b></p>

Ответ:1

Комментарий: SD005027

<p><b> Причинами низкой эффективности проектируемых БД могут быть:</b></p>

1. <p><b> количество подготовленных документов </b></p>

2. <p><b> большая длительность процесса структурирования </b></p>

3. <p><b> недостаточно глубокий анализ требований </b></p>

4. <p><b> Вариант 2 3</b></p>

Ответ:4

Комментарий: SD005028

<p><b> Обязательна ли функция close()?</b></p>

1.<p><b> Да </b></p>

2.<p><b> Нет </b></p>

3.<p><b> Такой функции не существует </b></p>

Ответ:2

Комментарий: SD005029

<p><b>На какой хедер похож **fstream**</b></p>

1. <p><b>**iostream**</b></p>

2. <p><b>string</b></p>

3. <p><b>iterator</b></p>

Ответ:1

Комментарий: SD005030

**<p><b>** [**#include**](https://vk.com/im?sel=523705059&st=%23include)**<stdio.h> <br>   
Int remove(const char**[**\*filename**](https://vk.com/id20019287)**) <br>   
Что делает данная программа? </b></p>**

1. <p><b>Изменяет файл</b></p>
2. <p><b>Удаляет файл</b></p>
3. <p><b>Ничего из выше перечисленного</b></p>­­­

Ответ: 2

Комментарий: SD005091

**<p><b> Что делает функция remove()? </b></p>**

1. <p><b>Перемещает файл </b></p>
2. <p><b>Добавляет файл </b></p>
3. <p><b>Удаляет файл </b></p>

Ответ: 3

Комментарий: SD005092

**<p><b> Какое значение возвращает функция remove() при успешном удалении? </b></p>**

1. <p><b>0 </b></p>
2. <p><b>1 </b></p>
3. <p><b>ничего не возвращает</b></p>

Ответ: 1

Комментарий: SD005093

**<p><b>В чем отличие DELETE от NULL? </b></p>**

1. <p><b>DELETE удаляет данные,а NULL нет </b></p>
2. <p><b>DELETE удаляет данные, а NULL удаляет сам файл </b></p>
3. <p><b>Они ни чем не отличаются </b></p>

Ответ: 2

Комментарий: SD005094

**<p><b> Сколько основных способов удаления из БД существует? </b></p>**

1. <p><b>6 </b></p>
2. <p><b>5 </b></p>
3. <p><b>2 </b></p>

Ответ: 3

Комментарий: SD005095

**<p><b> Что делает параметр where? </b></p>**

1. <p><b>Удаляет все данные </b></p>
2. <p><b>Удаляет лишь те элементы, которые подходят под условие. </b></p>
3. <p><b>Находит элементы, которые подходят под условие </b></p>

Ответ: 2

Комментарий: SD005096

**<p><b> Что означает параметр fname? <br>**

**int remove( const char \* fname )**

**</b></p>**

1. <p><b>Имя файла для удаления </b></p>
2. <p><b>Имя файла для изменения </b></p>
3. <p><b>Имя файла для добавления </b></p>

Ответ: 1

Комментарий: SD005097

**<p><b> Какое значение функция remove() возвращает в результате ошибки? </b></p>**

1. <p><b>-1 </b></p>
2. <p><b>NULL </b></p>
3. <p><b>ERROR </b></p>

Ответ: 1

Комментарий: SD005098

**<p><b> Какой заголовочный файл нужно подключить для функции remove() в языке C? </b></p>**

1. <p><b>stdio.h </b></p>
2. <p><b>windos.h</b></p>
3. <p><b>iostream </b></p>

Ответ: 1

Комментарий: SD005099

**<p><b> Функция remove() происходит из языка: </b></p>**

1. <p><b>С++ </b></p>
2. <p><b>С </b></p>
3. <p><b>Pascal </b></p>

Ответ: 2

Комментарий: SD005100

**<p><b> Какой заголовочный файл нужно подключить для функции remove() в языке C++? </b></p>**

1. <p><b>stdio.h </b></p>
2. <p><b>cstdio </b></p>
3. <p><b>iomanip </b></p>

Ответ: 2

Комментарий: SD005101

**<p><b> Что означает значение ENOENT ошибки errno? </b></p>**

1. <p><b>Файл не существует </b></p> (верный ответ )
2. <p><b>Доступ запрещен </b></p>
3. <p><b>Ничего из перечисленного </b></p>

Ответ: 1

Комментарий: SD005102

**<p><b> Что означает значение EACESS ошибки errno? </b></p>**

1. <p><b>Доступ запрещен</b></p>
2. <p><b>Файл не существует</b></p>
3. <p><b>Файл пуст</b></p>

Ответ: 1

Комментарий: SD005103

**<p><b> Какие достаточные условия для удаления?</b></p>**

1. <p><b>Файл существовал до запуска программы.</b></p>
2. <p><b>Файл существовал до запуска программы, к нему есть доступ.</b></p>
3. <p><b>В любых случаях удаление возможно.</b></p>

Ответ: 2

Комментарий: SD005104

**<p><b> Что выполняет функция remove\_if()? </b></p>**

1. <p><b>Удаляет все элементы, для которых выполняется указанное условие.</b></p>
2. <p><b>Удаляет все элементы,если хотя бы для одного элемента выполняется указанное условие</b></p>
3. <p><b>Удаляет все элементы, если элементы подходят под достаточные условия удаления.</b></p>

Ответ: 1

Комментарий: SD005105

**<p><b> #include <stdio.h><br>  
int main(void) <br>  
<br>{<br>  
char fname[80]; <br>  
printf ("Name of file to remove: ");<br>  
gets(fname); <br>   
if(remove(fname))**

**<br>{<br>   
printf("Error removing file");<br>  
return 1; <br>  
<br>}<br>  
else return 0; <br>  
}<b></p>**

1. <p><b>программа использует remove() для удаления файла, определяемого пользователем</b></p>
2. <p><b>программа использует remove() для сохранения файла, определяемого пользователем</b></p>

Ответ: 1

Комментарий: SD0051016

**<p><b> Оператор Delete освобождает память из кучи?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 1

Комментарий: SD005107

**<p><b> Команда**DELETE**удаляет данные из БД ?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 1

Комментарий: SD005108

**<p><b>При удалении записи генерируются события BeforeDelete и AfterDelete типа?</b></p>**

1. <p><b> TDataSetNotifyEvent </b></p>
2. <p><b>AFTER</b></p>

Ответ: 1

Комментарий: SD005109

**<p><b>Delete это?</b></p>**

1. <p><b>команда DELETE,обеспечивающая удаление информации из базы данных </b></p>
2. <p><b> команда DELETE,обеспечивающая обновление информации из базы данных </b></p>

Ответ: 1

Комментарий: SD005110

**<p><b>**

**Что делает программа ? <br>**

**void \_\_fastcall TForm1::Button1Click(TObjееt \*Sender) <br>**

**{<br>**

**int n; <br>**

**Table1->RecNo; <br>**

**while (n >= 1) <br>**

**{<br>**

**Table1->Delete();<br>**

**Table1->RecNo = Table1->RecNo - 1; <br>**

**n--;<br>**

**}<br>**

**}<br>**

**</b></p>**

1. <p><b> удаления записей набора данных с текущей по первую</b></p>
2. <p><b> сохранения записей набора данных с текущей по первую</b></p>

Ответ: 1

Комментарий: SD005111

**<p><b> верно ли написан код if(remove\_\_remo(file) == 0) <br>**

**printf("Файл %s удален.\n");?”</b></p>**

1. <p><b>нет </b></p>
2. <p><b>да </b></p>

Ответ: 1

Комментарий: SD005112

**<p><b> #include <stdio.h><br>**

**String FileName = "c:\\readme.txt";<br>**

**remove(FileName.c\_str());<br>**

**Файлы удаляются безвозвратно ? <br>**

**</b></p>**

1. <p><b>да </b></p>
2. <p><b>нет </b></p>

Ответ: 1

Комментарий: SD005113

**<p><b> "DELETE" переводится с английского как ? </b></p>**

1. <p><b>Удалить </b></p>
2. <p><b>Сохранить</b></p>
3. <p><b>Добавить</b></p>

Ответ: 1

Комментарий: SD005114

**<p><b> что значит выражение “DELETE FROM developers”?</b></p>**

1. <p><b>хотим удалить все данные из таблицы.</b></p>
2. <p><b> хотим добавить все данные из таблицы.</b></p>

Ответ: 1

Комментарий: SD005115

**<p><b> Можно ли удалять информацию в файле без дополнительных библиотек? </b></p**>

1. <p><b>да </b></p>
2. <p><b>нет </b></p>

Ответ: 1

Комментарий: SD005116

**<p><b> Через этот цикл будет удаляться информация ? <br>**

**for(i=0;i<=10;i++)<br>**

**{<br>**

**k=strcmp(buff1,p[i].FIO); <br>**

**if(k==0) <br>**

**{<br>**

**memset(&p[i], 0, sizeof(p[i])); <br>**

**}<br>**

**}<br>**

**for(int j=0;j<=10;j++)<br>**

**{if(p[j].FIO!=NULL&&p[j].number!=NULL&&p[j].pasport!=NULL&&p[j].punkt!=NULL&&p[j].ticket!=NULL) <br>**

**{ <br>**

**fprintf(t,"%s:%s\n","ФИО",p[j].FIO); <br>**

**fprintf(t,"%s:%s\n","Паспорт",p[j].pasport); <br>**

**fprintf(t,"%s:%s\n","Пункт",p[j].punkt); <br>**

**fprintf(t,"%s:%d\n","Номер",p[j].number); <br>**

**fprintf(t,"%s:%d\n","Билеты :",p[j].ticket); <br>**

**}<br>**

**else continue; <br>**

**} <br></b></p>**

1. <p><b>да</b></p>
2. <p><b>нет </b></p>

Ответ: 1

Комментарий: SD005117

**<p><b> Как удалить записи из файла в C++?</b></p>**

1. <p><b> Функция remove() удаляет **файлы**. Она имеет следующий прототип: int remove(const char \*имя\_файла); В случае удачного выполнения она возвращает ноль, а в случае неудачного - не ноль.</b></p>
2. <p><b>Функция used </b></p>

Ответ: 1

Комментарий: SD005118

**<p><b> Что делает программа remove(file) == 0 в коде? <br>**

**#include <stdio.h><br>**

**int main(void) <br>**

**{<br>**

**char file[80]; <br>**

**/\* запросить имя удаляемого файла \*/<br>**

**printf("Введите имя удаляемого файла");<br>**

**gets(file); <br>**

**if(remove(file) == 0) <br>**

**printf("Файл %s удален.\n");<br>**

**else<br>**

**perror("remove");<br>**

**return 0; <br>**

**}<br></b></p>**

1. <p><b> удалить файл.</b></p>
2. <p><b> сохранить файл </b></p>

Ответ: 1

Комментарий: SD005119

**<p><b> Что делает программа remove(file) == 0 в коде? <br>**

**int \_\_fastcall TForm1::DelListDirItems(char \*dir, char \*mask) <br>**

**{<br>**

**WIN32\_FIND\_DATA DIR={0};<br>**

**HANDLE hFile=0; <br>**

**BOOL done=TRUE; <br>**

**char dirname[128]={0};<br>**

**TCHAR LevelUp='.'; <br>**

**int ItemCount=0; <br>**

**char fullpath[256]={0};<br>**

**AnsiString TempString; <br>**

**char DelStr[256]={0};");<br>**

**}<br></b></p>**

1. <p><b> функция удаления файлов по маске mask из директории dir.</b></p>
2. <p><b> сохранить файл </b></p>

Ответ: 1

Комментарий: SD005120

**<p><b> Небольшая часть элемента, которая используется для управления сортировкой.</b></p**

1. <p> <b>Якорь </b> </p>
2. <p> <b> Ключь </b> </p>
3. <p> <b> id </b> </p>

Ответ: 2

Комментарий: SD016121

**<p><b> Конкретные характеристики ключей и элементов в разных приложениях могут существенно отличаться друг от друга, однако абстрактное понятие размещения ключей и связанной с ними информации в определенном порядке и представляет собой суть задачи …. </b></p>**

1. <p><b> Взаимодействия </b></p>
2. <p><b>Компоновки </b></p>
3. <p><b> Компенсации </b></p>
4. <p><b> Сортировки </b></p>

Ответ: 4

Комментарий: SD005122

**<p><b>Метод сортировки, при котором сортируемые файлы полностью помещаются в оперативной памяти, называется </b></p>**

1. <p><b> Прямым </b></p>
2. <p><b> Внутренним </b></p>
3. <p><b> Полноценным</b></p>
4. <p><b> Конкретным </b></p>

Ответ: 2

Комментарий: SD005123

**<p><b> Сортировка файлов, хранящихся на магнитной ленте или диске, называется </b></p>**

1. <p><b> Обратным </b></p>
2. <p><b> Ограниченным </b></p>
3. <p><b> Вложенным </b></p>
4. <p><b>Внешней</b></p>

Ответ: 4

Комментарий: SD005124

**<p><b> Функция (вставить слово) представляет собой шаблонную реализацию, которая обращается к сортируемым элементам только через первый аргумент и нескольких простых операций с данными. </b></p>**

1. <p><b> stl </b></p>
2. <p><b> sort </b></p>
3. <p><b> qsort </b></p>
4. <p><b> caller </b></p>

Ответ: 2

Комментарий: SD005125

**<p> <b> Сортировка, в которой последовательность выполняемых операций не зависит от упорядоченности данных, называется</b> </p>**

1. <p> <b> Линейный </b> </p>
2. <p> <b> Неадаптивной </b> </p>
3. <p> <b> Последовательной</b> </p>
4. <p> <b> Нет правильного ответа </b> </p>

Ответ: 2

Комментарий: SD005126

**<p> <b> Сортировка, в которой последовательность выполняемых операций зависит от упорядоченности данных, называется </b> </p>**

1. <p><b> Адаптивной</b></p>
2. <p><b>Обратной</b></p>
3. <p><b> Непоследовательной </b></p>
4. <p><b> Абстрактной </b></p>

Ответ: 1

Комментарий: SD005127

**<p> <b> Если метод сортировки сохраняет относительный порядок размещения в файле элементов с одинаковыми ключами, значит этот метод … </b> </p>**

1. <p><b> Устойчивый </b></p>
2. <p><b> Сбалансированный </b></p>
3. <p><b > Упорядоченный </b></p>
4. <p><b> Одинарный </b></p>

Ответ: 1

Комментарий: SD0005128

**<p> <b> Если переупорядочиваются не сами элементы, а массив указателей так, что первый указатель указывает на наименьший элемент, следующий — на наименьший из оставшихся и т.д. эта сортировка является </b> </p>**

1. <p><b> Прямой </b></p>
2. <p><b> Обрытный </b></p>
3. <p><b> Косвенной </b></p>
4. <p><b> Нет такой сортировки </b></p>

Ответ: 3

Комментарий: SD005129

**<p> <b> Метод, работующий по принципу приведенным далее называется? Сначала находится наименьший элемент массива и меняется местами с элементом, стоящим первым в сортируемом массиве. Потом находится второй наименьший элемент и меняется местами с элементом, стоящим вторым в исходном массиве. Этот процесс продолжается до тех пор, пока весь массив не будет отсортирован. </b> </p>**

1. <p><b> Сортировкой выбором </b></p>
2. <p><b> Пузырьковая сортировка</b></p>
3. <p><b> Сортировка расчёской</b></p>
4. <p><b> Сортировка перемешиванием</b></p>

Ответ: 1

Комментарий: SD005130

**<p> <b> При каждом проходе алгоритма по внутреннему циклу, очередной наибольший элемент массива ставится на своё место в конце массива рядом с предыдущим «наибольшим элементом», а наименьший элемент перемещается на одну позицию к началу массива. Такой метод сортировки называется </b> </p>**

1. <p><b> Чётно-нечётная сортировка </b></p>
2. <p><b> Шейкерная сортировка </b></p>
3. <p><b> Глупая сортировка </b></p>
4. <p><b> Сортировка пузырьком </b></p>

Ответ: 4

Комментарий: SD005131

**<p> <b> Является ли сортировка пузырьком устойчивой </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Она может быть как устойчивой, так и неустойчивой </b></p>

Ответ: 1

Комментарий: SD005132

**<p> <b> Сортировка выбором выполняет порядка … сравнений и … обменов элементов. (Вставить правильный ответ)</b> </p>**

1. <p><b> N2/ 4 сравнений, N / 2 обменов </b></p>
2. <p><b> N2/ 2 сравнений, N обменов </b></p>
3. <p><b> N2/ 4 сравнений, N обменов </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD0050133

**<p> <b> Сортировка вставками выполняет в среднем порядка …  сравнений и … полуобменов (перемещений). (Вставить правильный ответ)</b> </p>**

1. <p><b> N2/ 4 сравнений, N2/ 4 полуобменов </b></p>
2. <p><b> N2/ 4 сравнений, N2/ 2 полуобменов </b></p>
3. <p><b> N2/ 2 сравнений, N2/ 4 полуобменов </b></p>
4. <p><b> Нет правильного ответа</b></p>

Ответ: 1

Комментарий: SD005134

**<p> <b> Пузырьковая сортировка выполняет порядка …  сравнений и … обменов. (Вставить правильный ответ) </b> </p>**

1. <p><b> N2/ 4 сравнений, N2/ 2 полуобменов </b></p>
2. <p><b> N2/ 4 сравнений, N2/ 4 полуобменов </b></p>
3. <p><b> N2/ 2 сравнений, N2/ 2 обменов</b></p>
4. <p><b> Нет правильного ответа</b></p>

Ответ: 3

Комментарий: SD005135

**<p> <b>  Пара ключей, которые нарушают порядок в файле, называются </b> </p>**

1. <p><b> Тождественное преобразование</b></p>
2. <p><b> Инверсией </b></p>
3. <p><b> Нет правильных ответов </b></p>

Ответ: 2

Комментарий: SD005136

**<p> <b> Если отсортированному файлу добавляются нескольких элементов, либо в сортированном файле изменены ключи нескольких элементов, то для таких файлов наиболее эффективна сортировка является </b> </p>**

1. <p><b> Сортировка выбором </b></p>
2. <p><b> Сортировка вставками </b></p>
3. <p><b> Пузырьковая сортировка</b></p>
4. <p><b>Все эти сортировки одинаково хороши </b></p>

Ответ: 2

Комментарий: SD005137

**<p> <b> Время выполнения сортировки вставками зависит от </b> </p>**

1. <p><b> Общего числа инверсий </b></p>
2. <p><b>Характера распределения инверсий </b></p>
3. <p><b> Как от числа, так и от распределения инверсий </b></p>
4. <p><b> Не зависит от инверсий </b></p>

Ответ: 1

Комментарий: SD005138

**<p> <b> Время выполнения сортировки выбором линейно для файлов с  </b> </p>**

1. <p><b> Большими элементами и малыми ключами </b></p>
2. <p><b> Малыми элементами и большими ключами </b></p>
3. <p><b> Большими элементами и большими ключами </b></p>
4. <p><b> Малыми элементами и малыми ключами </b></p>

Ответ: 1

Комментарий: SD005139

**<p> <b> Когда к уже отсортированному файлу добавляются нескольких элементов, либо в сортированном файле изменены ключи нескольких элементов, то такие файлы называются </b> </p>**

1. <p><b> Нарушенными </b></p>
2. <p><b> Изменчивые </b></p>
3. <p><b> Частично упорядоченные </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005140

**<p> <b> Для файлов небольших размеров сортировка вставками и сортировка выбором работают примерно в два раза быстрее пузырьковой сортировки. Утверждение является правдой? </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Не хватает данных для ответа </b></p>

Ответ: 1

Комментарий: SD005141

**<p> <b> Для подсчета количества инверсий в файле необходимо для каждого элемента просуммировать число элементов слева, которые больше его. Так ли это? </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Нет ответа </b></p>

Ответ: 1

Комментарий: SD005142

**<p> <b>  Сортировка Шелла представляет собой простое расширение метода вставок, быстродействие которого достигается за счет возможности … </b> </p>**

1. <p><b> Сужать диапазон поиска элементов </b></p>
2. <p><b> Обмена далеко отстоящих друг от друга элементов </b></p>
3. <p><b> Обмена соседних элементов </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD0005143

**<p> <b> Устойчива ли сортировка Шелла? </b> </p>**

1. <p><b> Да </b></p>
2. <p><b> Нет </b></p>
3. <p><b> Может быть как устойчивой, так и неустойчивой </b></p>

Ответ: 2

Комментарий: SD005144

**<p> <b> Лучшее время работы сортировки Шелла </b> </p>**

1. <p><b> O(nlogn) </b></p>
2. <p><b> O(n log2 n) </b></p>
3. <p><b> O(n) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005145

**<p> <b> Худшее время работы сортировки Шелла </b> </p>**

1. <p><b> O(n^2) </b></p>
2. <p><b> O(nlogn) </b></p>
3. <p><b> O(nlgn) </b></p>
4. <p><b> O(nloglogn) </b></p>

Ответ: 1

Комментарий: SD005146

**<p> <b> Худшее время работы сортировки выбором </b> </p>**

1. <p><b> О(n^2) </b></p>
2. <p><b> O(nlgn) </b></p>
3. <p><b> O(n log2 n) </b></p>
4. <p><b> O(nloglogn) </b></p>

Ответ: 1

Комментарий: SD005147

**<p> <b> Лучшее время работы сортировки выбором </b> </p>**

1. <p><b> O(nlogn) </b></p>
2. <p><b> O(n) </b></p>
3. <p><b> О(n^2) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005148

**<p> <b> Лучшее время работы сортировки вставками </b> </p>**

1. <p><b> O(nlogn) </b></p>
2. <p><b> O(n) </b></p>
3. <p><b> О(n^2) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005149

**<p> <b> Худшее время работы сортировки вставками </b> </p>**

1. <p><b> O(nlgn) </b></p>
2. <p><b> O(nloglogn) </b></p>
3. <p><b> O(n^2) </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005150

**<p><b> Получение конкретного фрагмента или фрагментов информации из больших объемов ранее сохраненных данных это? </b></p>**

1. <p><b> Анализ </b></p>
2. <p><b> Поиск </b></p>
3. <p><b> Выборка </b></p>
4. <p><b> Извлечение </b></p>

Ответ: 2

Комментарий: SD005181

**<p><b>Структура данных элементов с ключами, которая поддерживает две базовые операции: вставку нового элемента и возврат элемента с заданным ключом, называется?</b></p>**

1. <p><b> Таблица символов</b></p>
2. <p><b> Бинарный код </b></p>
3. <p><b> Массив </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: SD005182

**<p><b> Язык С++ позволяет создавать типы данных, которые ведут себя аналогично базовым типам языка Си. Такие типы обычно называют </b></p>**

1. <p><b>Абстрактный тип данных</b></p>
2. <p><b> Примитивный тип данных </b></p>
3. <p><b> Взаимный тип данных </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: SD005183

**<p><b>  Если значения ключей - положительные целые числа, меньшие M, и элементы имеют различные ключи, то тип данных таблицы символов может быть реализован с помощью индексированных значениями ключей массивов так, что для выполнения операций вставить, найти и удалить потребуется </b></p>**

1. <p><b> Постоянное время </b></p>
2. <p><b> Всремя не зависит от значения ключа </b></p>
3. <p><b> N/2 </b></p>
4. <p><b> Мало данных для ответа </b></p>

Ответ: 1

Комментарий: SD005184

**<p><b>  Последовательный поиск в таблице символов с N элементами требует выполнения порядка … сравнений при успешном поиске (в среднем). </b></p>**

1. <p><b> N </b></p>
2. <p><b>N/4 </b></p>
3. <p><b>2N/3 </b></p>
4. <p><b> N/2 </b></p>

Ответ: 4

Комментарий: SD005185

**<p><b> Последовательный поиск в таблице символов с N неупорядоченными элементами требует … количества шагов для выполнения вставок и … сравнений при неудачном поиске (всегда).</b></p>**

1. <p><b> постоянного кол-во шагов, N/2 сравнений </b></p>
2. <p><b> постоянного кол-во шагов, N сравнений </b></p>
3. <p><b> N кол-во шагов, N/2 сравнений </b></p>
4. <p><b> N/2 кол-во шагов, N сравнений </b></p>

Ответ: 2

Комментарий: SD005186

**<p><b> Последовательный поиск в таблице символов из N упорядоченных элементов требует порядка … операций для вставки, успешного поиска и неудачного поиска (в среднем). </b></p>**

1. <p><b> N </b></p>
2. <p><b> N/2 </b></p>
3. <p><b> N^2/2 </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005187

**<p><b> Алгоритм, который изменяет порядок элементов так, чтобы часто запрашиваемые элементы встречались в начале поиска, называется? </b></p>**

1. <p><b> Аглоритм интерполирующего поиск </b></p>
2. <p><b> Алгоритм самоорганизующегося поиска </b></p>
3. <p><b> Алгоритм значемого поиска </b></p>

Ответ: 2

Комментарий: SD005188

**<p><b> Классический алгоритм поиска элемента в отсортированном массиве, использующий дробление массива на половины называется </b></p>**

1. <p><b> Линейный поиск  </b></p>
2. <p><b> Бинарный поиск </b></p>
3. <p><b> Интерполирующий поиск </b></p>
4. <p><b> Поиск подстроки в строке  </b></p>

Ответ: 2

Комментарий: SD005189

**<p><b> При бинарном поиске выполняется не более чем … сравнений (и при успешном, и при неудачном). </b></p>**

1. <p><b> N </b></p>
2. <p><b> N/2 </b></p>
3. <p><b> N - 1 </b></p>
4. <p><b> [lgN] + 1  </b></p>

Ответ: 4

Комментарий: SD005190

**<p><b> Наибольший недостаток использования бинарного поиска </b></p>**

1. <p><b> Потенциально высокие затраты памяти </b></p>
2. <p><b> Для поиска массив должен быть упорядочен </b></p>
3. <p><b> Длительное время работы </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: SD005191

**<p><b> Достоинства бинарного поиска </b></p>**

1. <p><b>  Быстрая работа алгоритма </b></p>
2. <p><b> Прост в реализации </b></p>
3. <p><b> Практичность </b></p>
4. <p><b> Все ответы верны </b></p>

Ответ: 4

Комментарий: SD005192

**<p><b>  Алгоритм который перебирает все элементы в массиве, сравнивая их с заданным ключом называется </b></p>**

1. <p><b> Линейный алгоритм </b></p>
2. <p><b> Бинарный алгоритм </b></p>
3. <p><b> [Интерполирующий поиск С++](https://purecodecpp.com/archives/2503)</b></p>
4. <p><b> Такого алгоритма нету </b></p>

Ответ: 1

Комментарий: SD005193

**<p><b> Иерархическая структура данных, в которой каждый узел имеет значение (оно же является в данном случае и ключом) и ссылки на левого и правого потомка, называется </b></p>**

1. <p><b> Родовая структура </b></p>
2. <p><b> Иерархическое дерево </b></p>
3. <p><b> Бинарное дерево </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: SD005194

**<p><b> Бинарное дерево поиска (binary search tree - BST) — это бинарное дерево, обладающее дополнительными свойствами, какими? </b></p>**

1. <p><b> значение левого потомка меньше значения родителя, а значение правого потомка больше значения родителя для каждого узла дерева. </b></p>
2. <p><b> значение левого потомка больше значения родителя, а значение правого потомка меньше значения родителя для каждого узла дерева. </b></p>
3. <p><b> значение левого потомка меньше значения родителя и значение правого потомка меньше значения родителя для каждого узла дерева. </b></p>
4. <p><b> значение левого потомка меньше значения родителя, а значение правого потомка равно значению родителя для каждого узла дерева. </b></p>

Ответ: 1

Комментарий: SD005195

**<p><b> BST-деревья используются для реализации операций </b></p>**

1. <p><b> Найти </b></p>
2. <p><b> Вставить </b></p>
3. <p><b> Создать </b></p>
4. <p><b> Найти, вставить, создать и подсчитать </b></p>

Ответ: 4

Комментарий: SD005196

**<p><b> Как ещё называют таблицу символов </b></p>**

1. <p><b> Карта </b></p>
2. <p><b> Таблица неясности </b></p>
3. <p><b> Словарь </b></p>

Ответ: 3

Комментарий: SD005197

**<p><b> В дереве бинарного поиска, образованном N случайными ключами, для успешного поиска в среднем требуется около … сравнений </b></p>**

1. <p><b> 2lgN ≈ 1.39lgN </b></p>
2. <p><b> lgN </b></p>
3. <p><b> 2lgN ≈ 1.50lgN </b></p>
4. <p><b> N </b></p>

Ответ: 1

Комментарий: SD005198

**<p><b> В дереве бинарного поиска, образованном N случайными ключами, для вставок и неудачного поиска в среднем требуется около … сравнений. </b></p>**

1. <p><b> 2lgN ≈ 1.50lgN </b></p>
2. <p><b> lgN </b></p>
3. <p><b> 2lgN ≈ 1.39lgN </b></p>
4. <p><b> N </b></p>

Ответ: 3

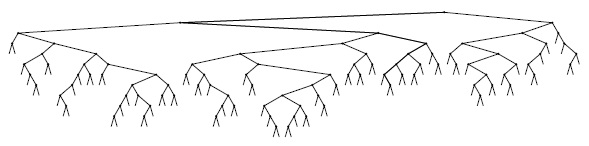
Комментарий: SD005199

**<p><b>  Для поиска в дереве бинарного поиска с N ключами в худшем случае может потребоваться … сравнений. </b></p>**

1. <p><b> lgN </b></p>
2. <p><b> N </b></p>
3. <p><b> 2N </b></p>
4. <p><b> 1/2\*N </b></p>

Ответ: 2

Комментарий: SD005200

**<p><b> Что изображено на рисунке? </b></p>**

1. <p><b> Интерполяционный поиск </b></p>
2. <p><b> Фактографический поиск </b></p>
3. <p><b> Дерево бинарного поиска </b></p>

Ответ: 3

Комментарий: SD005201

**<p><b> Функция, которая вызывает сама себя, называется </b></p>**

1. <p><b> Саморегулируемая функция </b></p>
2. <p><b> Обратная функция </b></p>
3. <p><b> Рекурсивная функция </b></p>
4. <p><b> Функция ротации </b></p>

Ответ: 3

Комментарий: SD005202

**<p><b> Динамическая структура, в которой хранится информация о функциях, выполняющихся в процессоре, назывется </b></p>**

1. <p><b> Стек вызовов </b></p>
2. <p><b> Стек памяти </b></p>
3. <p><b> Стек динамической функции </b></p>
4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: SD005203

**<p><b>  Узел, содержащий ключ S, перемещается в дереве вниз и становится правым дочерним узлом своего прежнего левого дочернего узла. Это?</b></p>**

1. <p><b> Ротация влево в BST-дереве </b></p>
2. <p><b> Ротация вправо в BST-дереве </b></p>
3. <p><b> Без направленная ротация в BST-дереве </b></p>
4. <p><b> Правый дочерний узел </b></p>

Ответ: 2

Комментарий: SD005204

**<p><b> Узел, содержащий ключ A, перемещается в дереве вниз и становится левым дочерним узлом своего прежнего правого дочернего узла. Это? </b></p>**

1. <p><b> Ротация влево в BST- </b></p>
2. <p><b> Ротация вправо в BST-дереве </b></p>
3. 3. <p><b> Левый дочерний узел </b></p>

Ответ: 1

Комментарий: SD005205

**<p><b> Локальное изменение позволяющие перемещать узлы по деревьям без изменения глобальных свойств упорядоченности, которые и делают BST-дерево полезной для поиска структурой, это? </b></p>**

1. <p><b> Аменция </b></p>
2. <p><b> Проекция </b></p>
3. <p><b> Ротация </b></p>
4. <p><b> Абсолюция </b></p>

Ответ: 3

Комментарий: SD005206

**<p><b> Ротации применяются для </b></p>**

1. <p><b> Удаления определенных узлов дерева, для последующего использования. </b></p>
2. <p><b> Перемещения конкретных узлов по дереву и предотвращения разбалансировки деревьев. </b></p>
3. <p><b> Выбора и последующего перемещения для разбалансировки дерева. </b></p>

Ответ: 2

Комментарий: SD005207

**<p><b> Что такое вес бинарного дерева? </b></p>**

1. <p><b> Число листьев </b></p>
2. <p><b> Число узлов </b></p>
3. <p><b> Число узлов, ссылающихся на нулевой элемент </b></p>
4. <p><b> Число дуг дерева </b></p>

Ответ: 1

Комментарий: SD005208

**<p><b> Выберите утверждения, верные для идеально сбалансированного дерева.**

**</b></p>**

1. 1. <p><b> Выберите утверждения, верные для идеально сбалансированного дерева. </b></p>
2. <p><b> Каждый уровень содержит не более узлов, где корень - нулевой узел. </b></p>
3. <p><b> Для каждой его вершины высота ее двух поддеревьев различается не более, чем на 1 </b></p>
4. <p><b> Для поиска любого элемента перебирается не более log2n вершин, где n - число вершин в дереве </b></p>

Ответ: 1, 3, 4

Комментарий: SD005209

**<p><b> Сбалансированным бинарным деревом называется... </b></p>**

1. <p><b> Дерево, для каждой вершины которого высота двух его поддеревьев различается не более чем на единицу</b></p>
2. <p><b> Дерево, элементы которого распределены таким образом, что слева от корня находятся элементы с меньшим индексом, а справа - с большим</b></p>
3. <p><b> Дерево, в котором у каждого поддерева имеется пара потомков </b></p>
4. <p><b> Дерево, индекс корня которого равен среднему арифметическому всех остальных индексов </b></p>

Ответ: 1

Комментарий: SD005210

<p><b>БД, как правило, одновременно является и интегрированной, и общедоступной. Под "интегрированностью" имеется в виду то, что базу данных можно представить как</b></p>

1. <p><b> объединение нескольких отдельных неизменяемых файлов данных</b></p>

2. <p><b> объединение нескольких отдельных изменяемых файлов данных</b></p>

3. <p><b>объект, характеризующийся набором значений некоторой совокупности атрибутов</b></p>

Ответ: 2

Комментарий: AP01401

<p><b>Что же такое файл?</b></p>

1. <p><b>Именованный набор байтов, который может быть сохранен на некотором накопителе.</b></p>

2. <p><b>Набор байтов, который может быть сохранен на некотором накопителе.</b></p>

3. <p><b>Именованный набор нулей и единиц, который может быть сохранен на некотором накопителе.</b></p>

Ответ: 1

Комментарий: AP01402

<p><b>Задачи редактирования файлов в БД?</b></p>

1. <p><b>Исключение из файла фрагмента текста, получение информации о количестве символов текста.</b></p>

2. <p><b> Вставка в файл фрагмента текста, исключение из файла фрагмента текста, упорядочивание элементов файла по определенному ключу.</b></p>

3. <p><b>Упорядочивание элементов файла по определенному ключу.</b></p>

Ответ: 2

Комментарий: AP01403

<p><b>Операцию замены можно выполнить с помощью</b></p>

1. <p><b>исключение из файла фрагмента текста и вставки в файл фрагмента текста.</b></p>

2. <p><b>упорядочивание элементов файла по определенному ключу и вставки в файл фрагмента текста.</b></p>

3. <p><b>исключение из файла фрагмента текста и замена фрагмента текста файла на другой фрагмент.</b></p>

Ответ: 1

Комментарий: AP01404

<p><b>Во время добавления данных в БД происходит аварийное закрытия файла. Но данные были сохраненный и не испорчены. По какой причине данные сохранились?</b></p>

1. <p><b>Утверждение не является верным. Данные были испорчены.</b></p>

2. <p><b>Пользователь успел применить сочетание клавиш ctrl+s.</b></p>

3. <p><b> в конце файла был записан *управляющий символ* конца файла.</b></p>

Ответ: 3

Комментарий: AP01405

<p><b>Работа с файлом осуществляется через (работает осуществляется помощи консоли) </b></p>

1. <p><b>таблицы</b></p>

2. <p><b>периферийное устройство</b></p>

3. <p><b>буфер</b></p>

Ответ: 3

Комментарий: AP01406

<p><b>Что произойдет после записи символа в требуемом месте?</b></p>

1. <p><b>Файла больше считываться не будет.</b></p>

2. <p><b>Файл считается до конца.</b></p>

3. <p><b>Файл закроется (символ не добавиться).</b></p>

Ответ: 1

Комментарий: AP01407

<p><b>Таким образом, для выполнения операций исключения, вставки и замены фрагментов текста в файлах БД можно использовать?</b></p>

1. <p><b>Два варианта верные.</b></p>

2. <p><b>Использовать вспомогательный файл.</b></p>

3. <p><b>Пользоваться массивами указателей.</b></p>

Ответ: 1

Комментарий: AP01408

<p><b>*Если файл* целиком нельзя разместить в памяти программы то</b></p>

1. <p><b>*файл* приходится считывать *по* частям и осуществлять перенос информации из одной части в другую.</b></p>

2. <p><b>*файл* приходится считывать полностью и делить его на части</b></p>

3. <p><b>файл самостоятельно уменьшиться в размере и считается</b></p>

Ответ: 1

Комментарий: AP01409

<p><b>Чтобы вставить строку в середину файла, используя вспомогательный *файл*, необходимо проделать следующий *алгоритм*. Является ли алгоритм верным?

1. Открыть входной файл в режиме чтения и вспомогательный файл в режиме записи (этот файл будет создан автоматически).

2.Прочесть из входного файла текст до места вставки.

3. Записать прочитанный фрагмент во вспомогательный файл.

4 Записать во вспомогательный файл вставляемый фрагмент.

5. Прочесть из входного файла остаточный фрагмент и записать его во вспомогательный файл.

6. Закрыть оба файла.</b></p>

1. <p><b>нет</b></p>

2. <p><b>частично</b></p>

3. <p><b>да</b></p>

Ответ: 3

Комментарий: AP01410

<p><b>Редактирование файла на программном уровне это</b></p>

1. <p><b>изменения, производимые с элементами файла после завершения программы </b></p>

2. <p><b> изменения, производимые с элементами файла в процессе работы</b></p>

3. <p><b>изменения, производимые с файлом в процессе работы</b></p>

Ответ: 2

Комментарий: AP014011

<p><b>Как поможет Метод seekg() в редактировании файл?</b></p>

1. <p><b>Произведёт установку текущей позиции в нужную.</b></p>

2. <p><b>Проверяет не достигнут ли конец файла.</b></p>

3. <p><b>Закрывает файл.</b></p>

Ответ: 1

Комментарий: AP014012

<p><b>Что значит строчка file.seekg(30,ios\_base::beg);</b></p>

1. <p><b>Стать на 31-й байт</b></p>

2. <p><b>Стать на 31 байт с конца</b></p>

3. <p><b>Перепрыгнуть через 3 байта</b></p>

Ответ: 1

Комментарий: AP014013

<p><b>Выберите простой способ форматирования данных</b></p>

1. <p><b>cin,cout</b></p>

2. <p><b>printf scanf</b></p>

3. <p><b>оба варианта верны</b></p>

Ответ: 3

Комментарий: AP014014

<p><b>Считаются ли флаги форматированием частью редактирования файла?</b></p>

1. <p><b>да</b></p>

2. <p><b>нет</b></p>

3. <p><b>зависит от использования</b></p>

Ответ: 3

Комментарий: AP014015

<p><b>Флаги форматирования класса ios dec выполняет функцию</b></p>

1. <p><b>перевода значения в десятичную форму</b></p>

2. <p><b>перевода значения в восьмеричную форму</b></p>

3. <p><b>перевода значения в шестнадцатеричную форму</b></p>

Ответ: 1

Комментарий: AP014016

<p><b>Флаги форматирования класса ios right выполняет функцию</b></p>

1. <p><b>выводит первое слово</b></p>

2. <p><b>выравнивание по правому краю</b></p>

3. <p><b>выводит последние слово</b></p>

Ответ: 2

Комментарий: AP014017

<p><b>Флаги форматирования класса ios fixed выполняет функцию</b></p>

1. <p><b>Экспоненциальный вывод чисел с плавающей запятой</b></p>

2. <p><b>Фиксированный вывод чисел с плавающей запятой</b></p>

3. <p><b>нету верного утверждения</b></p>

Ответ: 2

Комментарий: AP014018

<p><b>Флаги форматирования класса ios skipws выполняет функцию</b></p>

1. <p><b>перевод в восьмеричную форму</b></p>

2. <p><b>показывать десятичную точку при выводе</b></p>

3. <p><b> пропуск пробелов при вводе</b></p>

Ответ: 3

Комментарий: AP014019

<p><b>Инструкции редактирования, которые вставляются прямо в поток.</b></p>

1. <p><b>Манипуляторы</b></p>

2. <p><b>Структуры</b></p>

3. <p><b>Классы</b></p>

Ответ: 1

Комментарий: AP014020

<p><b>Манипуляторы бывают двух видов:

(укажите не верный вариант)</b></p>

1. <p><b>константным</b></p>

2. <p><b>с аргументом</b></p>

3. <p><b>без аргумента</b></p>

Ответ: 1

Комментарий: AP014021

<p><b>Манипуляторы ios setw() используется для</b></p>

1. <p><b>Устанавливает ширину поля для вывода данных</b></p>

2. <p><b>Устанавливает указанные флаги форматирования</b></p>

3. <p><b>Устанавливает указанные флаги форматирования</b></p>

Ответ: 1

Комментарий: AP014022

<p><b>Какая функция ostream устанавливает позицию в байтах файлового указателя относительно указанного места в файле?</b></p>

1. <p><b>seekp(position, seek\_dir)</b></p>

2. <p><b>tellp()</b></p>

3. <p><b>write()</b></p>

Ответ: 1

Комментарий: AP014023

<p><b>Какая функция ostream устанавливает SIZE символов из массива str в файл</b></p>

1. <p><b> seekp(position, seek\_dir)</b></p>

2. <p><b>write()</b></p>

3. <p><b>flush()</b></p>

Ответ: 2

Комментарий: AP014024

<p><b>Для чего создается массив с именем buff</b></p>

1. <p><b>для хранение измененных данных</b></p>

2. <p><b>для хранения всех данных</b></p>

3. <p><b>для передачи данных в другом порядке</b></p>

Ответ: 1

Комментарий: AP014025

<p><b>Для **манипулирования** **данными**, хранящимися **в** **БД**, используется группа операторов</b></p>

1. <p><b>sql</b></p>

2. <p><b>oracle</b></p>

3. <p><b>два ответа являются верными</b></p>

Ответ: 3

Комментарий: AP014026

<p><b>Файл test1.txt заполнены цифрами, все четные цифры записываться в test2.txt, а оставшиеся в test3.txt. Запишите строчку с ошибкой.<br>

#include <iostream><br>

#include <fstream><br>

using namespace std; <br>

void main()<br>

{<br>

ifstream ifs;<br>

ofstream ofs1; <br>

ofstream ofs2; <br>

string first\_file\_name = "C:\\programming\\test1.txt";<br>

string second\_file\_name = "C:\\programming\\test2.txt";<br>

string third\_file\_name = "C:\\programming\\test3.txt";<br>

ifs.open(first\_file\_name.c\_str());<br>

ofs1.open(second\_file\_name.c\_str());<br>

ofs2.open(third\_file\_name.c\_str());<br>

int buf; <br>

while (ifs >> buf) <br>

{<br>

if (buf % 2 != 0) <br>

ofs1 << buf; <br>

else<br>

ofs2 << buf; <br>

}<br>

ifs.close();<br>

ofs1.close();<br>

ofs2.close();<br>

}<br></b></p>

1. p><b>while (ifs >> buf)</b></p>

2. <p><b>if (buf % 2 != 0)</b></p>

3. <p><b>ifs.open(first\_file\_name.c\_str())</b></p>

Ответ: 2

Комментарий: AP014027

<p><b>Создан файл, содержащий десять цифр. цифры, отсортированные по возрастанию записываться в другой файл. Укажите строчку в которой есть ошибка. <br>

#include <iostream><br>

#include <fstream><br>

int main()<br>

{<br>

std::ifstream fin("number.txt");<br>

int m[10]; <br>

// сохранение чисел в массив<br>

for (int i = 0; i < 10; i++) {<br>

fin >> m[i]; <br>

}<br>

bool flag = false; // сортировка пузырьком<br>

while (!flag) {<br>

flag = true; <br>

for (int i = 0; i < 10; i++) {<br>

if (m[i] > m[i + 1]) {<br>

int tmp = m[i]; <br>

m[i] = m[i + 1]; <br>

m[i + 1] = tmp; <br>

flag = false; <br>

}<br>

}<br>

}<br>

std::ofstream fout("result.txt"); // запись отсортированных чисел в новый файл<br>

for (int i = 0; i < 10; i++) {<br>

fout << m[i] << " ";<br>

}<br>

return 0; <br>

}<br></b></p>

1. <p><b>flag = true; </b></p>

2. <p><b>if (m[i] > m[i + 1]) </b></p>

3. <p><b>bool flag = false; </b></p>

Ответ: 3

Комментарий: AP014028

<p><b>Пример чтения из одного файла и записи в другой с переводом символов в верхний регистр. Укажите строчку, в которой есть ошибка. <br>

#include <iostream><br>

#include <fstream><br>

#include<string><br>

#include <algorithm><br>

using namespace std; <br>

int main() {<br>

ifstream fin; // входной файл<br>

ofstream fout; // выходной файл<br>

string fin\_name("input.txt"), fout\_name("output.txt"), content("");<br>

// открываем файлы для работы<br>

fin.open(fin\_name); <br>

fout.open(fout\_name); <br>

if (fin.is\_open() || fout.is\_open()) {<br>

// чтение файла в строку<br>

getline(fin, content, '\0'); <br>

// закрытие вх. файла<br>

fin.close();<br>

// перевод букв в верхний регистр<br>

transform(content.begin(), content.end(), content.begin(), ::toupper); <br>

// запись строки в вых. файл<br>

fout << content; <br>

// закрытие вых. файла<br>

fout.close();<br>

}<br>

return 0; <br>

}<br></b></p>

1. <p><b>string fin\_name("input.txt"), fout\_name("output.txt"), content("");</b></p>

2. <p><b> if (fin.is\_open() || fout.is\_open())</b></p>

3. <p><b>fout.close();</b></p>

Ответ: 2

Комментарий: AP014029

<p><b>Дан файл f, компоненты которого являются целыми числами.  Файл f содержит столько же отрицательных чисел, сколько и положительных. Используя вспомогательный файл h, переписать компоненты файла f в файл g так, чтобы в файле g:сначала шли положительные, потом отрицательные числа. Укажите строчку, в которой есть ошибка. <br>

#include <iostream><br>

#include <fstream><br>

#include <Windows.h><br>

using namespace std; <br>

int main()<br>

{<br>

setlocale(LC\_ALL, "RUS");<br>

int a = 0; <br>

ifstream fin; <br>

ofstream fout; <br>

fout.open("g.txt");<br>

//fout.open("h.txt");// вспомогательный файл//конечный файл<br>

fin.open("f.txt");//начальный файл<br>

if (!(fin.is\_open()))//удалось ли открыть файл<br>

{<br>

cout << "Такого файла не существует!!"; <br>

Sleep(5000); <br>

return -1; <br>

}<br>

const int kol = 10; <br>

int A[kol]; <br>

for (int i(0); i < 0; i++)<br>

{<br>

a = 0; <br>

fin >> a; <br>

A[i] = a; //преобразовал жлементы файла в массив<br>

cout << A[i] << " ";<br>

}<br>

cout << endl << endl; <br>

int i = 0; <br>

for (int i(0); i < kol; i++)<br>

{<br>

while (A[i] > 0) <br>

{<br>

cout << A[i] << " ";<br>

i++;<br>

if (i > 2) <br>

break; <br>

}<br>

while (A[i] < 0) <br>

{<br>

cout << A[i] << " ";<br>

i++;<br>

if (i > 2) <br>

break; <br>

}<br>

}<br>

cout << endl; <br>

system("pause");<br>

return 0; <br>

}<br></b></p>

1. <p><b>fout.open("g.txt”)</b></p>

2. <p><b>const int kol = 10;</b></p>

3. <p><b>for (int i(0); i < 0; i++)</b></p>

Ответ: 3

Комментарий: AP014030

<p><b> Небольшая часть элемента, которая используется для управления сортировкой.</b></p

1.<p> <b>Якорь </b> </p>

2.<p> <b> Ключь </b> </p>

3.<p> <b> id </b> </p>

Ответ: 2

Комментарий: AP01601

<p><b> Конкретные характеристики ключей и элементов в разных приложениях могут существенно отличаться друг от друга, однако абстрактное понятие размещения ключей и связанной с ними информации в определенном порядке и представляет собой суть задачи …. </b></p>

1. <p><b> Взаимодействия </b></p>

2. <p><b>Компоновки </b></p>

3. <p><b> Компенсации </b></p>

4. <p><b> Сортировки </b></p>

Ответ: 4

Комментарий: AP01602

<p><b>Метод сортировки, при котором сортируемые файлы полностью помещаются в оперативной памяти, называется </b></p>

1. <p><b> Прямым </b></p>

2. <p><b> Внутренним </b></p>

3. <p><b> Полноценным</b></p>

4. <p><b> Конкретным </b></p>

Ответ: 2

Комментарий: AP01603

<p><b> Сортировка файлов, хранящихся на магнитной ленте или диске, называется </b></p>

1. <p><b> Обратным </b></p>

2. <p><b> Ограниченным </b></p>

3. <p><b> Вложенным </b></p>

4. <p><b>Внешней</b></p>

Ответ: 4

Комментарий: AP01604

<p><b> Функция (вставить слово) представляет собой шаблонную реализацию, которая обращается к сортируемым элементам только через первый аргумент и нескольких простых операций с данными. </b></p>

1. <p><b> stl </b></p>

2. <p><b> sort </b></p>

3. <p><b> qsort </b></p>

4. <p><b> caller </b></p>

Ответ: 2

Комментарий: AP01605

<p> <b> Сортировка, в которой последовательность выполняемых операций не зависит от упорядоченности данных, называется</b> </p>

1. <p> <b> Линейный </b> </p>
2. <p> <b> Неадаптивной </b> </p>
3. <p> <b> Последовательной</b> </p>
4. <p> <b> Нет правильного ответа </b> </p>

Ответ: 2

Комментарий: AP01606

<p> <b> Сортировка, в которой последовательность выполняемых операций зависит от упорядоченности данных, называется </b> </p>

1. <p><b> Адаптивной</b></p>

2. <p><b>Обратной</b></p>

3. <p><b> Непоследовательной </b></p>

4. <p><b> Абстрактной </b></p>

Ответ: 1

Комментарий: AP01607

<p> <b> Если метод сортировки сохраняет относительный порядок размещения в файле элементов с одинаковыми ключами, значит этот метод … </b> </p>

1. <p><b> Устойчивый </b></p>

2. <p><b> Сбалансированный </b></p>

3. <p><b > Упорядоченный </b></p>

4. <p><b> Одинарный </b></p>

Ответ: 1

Комментарий: AP01608

<p> <b> Если переупорядочиваются не сами элементы, а массив указателей так, что первый указатель указывает на наименьший элемент, следующий — на наименьший из оставшихся и т.д. эта сортировка является </b> </p>

1. <p><b> Прямой </b></p>

2. <p><b> Обрытный </b></p>

3. <p><b> Косвенной </b></p>

4. <p><b> Нет такой сортировки </b></p>

Ответ: 3

Комментарий: AP01609

<p> <b> Метод, работующий по принципу приведенным далее называется? Сначала находится наименьший элемент массива и меняется местами с элементом, стоящим первым в сортируемом массиве. Потом находится второй наименьший элемент и меняется местами с элементом, стоящим вторым в исходном массиве. Этот процесс продолжается до тех пор, пока весь массив не будет отсортирован. </b> </p>

1. <p><b> Сортировкой выбором </b></p>

2. <p><b> Пузырьковая сортировка</b></p>

3. <p><b> Сортировка расчёской</b></p>

4. <p><b> Сортировка перемешиванием</b></p>

Ответ: 1

Комментарий: AP01610

<p> <b> При каждом проходе алгоритма по внутреннему циклу, очередной наибольший элемент массива ставится на своё место в конце массива рядом с предыдущим «наибольшим элементом», а наименьший элемент перемещается на одну позицию к началу массива. Такой метод сортировки называется </b> </p>

1.<p><b> Чётно-нечётная сортировка </b></p>

2. <p><b> Шейкерная сортировка </b></p>

3. <p><b> Глупая сортировка </b></p>

4. <p><b> Сортировка пузырьком </b></p>

Ответ: 4

Комментарий: AP01611

<p> <b> Является ли сортировка пузырьком устойчивой </b> </p>

1.<p><b> Да </b></p>

2. <p><b> Нет </b></p>

3. <p><b> Она может быть как устойчивой, так и неустойчивой </b></p>

Ответ: 1

Комментарий: AP01612

<p> <b> Сортировка выбором выполняет порядка … сравнений и … обменов элементов. (Вставить правильный ответ)</b> </p>

1.<p><b> N2/ 4 сравнений, N / 2 обменов </b></p>

2. <p><b> N2/ 2 сравнений, N обменов </b></p>

3. <p><b> N2/ 4 сравнений, N обменов </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: AP016013

<p> <b> Сортировка вставками выполняет в среднем порядка …  сравнений и … полуобменов (перемещений). (Вставить правильный ответ)</b> </p>

1.<p><b> N2/ 4 сравнений, N2/ 4 полуобменов </b></p>

2. <p><b> N2/ 4 сравнений, N2/ 2 полуобменов </b></p>

3. <p><b> N2/ 2 сравнений, N2/ 4 полуобменов </b></p>

4. <p><b> Нет правильного ответа</b></p>

Ответ: 1

Комментарий: AP01614

<p> <b> Пузырьковая сортировка выполняет порядка …  сравнений и … обменов. (Вставить правильный ответ) </b> </p>

1.<p><b> N2/ 4 сравнений, N2/ 2 полуобменов </b></p>

2. <p><b> N2/ 4 сравнений, N2/ 4 полуобменов </b></p>

3. <p><b> N2/ 2 сравнений, N2/ 2 обменов</b></p>

4. <p><b> Нет правильного ответа</b></p>

Ответ: 3

Комментарий: AP01615

<p> <b>  Пара ключей, которые нарушают порядок в файле, называются </b> </p>

1.<p><b> Тождественное преобразование</b></p>

2. <p><b> Инверсией </b></p>

3. <p><b> Нет правильных ответов </b></p>

Ответ: 2

Комментарий: AP01616

<p> <b> Если отсортированному файлу добавляются нескольких элементов, либо в сортированном файле изменены ключи нескольких элементов, то для таких файлов наиболее эффективна сортировка является </b> </p>

1.<p><b> Сортировка выбором </b></p>

2. <p><b> Сортировка вставками </b></p>

3. <p><b> Пузырьковая сортировка</b></p>

4. <p><b>Все эти сортировки одинаково хороши </b></p>

Ответ: 2

Комментарий: AP01617

<p> <b> Время выполнения сортировки вставками зависит от </b> </p>

1.<p><b> Общего числа инверсий </b></p>

2. <p><b>Характера распределения инверсий </b></p>

3. <p><b> Как от числа, так и от распределения инверсий </b></p>

4. <p><b> Не зависит от инверсий </b></p>

Ответ: 1

Комментарий: AP01618

<p> <b> Время выполнения сортировки выбором линейно для файлов с  </b> </p>

1.<p><b> Большими элементами и малыми ключами </b></p>

2. <p><b> Малыми элементами и большими ключами </b></p>

3. <p><b> Большими элементами и большими ключами </b></p>

4. <p><b> Малыми элементами и малыми ключами </b></p>

Ответ: 1

Комментарий: AP01619

<p> <b> Когда к уже отсортированному файлу добавляются нескольких элементов, либо в сортированном файле изменены ключи нескольких элементов, то такие файлы называются </b> </p>

1.<p><b> Нарушенными </b></p>

2. <p><b> Изменчивые </b></p>

3. <p><b> Частично упорядоченные </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: AP01620

<p> <b> Для файлов небольших размеров сортировка вставками и сортировка выбором работают примерно в два раза быстрее пузырьковой сортировки. Утверждение является правдой? </b> </p>

1.<p><b> Да </b></p>

2. <p><b> Нет </b></p>

3. <p><b> Не хватает данных для ответа </b></p>

Ответ: 1

Комментарий: AP01621

<p> <b> Для подсчета количества инверсий в файле необходимо для каждого элемента просуммировать число элементов слева, которые больше его. Так ли это? </b> </p>

1.<p><b> Да </b></p>

2. <p><b> Нет </b></p>

3. <p><b> Нет ответа </b></p>

Ответ: 1

Комментарий: AP01622

<p> <b>  Сортировка Шелла представляет собой простое расширение метода вставок, быстродействие которого достигается за счет возможности … </b> </p>

1.<p><b> Сужать диапазон поиска элементов </b></p>

2. <p><b> Обмена далеко отстоящих друг от друга элементов </b></p>

3. <p><b> Обмена соседних элементов </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: AP01623

<p> <b> Устойчива ли сортировка Шелла? </b> </p>

1.<p><b> Да </b></p>

2. <p><b> Нет </b></p>

3. <p><b> Может быть как устойчивой, так и неустойчивой </b></p>

Ответ: 2

Комментарий: AP01624

<p> <b> Лучшее время работы сортировки Шелла </b> </p>

1.<p><b> O(nlogn) </b></p>

2. <p><b> O(n log2 n) </b></p>

3. <p><b> O(n) </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: AP01625

<p> <b> Худшее время работы сортировки Шелла </b> </p>

1.<p><b> O(n^2) </b></p>

2. <p><b> O(nlogn) </b></p>

3. <p><b> O(nlgn) </b></p>

4. <p><b> O(nloglogn) </b></p>

Ответ: 1

Комментарий: AP01626

<p> <b> Худшее время работы сортировки выбором </b> </p>

1.<p><b> О(n^2) </b></p>

2. <p><b> O(nlgn) </b></p>

3. <p><b> O(n log2 n) </b></p>

4. <p><b> O(nloglogn) </b></p>

Ответ: 1

Комментарий: AP01627

<p> <b> Лучшее время работы сортировки выбором </b> </p>

1.<p><b> O(nlogn) </b></p>

2. <p><b> O(n) </b></p>

3. <p><b> О(n^2) </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: AP01628

<p> <b> Лучшее время работы сортировки вставками </b> </p>

1.<p><b> O(nlogn) </b></p>

2. <p><b> O(n) </b></p>

3. <p><b> О(n^2) </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: AP01629

<p> <b> Худшее время работы сортировки вставками </b> </p>

1.<p><b> O(nlgn) </b></p>

2. <p><b> O(nloglogn) </b></p>

3. <p><b> O(n^2) </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: AP01630

ранее сохраненных данных это? </b></p>

1. <p><b> Анализ </b></p>

2. <p><b> Поиск </b></p>

3. <p><b> Выборка </b></p>

4. <p><b> Извлечение </b></p>

Ответ: 2

Комментарий: AP01801

<p><b>Структура данных элементов с ключами, которая поддерживает две базовые операции: вставку нового элемента и возврат элемента с заданным ключом, называется?</b></p>

1. <p><b> Таблица символов</b></p>

2. <p><b> Бинарный код </b></p>

3. <p><b> Массив </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: AP01802

<p><b> Язык С++ позволяет создавать типы данных, которые ведут себя аналогично базовым типам языка Си. Такие типы обычно называют </b></p>

1. <p><b>Абстрактный тип данных</b></p>

2. <p><b> Примитивный тип данных </b></p>

3. <p><b> Взаимный тип данных </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: AP01803

<p><b>  Если значения ключей - положительные целые числа, меньшие M, и элементы имеют различные ключи, то тип данных таблицы символов может быть реализован с помощью индексированных значениями ключей массивов так, что для выполнения операций вставить, найти и удалить потребуется </b></p>

1. <p><b> Постоянное время </b></p>

2. <p><b> Всремя не зависит от значения ключа </b></p>

3. <p><b> N/2 </b></p>

4. <p><b> Мало данных для ответа </b></p>

Ответ: 1

Комментарий: AP01804

<p><b>  Последовательный поиск в таблице символов с N элементами требует выполнения порядка … сравнений при успешном поиске (в среднем). </b></p>

1. <p><b> N </b></p>

2. <p><b>N/4 </b></p>

3. <p><b>2N/3 </b></p>

4. <p><b> N/2 </b></p>

Ответ: 4

Комментарий: AP01805

<p><b> Последовательный поиск в таблице символов с N неупорядоченными элементами требует … количества шагов для выполнения вставок и … сравнений при неудачном поиске (всегда).</b></p>

1. <p><b> постоянного кол-во шагов, N/2 сравнений </b></p>

2. <p><b> постоянного кол-во шагов, N сравнений </b></p>

3. <p><b> N кол-во шагов, N/2 сравнений </b></p>

4. <p><b> N/2 кол-во шагов, N сравнений </b></p>

Ответ: 2

Комментарий: AP01806

<p><b> Последовательный поиск в таблице символов из N упорядоченных элементов требует порядка … операций для вставки, успешного поиска и неудачного поиска (в среднем). </b></p>

1. <p><b> N </b></p>

2. <p><b> N/2 </b></p>

3. <p><b> N^2/2 </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: AP01807

<p><b> Алгоритм, который изменяет порядок элементов так, чтобы часто запрашиваемые элементы встречались в начале поиска, называется? </b></p>

1. <p><b> Аглоритм интерполирующего поиск </b></p>

2. <p><b> Алгоритм самоорганизующегося поиска </b></p>

3. <p><b> Алгоритм значемого поиска </b></p>

Ответ: 2

Комментарий: AP01808

<p><b> Классический алгоритм поиска элемента в отсортированном массиве, использующий дробление массива на половины называется </b></p>

1. <p><b> Линейный поиск  </b></p>

2. <p><b> Бинарный поиск </b></p>

3. <p><b> Интерполирующий поиск </b></p>

4. <p><b> Поиск подстроки в строке  </b></p>

Ответ: 2

Комментарий: AP01809

<p><b> При бинарном поиске выполняется не более чем … сравнений (и при успешном, и при неудачном). </b></p>

1. <p><b> N </b></p>

2. <p><b> N/2 </b></p>

3. <p><b> N - 1 </b></p>

4. <p><b> [lgN] + 1  </b></p>

Ответ: 4

Комментарий: AP01810

<p><b> Наибольший недостаток использования бинарного поиска </b></p>

1. <p><b> Потенциально высокие затраты памяти </b></p>

2. <p><b> Для поиска массив должен быть упорядочен </b></p>

3. <p><b> Длительное время работы </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 2

Комментарий: AP01811

<p><b> Достоинства бинарного поиска </b></p>

1. <p><b>  Быстрая работа алгоритма </b></p>

2. <p><b> Прост в реализации </b></p>

3. <p><b> Практичность </b></p>

4. <p><b> Все ответы верны </b></p>

Ответ: 4

Комментарий: AP01812

<p><b>  Алгоритм который перебирает все элементы в массиве, сравнивая их с заданным ключом называется </b></p>

1. <p><b> Линейный алгоритм </b></p>

2. <p><b> Бинарный алгоритм </b></p>

3. <p><b> [Интерполирующий поиск С++](https://purecodecpp.com/archives/2503)</b></p>

4. <p><b> Такого алгоритма нету </b></p>

Ответ: 1

Комментарий: AP01813

<p><b> Иерархическая структура данных, в которой каждый узел имеет значение (оно же является в данном случае и ключом) и ссылки на левого и правого потомка, называется </b></p>

1. <p><b> Родовая структура </b></p>

2. <p><b> Иерархическое дерево </b></p>

3. <p><b> Бинарное дерево </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 3

Комментарий: AP01814

<p><b> Бинарное дерево поиска (binary search tree - BST) — это бинарное дерево, обладающее дополнительными свойствами, какими? </b></p>

1. <p><b> значение левого потомка меньше значения родителя, а значение правого потомка больше значения родителя для каждого узла дерева. </b></p>

2. <p><b> значение левого потомка больше значения родителя, а значение правого потомка меньше значения родителя для каждого узла дерева. </b></p>

3. <p><b> значение левого потомка меньше значения родителя и значение правого потомка меньше значения родителя для каждого узла дерева. </b></p>

4. <p><b> значение левого потомка меньше значения родителя, а значение правого потомка равно значению родителя для каждого узла дерева. </b></p>

Ответ: 1

Комментарий: AP01815

<p><b> BST-деревья используются для реализации операций </b></p>

1. <p><b> Найти </b></p>

2. <p><b> Вставить </b></p>

3. <p><b> Создать </b></p>

4. <p><b> Найти, вставить, создать и подсчитать </b></p>

Ответ: 4

Комментарий: AP01816

<p><b> Как ещё называют таблицу символов </b></p>

1. <p><b> Карта </b></p>

2. <p><b> Таблица неясности </b></p>

3. <p><b> Словарь </b></p>

Ответ: 3

Комментарий: AP01817

<p><b> В дереве бинарного поиска, образованном N случайными ключами, для успешного поиска в среднем требуется около … сравнений </b></p>

1. <p><b> 2lgN ≈ 1.39lgN </b></p>

2. <p><b> lgN </b></p>

3. <p><b> 2lgN ≈ 1.50lgN </b></p>

4. <p><b> N </b></p>

Ответ: 1

Комментарий: AP01818

<p><b> В дереве бинарного поиска, образованном N случайными ключами, для вставок и неудачного поиска в среднем требуется около … сравнений. </b></p>

1. <p><b> 2lgN ≈ 1.50lgN </b></p>

2. <p><b> lgN </b></p>

3. <p><b> 2lgN ≈ 1.39lgN </b></p>

4. <p><b> N </b></p>

Ответ: 3

Комментарий: AP01819

<p><b>  Для поиска в дереве бинарного поиска с N ключами в худшем случае может потребоваться … сравнений. </b></p>

1. <p><b> lgN </b></p>

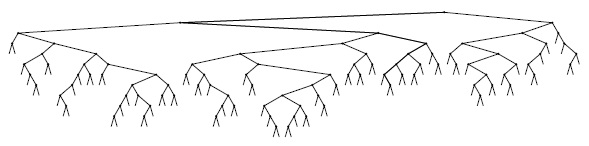
2. <p><b> N </b></p>

3. <p><b> 2N </b></p>

4. <p><b> 1/2\*N </b></p>

Ответ: 2

Комментарий: AP01820

<p><b> Что изображено на рисунке? </b></p>

1. <p><b> Интерполяционный поиск </b></p>

2. <p><b> Фактографический поиск </b></p>

3. <p><b> Дерево бинарного поиска </b></p>

Ответ: 3

Комментарий: AP01821

<p><b> Функция, которая вызывает сама себя, называется </b></p>

1. <p><b> Саморегулируемая функция </b></p>

2. <p><b> Обратная функция </b></p>

3. <p><b> Рекурсивная функция </b></p>

4. <p><b> Функция ротации </b></p>

Ответ: 3

Комментарий: AP01822

<p><b> Динамическая структура, в которой хранится информация о функциях, выполняющихся в процессоре, назывется </b></p>

1. <p><b> Стек вызовов </b></p>

2. <p><b> Стек памяти </b></p>

3. <p><b> Стек динамической функции </b></p>

4. <p><b> Нет правильного ответа </b></p>

Ответ: 1

Комментарий: AP01823

<p><b>  Узел, содержащий ключ S, перемещается в дереве вниз и становится правым дочерним узлом своего прежнего левого дочернего узла. Это?</b></p>

1. <p><b> Ротация влево в BST-дереве </b></p>

2. <p><b> Ротация вправо в BST-дереве </b></p>

3. <p><b> Без направленная ротация в BST-дереве </b></p>

4. <p><b> Правый дочерний узел </b></p>

Ответ: 2

Комментарий: AP01824

<p><b> Узел, содержащий ключ A, перемещается в дереве вниз и становится левым дочерним узлом своего прежнего правого дочернего узла. Это? </b></p>

1. <p><b> Ротация влево в BST- </b></p>

2. <p><b> Ротация вправо в BST-дереве </b></p>

3. <p><b> Левый дочерний узел </b></p>

Ответ: 1

Комментарий: AP01825

<p><b> Локальное изменение позволяющие перемещать узлы по деревьям без изменения глобальных свойств упорядоченности, которые и делают BST-дерево полезной для поиска структурой, это? </b></p>

1. <p><b> Аменция </b></p>

2. <p><b> Проекция </b></p>

3. <p><b> Ротация </b></p>

4. <p><b> Абсолюция </b></p>

Ответ: 3

Комментарий: AP01826

<p><b> Ротации применяются для </b></p>

1. <p><b> Удаления определенных узлов дерева, для последующего использования. </b></p>

2. <p><b> Перемещения конкретных узлов по дереву и предотвращения разбалансировки деревьев. </b></p>

3. <p><b> Выбора и последующего перемещения для разбалансировки дерева. </b></p>

Ответ: 2

Комментарий: AP01827

<p><b> Что такое вес бинарного дерева? </b></p>

1. <p><b> Число листьев </b></p>

2. <p><b> Число узлов </b></p>

3. <p><b> Число узлов, ссылающихся на нулевой элемент </b></p>

4. <p><b> Число дуг дерева </b></p>

Ответ: 1

Комментарий: AP01828

<p><b> Выберите утверждения, верные для идеально сбалансированного дерева.

</b></p>

1. <p><b> Выберите утверждения, верные для идеально сбалансированного дерева. </b></p>

2. <p><b> Каждый уровень содержит не более узлов, где корень - нулевой узел. </b></p>

3. <p><b> Для каждой его вершины высота ее двух поддеревьев различается не более, чем на 1 </b></p>

4. <p><b> Для поиска любого элемента перебирается не более log2n вершин, где n - число вершин в дереве </b></p>

Ответ: 1, 3, 4

Комментарий: AP01829

<p><b> Сбалансированным бинарным деревом называется... </b></p>

1. <p><b> Дерево, для каждой вершины которого высота двух его поддеревьев различается не более чем на единицу</b></p>

2. <p><b> Дерево, элементы которого распределены таким образом, что слева от корня находятся элементы с меньшим индексом, а справа - с большим</b></p>

3. <p><b> Дерево, в котором у каждого поддерева имеется пара потомков </b></p>

4. <p><b> Дерево, индекс корня которого равен среднему арифметическому всех остальных индексов </b></p>

Ответ: 1

Комментарий: AP01830

**<p><b>1.Как правильно задать стек в с++?(При помощи стандартной библиотеки)</b></p>**

**1. <p><b>stuck <int> <stk>;</b></p>**

**2. <p><b> stack <int> stk; </b></p>**

**3. <p><b>stk int <stk>;</b></p>**

**4. <p><b> steck int stk; </b></p>**

**Ответ: 2**

Комментарий: AP00401

**<p><b>2.Какая функция добавляет элемент в стек?**

1. **<p><b> empty()</b></p>**

**2. <p><b>pop()</b></p>**

**3. <p><b>push()</b></p>**

**4. <p><b> top()</b></p>**

**Ответ: 3**

Комментарий: AP00402

**<p><b>3. Какая функция убирает элемент из стека? </b></p>**

**1. <p><b>empty()</b></p>**

**2. <p><b> pop()</b></p>**

**3. <p><b>push()</b></p>**

**4. <p><b>top()</b></p>**

**Ответ: 2**

Комментарий: AP00403

**<p><b>4. Какую библиотеку необходимо подключить, чтобы задать стек таким образом: (stack <int> stk)? </b></p>**

**1. <p><b>steck </b></p>**

**2. <p><b>stk</b></p>**

**3. <p><b>stack</b></p>**

**4. <p><b>стек можно задать без подключения специальной библиотеки</b></p>**

**Ответ: 3**

Комментарий: AP00404

**<p><b>5. Функция pop() возвращает удаленный символ стека. Если эту функцию применить к пустому стеку, то она вернет символ с кодом: </b></p>**

**1. <p><b>254</b></p>**

**2. <p><b>255</b></p>**

**3. <p><b>256</b></p>**

**4. <p><b>257</b></p>**

**Ответ: 2.**

Комментарий: AP00405

**<p><b>6. На фрагменте кода представлена работа со стеком. Если во время ввода элементов стека ввести элементы 1, 2, f, то: </b></p>**

**<p><b>**

**Int I = 1; <br>**

**Stack<int> stk; <br>**

**While(I <= 5) <br>**

**{<br>**

**Int a; <br>**

**Cin a; <br>**

**Stk.push(a); <br>**

**I++;<br>**

**}<br>**

**I = 1; <br>**

**While(I <= 5) <br>**

**{<br>**

**Cout<<stk.top()<<endl; <br>**

**Stk.pop();<br>**

**I++;<br>**

**}<br>**

**</b></p>**

**1. <p><b>Программа выведет: 0 0 0 2 1</b></p>**

**2. <p><b>Программа выведет: 0 0 f 2 1</b></p>**

**3. <p><b>Программа проигнорирует f и запросит ввести 4 элемент</b></p>**

**3. <p><b>Программа выдаст ошибку и ничего не выведет</b></p>**

**Ответ: 1**

Комментарий: AP00406

**<p><b>7. Какая функция проверяет, пуст ли стек? </b></p>**

**1. <p><b>empty()</b></p>**

**2. <p><b>pop()</b></p>**

**3. <p><b>push()</b></p>**

**4. <p><b>top()</b></p>**

**Ответ: 1**

Комментарий: AP00407

**<p><b>8.В стеке , включение и исключение элементов которого осуществляется в соответствии с правилами: </b></p>**

**1. <p><b>"последним введен, последним выведен"</b></p>**

**2. <p><b>"первым введен, последним выведен"</b></p>**

**3. <p><b>"первым введен, первым выведен"</b></p>**

**4. <p><b>"последним введен, первым выведен"</b></p>**

**Ответ: 2 4**

Комментарий: AP00408

**<p><b>9. стек можно задать</b></p>**

**1. <p><b> при помощи массива</b></p>**

**2. <p><b>при помощи списка</b></p>**

**3. <p><b>при помощи стандартной библиотеки <stack></b></p>**

**4. <p><b> при помощи очереди</b></p>**

**Ответ: 1 2 3**

Комментарий: AP00409

**<p><b>10. Какая функция обращается к последнему элементу стека? </b></p>**

**1. <p><b>empty()</b></p>**

**2. <p><b>pop()</b></p>**

**3. <p><b>push()</b></p>**

**4. <p><b>top()</b></p>**

**Ответ: 4**

Комментарий: AP00410

**<p><b>11. Стек - это</b></p>**

**1. <p><b>Упорядоченный набор элементов, в котором добавление новых и удаление существующих элементов допустимо с любого конца. </b></p>**

**2. <p><b> Упорядоченный набор элементов, в котором добавление новых элементов допустимо с одного конца, а удаление существующих элементов – только с другого конца. </b></p>**

**3. <p><b> Упорядоченный набор элементов, в котором добавление новых и удаление существующих элементов допустимо только с одного конца. </b></p>**

**Ответ: 3**

Комментарий: AP00411

**<p><b>12. стек можно реализовать </b></p>**

**1. <p><b>только как статическую структуру</b></p>  
2. <p><b> только как динамическую структуру</b></p>  
3. <p><b> как статическую или динамическую структуру</b></p>**

**Ответ: 3**

Комментарий: AP00412

**<p><b>13. Какая функция проверяет размер стека? </b></p>**

**1. <p><b> empty()</b></p>**

**2. <p><b>pop()</b></p>**

**3. <p><b> push()</b></p>**

**4. <p><b>size()</b></p>**

**Ответ: 4**

Комментарий: AP00413

**<p><b>14. На фрагменте кода представлена работа со стеком. Сколько раз программа выведет текст “Стек пуст”?</b></p>**

**<p><b>**

**Int I = 1; <br>**

**Stack<int> stk; <br>**

**While(I <= 5) <br>**

**{<br>**

**Int a; <br>**

**Cin a; <br>**

**Stk.push(a); <br>**

**I++;<br>**

**}<br>**

**I = 1; <br>**

**Stk.pop();<br>**

**Stk.pop();<br>**

**While(I <= 5) <br>**

**{<br>**

**If(stk.size() == 0) <br>**

**{<br>**

**Cout<<”Стек пуст”<<endl; <br>**

**}<br>**

**Else**

**{<br>**

**Cout<<stk.top()<<endl; <br>**

**Stk.pop();<br>**

**}<br>**

**I++;<br>**

**}<br>**

**</b></p>**

1. **<p><b>0</b></p>  
   2. <p><b>1</b></p>  
   3. <p><b> 2</b></p>  
   ответ: 3**
2. Комментарий: AP00414

**<p><b>15. На фрагменте кода представлена работа со стеком. Что нужно написать вместо комментария, чтобы программа вывела текст “Стек пуст”?</b></p>**

**<p><b>**

**Int I = 1; <br>**

**Stack<int> stk; <br>**

**While(I <= 5) <br>**

**{<br>**

**Int a; <br>**

**Cin a; <br>**

**Stk.push(a); <br>**

**I++;<br>**

**}<br>**

**I = 1; <br>**

**\\???<br>**

**Stk.pop();<br>**

**Stk.pop();<br>**

**While(I <= 5) <br>**

**{<br>**

**If(stk.size() == 0) <br>**

**{<br>**

**Cout<<”Стек пуст”<<endl; <br>**

**}<br>**

**Else**

**{<br>**

**Cout<<stk.top()<<endl; <br>**

**Stk.pop();<br>**

**}<br>**

**I++;<br>**

**}<br>**

**</b></p>**

**1. <p><b> empty();</b></p>**

**2. <p><b> pop();</b></p>**

**3. <p><b> push();</b></p>**

**4. <p><b> top();</b></p>**

**Ответ: 2**

Комментарий: AP00415

**<p><b>16. На фрагменте кода представлена работа со стеком. Какую функцию необходимо написать вместо комментария в 22 строке, чтобы вывести 1 раз текст “Стек пуст”?</b></p>**

**<p><b>**

**Int I = 1; <br>**

**Stack<int> stk; <br>**

**While(I <= 4) <br>**

**{<br>**

**Int a; <br>**

**Cin a; <br>**

**Stk.push(a); <br>**

**I++;<br>**

**}<br>**

**I = 1; <br>**

**Stk.pop();<br>**

**Stk.pop();<br>**

**While(I <= 5) <br>**

**{<br>**

**If(stk.\\??? == 1) <br>**

**{<br>**

**Cout<<”Стек пуст”<<endl; <br>**

**}<br>**

**Else**

**{<br>**

**Cout<<stk.top()<<endl; <br>**

**Stk.pop();<br>**

**}<br>**

**I++;<br>**

**}<br>**

**</b></p>**

**1. <p><b>empty();</b></p>**

**2. <p><b> pop();</b></p>**

**3. <p><b> size();</b></p>**

**4. <p><b>top();</b></p>**

**Ответ: 1**

Комментарий: AP00416

**<p><b>17. На фрагменте кода представлена работа со стеком. Какую функцию необходимо написать вместо комментария в 22 строке, чтобы вывести 2 раза текст “Стек пуст”?</b></p>**

**1. <p><b> empty();</b></p>**

**2. <p><b>pop();</b></p>**

**3. <p><b>size();</b></p>**

**4. <p><b> top();</b></p>**

**Ответ: 3**

Комментарий: AP00417

**<p><b>18. Для записи числа в стек используется</b></p>**

**1. <p><b>косвенно-регистровый метод адресации</b></p>  
2. <p><b>относительный метод адресации</b></p>  
3. <p><b>автодекрементный метод адресации</b></p>  
4. <p><b>регистровый метод адресации</b></p>**

**Ответ: 3**

Комментарий: AP00418

**<p><b>19. Для чтения числа из стека используется</b></p>**

**1. <p><b> автоинкрементный метод адресации</b></p>  
2. <p><b> абсолютный метод адресации</b></p>  
3. <p><b>индексный метод адресации</b></p>  
4. <p><b>относительный метод адресации</b></p>**

**Ответ: 1**

Комментарий: AP00419

**<p><b>20. Для запоминающего устройства, организованного в виде стека, имеет место следующее</b></p>**

**1. <p><b> В каждый момент времени для чтения доступно любая ячейка независимо от ее расположения</b></p>  
2. <p><b> Для доступа к данным надо указать адрес ячейки памяти</b></p>  
3. <p><b> Для доступа к данным не надо указывать адрес ячейки памяти</b></p>  
4. <p><b> В каждый момент времени для чтения доступна только ячейка памяти, являющаяся вершиной стека</b></p>**

**Ответ: 3 4**

Комментарий: AP00420

**<p><b>21. Укажите верное(ые) утверждение(я) о стеке</b></p>**

**1. <p><b>Информация о прочитанных один раз данных не теряется</b></p>  
2. <p><b>Прочитать слово, находящееся на вершине стека, можно только один раз</b></p>  
3. <p><b> Информация о прочитанных один раз данных теряется</b></p>  
4. <p><b> Записанные данные могут быть последовательно прочитаны только в порядке, обратном порядку их записи</b></p>**

**Ответ: 3 4**

Комментарий: AP00421

**<p><b>22. Регистр - указатель стека всегда содержит</b></p>**

**1. <p><b>первое записанное в стек число</b></p>  
2. <p><b>адрес вершины стека</b></p>  
3. <p><b>адрес последней записанной ячейки стека</b></p>  
4. <p><b> последнее записанное в стек число</b></p>**

**Ответ: 2 3**

Комментарий: AP00422

**<p><b>23.Способ доступа к ячейкам памяти организованным в виде стека</b></p>**

**1. <p><b> Данные, записанные последними, читаются первыми</b></p>  
2. <p><b> В каждый момент времени можно прочитать данные из любой ячейки</b></p>  
3. <p><b>Данные, записанные последними, читаются последними</b></p>  
4. <p><b>Данные, записанные первыми, читаются последними</b></p>**

**Ответ: 1 4**

Комментарий: AP00423

**<p><b>24. Стек является структурой: </b></p>**

1. **<p><b> С организационной системой доступа к данным</b></p>  
   2. <p><b> С произвольной системой доступа к данным</b></p>  
   3. <p><b>Со смешанной системой доступа к данным</b></p>  
   Ответ: 1**

Комментарий: AP00424

**<p><b>25. Способ доступа к ячейкам памяти с произвольным доступом</b></p>**

**1. <p><b> Данные, записанные первыми, читаются последними</b></p>  
2. <p><b> Данные, записанные последними, читаются последними</b></p>  
3. <p><b>Данные, записанные первыми, читаются первыми</b></p>  
4. <p><b>В каждый момент времени можно прочитать данные из любой ячейки</b></p>**

Комментарий: AP00425

**Ответ: 4**

**<p><b>26. Стек – это: </b></p>**

**1. <p><b>последовательный контейнер, обеспечивающий вставку и удаление элементов за постоянное время. Не предоставляет произвольный доступ к своим элементам. </b></p>  
2. <p><b>последовательный контейнер, обеспечивающий вставку элемента в вершину контейнера и удаление элемента из вершины контейнера. </b></p>  
3. <p><b> последовательный контейнер, обеспечивающий добавление элементов в конец и извлечение элементов с начала контейнера. </b></p>  
4. <p><b> структура, реализованная при помощи очереди на основе контейнера, допускающего произвольный доступ к элементам</b></p>**

**Ответ: 2**

Комментарий: AP00426

**<p><b>27. Укажите верное утверждение</b></p>**

**1. <p><b>Стек является абстрактной структурой данных</b></p>  
2. <p><b> Стек не является абстрактной структурой данных</b></p>**

**Ответ: 1**

Комментарий: AP00427

**<p><b>28. Укажите верное(ые) утверждение(я) о стеке</b></p>**

**1. <p><b>В каждый момент времени для чтения доступна только ячейка памяти, являющаяся вершиной стека</b></p>  
2. <p><b> Для записи числа в стек используется косвенно-регистровый метод адресации</b></p>  
3. <p><b>Стек - это Упорядоченный набор элементов, в котором добавление новых и удаление существующих элементов допустимо с любого конца. </b></p>**

**Ответ: 1**

Комментарий: AP00428

**<p><b>29. Возможно ли использовать динамический массив в качестве стека? </b></p>**

**1. <p><b>Да</b></p>  
2. <p><b>Нет</b></p>**

**Ответ: 1**

Комментарий: AP00429

**<p><b>30. Укажите верное(ые) утверждение(я) о стеке**

**1. <p><b>Функция size() проверяет размер стека</b></p>  
2. <p><b> Стек можно задать через массив, список </b></p>  
3. <p><b> Функция push() удаляет элемент стека</b></p>  
4. <p><b> Стек является структурой со смешанной системой доступа к данным </b></p>**

**Ответ: 1 2**

Комментарий: AP00430

**01\_Выбор структур данных**

**<p><b>Виды линейных структур данных: </b></p>**

1. <p><b>Массив</b></p>
2. <p><b>Словарь данных</b></p>
3. <p><b>Динамический массив</b></p>
4. <p><b>Связные списки</b></p>

Ответ: 1, 3, 4

Комментарий: SD001001

**<p><b>Что такое массив? </b></p>**

1. <p><b>структура данных, хранящая набор значений идентифицируемых по индексу</b></p>
2. <p><b>пользовательский(собственный) тип данных</b></p>
3. <p><b>специальный блок инструкций, вызываемый при создании объекта</b></p>
4. <p><b>область в памяти, обеспечивающая начальные значения состояний</b></p>

Ответ: 1

Комментарий: SD001002

**<p><b>Основные операции с массивами: </b></p>**

1. <p><b>Вставка элемента</b></p>
2. <p><b>Получение элемента</b></p>
3. <p><b>Получение общего количества элементов в массиве</b></p>
4. <p><b>Получение последнего элемента и его удаление</b></p>

Ответ: 1, 2, 3

Комментарий: SD001003

**<p><b>Как создать массив, вмещающий в себя 5 элементов целочисленного типа? </b></p>**

1. <p><b>float arr[5] = {0};</b></p>
2. <p><b>int arr{5} = [0]; </b></p>
3. <p><b>int arr [4] = {0};</b></p>
4. <p><b>int arr[5] = {0};</b></p>

Ответ: 4

Комментарий: SD001004

**<p><b>Каким образом можно добавить ещё один элемент в простой массив? (arr – название массива, NUM –  const, кол-во элементов массива, n – элемент, который нужно добавить) </b></p>**

1. <p><b>arr[NUM + 1] =  n; </b></p>
2. <p><b>arr[NUM – 1] = n; </b></p>
3. <p><b>arr[NUM] = n; </b></p>
4. <p><b>Нет правильного ответа</b></p>

Ответ: 4

Комментарий: SD001005

**<p><b>Как вывести в консоль все элементы массива? (arr – название массива, NUM – const, кол-во элементов массива) </b></p>**

1. <p><b> for (int i = 0; i < NUM; i++) cout << arr[i] << endl; </b></p>
2. <p><b>for (int i = 0; i <= NUM; i++) cout << arr[i] << endl; </b></p>
3. <p><b>for (int i = 1; i <= NUM; i++) cout << arr[i] << endl; </b></p>
4. <p><b>for (int i = 1; i < NUM; i++) cout << arr[i] << endl; </b></p>

Ответ: 1

Комментарий: SD001006

**<p><b>Недостатки линейного массива: </b></p>**

1. <p><b>Структурированность </b></p>
2. <p><b>Неизменное количество объектов</b></p>
3. <p><b>Сложность в  использовании</b></p>
4. <p><b>Нагрузка на систему в отличии от создание большого количества переменных</b></p>

Ответ: 2

Комментарий: SD001007

**<p><b>Как заполнить массив, вмещающий в себя 10 элементов? </b></p>**

1. <p><b>for(int i = 1; i < 10; i++) cin >> a[i];</b></p>
2. <p><b> for(int i = 0; i < 10; i++) cin >> a[i];</</b></p>
3. <p><b> for(int i = 0; i <= 10; i++) cin >> a[i];</ </b></p>
4. <p><b> for(int i = 1; i <= 10; i++) cin >> a[i];</</b></p>

Ответ: 2

Комментарий: SD001008

**<p><b>Результатом выполнения программы ниже будет:<br>int a[1];<br>for(int i = 0; i < 5; i++)<br>&ensp;cout <&lt; a[i];</b></p>**

1. <p><b> будет выведено 5 случайны чисел, хранящихся в памяти(“мусор”)</b></p>
2. <p><b>будет выведено 5 единиц </b></p>
3. <p><b>выведет ошибку</b></p>
4. <p><b>выведет 5 нулей</b></p>

Ответ: 1

Комментарий: SD001009

**<p><b>Операторы, используемые для выделения динамического массива и работы с ним: </b></p>**

1. <p><b>new[]</b></p>
2. <p><b>array</b></p>
3. <p><b>delete[]</b></p>
4. <p><b>length</b></p>

Ответ: 1, 3

Комментарий: SD001010

**<p><b>Может ли динамический массив содержать в себе одномерные массивы? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001011

**<p><b>Виды линейных структур данных с конечными точками: </b></p>**

1. <p><b>Динамический массив</b></p>
2. <p><b>Очередь с приоритетами</b></p>
3. <p><b>Стек</b></p>
4. <p><b>Очередь</b></p>

Ответ: 2, 3, 4

Комментарий: SD001012

**<p><b>Способы реализации стека: </b></p>**

1. <p><b>С помощью одномерного массива</b></p>
2. <p><b>С помощью связанного списка</b></p>
3. <p><b>С помощью класса объектно-ориентированного программирования </b></p>
4. <p><b>Нет правильных ответов</b></p>

Ответ: 1, 2, 3

Комментарий: SD001013

**<p><b>Что из себя представляют связные списки? </b></p>**

1. <p><b>область памяти, где могут последовательно храниться несколько значений</b></p>
2. <p><b>динамические структуры данных, в которых однотипные элементы связаны между собой</b></p>
3. <p><b>имена кусочков памяти, которые могут хранить информацию</b></p>
4. <p><b>многократное прохождение по одному и тому же коду программы</b></p>

Ответ: 2

Комментарий: SD001014

**<p><b>Типы связных списков: </b></p>**

1. <p><b>Двунаправленный</b></p>
2. <p><b>Кольцевой</b></p>
3. <p><b>Перекрёстный</b></p>
4. <p><b>Однонаправленный</b></p>

Ответ: 1, 4

Комментарий: SD001015

**<p><b>Самый простой вид связных списков? </b></p>**

1. <p><b>Линейный двусвязной список</b></p>
2. <p><b>Кольцевой список</b></p>
3. <p><b>Линейный односвязный список</b></p>
4. <p><b>Многосвязный список</b></p>

Ответ: 3

Комментарий: SD001016

**<p><b>Одними из операций, которые включают в себя списки, являются: </b></p>**

1. <p><b>добавление нового звена списка</b></p>
2. <p><b>сортировка списка</b></p>
3. <p><b>создание ведущего звена</b></p>
4. <p><b>перестановка всех его звеньев в обратном порядке</b></p>

Ответ: 1, 2, 3, 4

Комментарий: SD001017

**<p><b>Недостатки односвязных списков: </b></p>**

1. <p><b>усложнение взаимодействия операций поиска и удаления</b></p>
2. <p><b>простота операций </b></p>
3. <p><b>наличие только одной связи снижает надёжность хранения данных </b></p>
4. <p><b>меньший расход памяти</b></p>

Ответ: 1, 3

Комментарий: SD001018

**<p><b>Значение, которое содержит в себе нулевой показатель в С++, это: </b></p>**

1. <p><b>NIL</b></p>
2. <p><b>NUL</b></p>
3. <p><b>0</b></p>
4. <p><b>NULL</b></p>

Ответ: 3, 4

Комментарий: SD001019

**<p><b>Пример структуры, описывающей узел связанного списка: </b></p>**

1. <p><b>struct Node <br />

{<br />

int data ; <br />

Node \*next; <br />

};</b></p>

1. <p><b>class Node<br />

{<br />

public:<br />

int firstNode;<br />

int secondNode;<br />

int thirdNode;<br />

};</b></p>

1. <p><b>struct NodeStruct<br />

{<br />

int first;<br />

int second;<br />

int third;<br />

};</b></p>

Ответ: 1

Комментарий: SD001020

**<p><b>На чем построен стек? </b></p>**

1. <p><b>На массивах</b></p>
2. <p><b>На классах</b></p>
3. <p><b>На связных списках</b></p>
4. <p><b>На циклах</b></p>

Ответ: 3

Комментарий: SD001021

**<p><b>Правда ли, что элементы стека извлекаются из него в обратном порядке тому, в котором они добавлялись? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001022

**<p><b>Сокращенное наименование принципа работы стека: </b></p>**

1. <p><b>FLIP</b></p>
2. <p><b>LIFO</b></p>
3. <p><b>LOFI</b></p>
4. <p><b>FILO</b></p>

Ответ: 2

Комментарий: SD001023

**<p><b>Каким образом можно обратиться к элементы, находящемуся в середина стека?: </b></p>**

1. <p><b>по его собственному номеру</b></p>
2. <p><b>по его номеру относительно последнего занесенного элемента</b></p>
3. <p><b>по его номеру относительно первого занесенного элемента</b></p>
4. <p><b>Нет правильного ответа</b></p>

Ответ: 4

Комментарий: SD001024

**<p><b>Преимущества стека, реализованного с помощью односвязного списка: </b></p>**

1. <p><b>добавление элемента всегда занимает одно и то же время</b></p>
2. <p><b>элементы могут располагаться в памяти разреженно </b></p>
3. <p><b>возможность перемещения по стеку лишь в одном направлении</b></p>

Ответ: 1

Комментарий: SD001025

**<p><b> Сокращенное наименование принципа работы очереди: </b></p>**

1. <p><b>LOLO</b></p>
2. <p><b>LOLI</b></p>
3. <p><b>LILO</b></p>
4. <p><b>QUEUE</b></p>

Ответ: 3

Комментарий: SD001026

**<p><b>Виды очередей: </b></p>**

1. <p><b>Простая очередь</b></p>
2. <p><b>Смешанная очередь</b></p>
3. <p><b>Кольцевая очередь</b></p>
4. <p><b>Очередь с приоритетами</b></p>

Ответ: 1, 3, 4

Комментарий: SD001027

**<p><b>Суть кольцевой очереди: </b></p>**

1. <p><b>Элемент, который выходит из очереди перемещается в её конец</b></p>
2. <p><b>Элемент, которые заходит в очередь перемещается в её конец(реверсия) </b></p>
3. <p><b>После обращения к последнему элементу, вы обращаетесь снова к первому</b></p>
4. <p><b>После обращения к первому элементу он меняется с последний, второй с предпоследним и т.д. </b></p>

Ответ: 1

Комментарий: SD001028

**<p><b>Примеры кольцевой очереди на практике: </b></p>**

1. <p><b>Очередь у кассы в магазине</b></p>
2. <p><b>Движение трамвая по кольцевому маршруту</b></p>
3. <p><b>Очередь событий для их отработки в Windows</b></p>
4. <p><b>Принцип зарядки и разрядки обоймы пистолета</b></p>

Ответ: 2, 3

Комментарий: SD001029

**<p><b>Какой элемент выйдет последним из очереди с приоритетами? </b></p>**

1. <p><b>Тот, что был добавлен первым</b></p>
2. <p><b>Тот, что имеет наивысший приоритет</b></p>
3. <p><b>Тот, что был добавлен последним</b></p>
4. <p><b>Тот, что имеет наименьший приоритет</b></p>

Ответ: 4

Комментарий: SD001030

**<p><b>Что необходимо указывать при объявлении структуры дерева? <p><b>**

1. <p><b>потомки </b></p>
2. <p><b>число ветвей</b></p>
3. <p><b>тип данных потомков</b></p>
4. <p><b>указатель на предка</b></p>

Ответ: 1

Комментарий: SD001031

**<p><b>Способ преобразования блока данных в число – это</b></p>**

1. <p><b>Индексирование</b></p>
2. <p><b>Хеширование</b></p>
3. <p><b>Маркировка</b></p>
4. <p><b>Сортировка</b></p>

Ответ: 2

Комментарий: SD001032

**<p><b>Как называется узел дерева, не имеющий потомков? </b></p>**

1. <p><b>последний </b></p>
2. <p><b>терминальный </b></p>
3. <p><b>окончательный </b></p>
4. <p><b>финальный</b></p>

Ответ: 2

Комментарий: SD001033

**<p><b>У всех узлов левого поддерева произвольного узла X бинарного дерева поиска значения ключей данных ПРОПУСК значения ключа данных самого узла X. Заполните пропуск</b></p>**

1. <p><b>больше </b></p>
2. <p><b>меньше </b></p>
3. <p><b>больше или равно </b></p>
4. <p><b>меньше или равно</b></p>

Ответ: 2

Комментарий: SD001034

**<p><b>Какое максимальное количество узлов-потомков у каждого узла кучи? </b></p>**

1. <p><b>2</b></p>
2. <p><b>8</b></p>
3. <p><b>64</b></p>
4. <p><b>неограниченное</b></p>

Ответ: 4

Комментарий: SD001035

**<p><b>С помощью какого типа данных обычно реализуется куча? </b></p>**

1. <p><b>дерево</b></p>
2. <p><b>класс</b></p>
3. <p><b>динамический массив</b></p>
4. <p><b>стек</b></p>

Ответ: 1

Комментарий: SD001036

**<p><b>Из скольки объектов состоит элемент словаря? </b></p>**

1. <p><b>1</b></p>
2. <p><b>2</b></p>
3. <p><b>4</b></p>
4. <p><b>число неограниченно</b></p>

Ответ: 2

Комментарий: SD001037

**<p><b>Может ли соответствовать одно и то же значение двум разным ключам словаря? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001038

**<p><b>Дано пустое множество mySet. Что выведет данная программа? <br>mySet.insert('I');<br>mySet.insert('n');<br>mySet.insert('f');<br>mySet.insert('i');<br>mySet.insert('n'); <br>copy( mySet.begin(), mySet.end(), ostream\_iterator<char>(cout, " "));</b></p>**

1. <p><b>I n f i n </b></p>
2. <p><b>I f i n</b></p>
3. <p><b>f i n</b></p>
4. <p><b>I f i n n</b></p>

Ответ: 2

Комментарий: SD001039

**<p><b>Может ли множество содержать дубликаты? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD001040

**<p><b>Чему равно количество строк матрицы смежности графа? </b></p>**

1. <p><b>Количеству рёбер графа</b></p>
2. <p><b>Количеству вершин графа</b></p>
3. <p><b>Максимальной степени входа вершины графа</b></p>
4. <p><b>Минимальной степени входа вершины графа</b></p>

Ответ: 2

Комментарий: SD001041

**<p><b>В чём отличие очереди от дека? </b></p>**

1. <p><b>В очередь можно только добавлять элементы</b></p>
2. <p><b>В очереди с каждого конца можно либо только удалять, либо только добавлять элементы</b></p>
3. <p><b>В дек можно добавлять/удалять элементы только с одного конца</b></p>
4. <p><b>Они предназначены для разных типов данных</b></p>

Ответ: 2

Комментарий: SD001042

**<p><b>Почему сложных типов данных нет в стандартных библиотеках? </b></p>**

1. <p><b>они занимают слишком много места, это усложняет подключение</b></p>
2. <p><b>нет универсального решения, проще написать самому</b></p>
3. <p><b>их реализация требует подключения дополнительных библиотек с необходимыми функциями</b></p>
4. <p><b>нет единого мнения, как они должны быть реализованы</b></p>

Ответ: 2

Комментарий: SD001043

**<p><b>Тип данных, который удобно использовать при необходимости добавлять и удалять элементы в середине</b></p>**

1. <p><b>дек</b></p>
2. <p><b>одномерный массив</b></p>
3. <p><b>связанный список</b></p>
4. <p><b>очередь</b></p>

Ответ: 3

Комментарий: SD001044

**<p><b>Какой тип данных самый универсальный? </b></p>**

1. <p><b>одномерный массив</b></p>
2. <p><b>граф</b></p>
3. <p><b>динамический массив</b></p>
4. <p><b>куча</b></p>

Ответ: 3

Комментарий: SD001045

**<p><b>С помощью какого типа данных лучше всего решать численные задачи типа «Ханойская башня»? </b></p>**

1. <p><b>очередь</b></p>
2. <p><b>стек</b></p>
3. <p><b>динамический массив</b></p>
4. <p><b>дек</b></p>

Ответ: 2

Комментарий: SD001046

**<p><b>Возможно ли реализовать кучу при помощи одномерного массива? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001047

**<p><b>Пустой указатель – это</b></p>**

1. <p><b>NULL</b></p>
2. <p><b>nullptr</b></p>
3. <p><b>null</b></p>
4. <p><b>\_ptr</b></p>

Ответ: 2

Комментарий: SD001048

**<p><b>Как в памяти хранится связный список? </b></p>**

1. <p><b>последовательно</b></p>
2. <p><b>распределённо</b></p>

Ответ: 2

Комментарий: SD001049

**<p><b>Укажите неверное утверждение для полного бинарного дерева</b></p>**

1. <p><b>у каждой вершины не более двух потомков</b></p>
2. <p><b>заполнение вершин идёт сверху вниз</b></p>
3. <p><b>в пределах одного уровня заполнение идёт справа налево</b></p>
4. <p><b>является кучей</b></p>

Ответ: 3

Комментарий: SD001050

**<p><b>Какой индекс будет иметь левый потомок вершины i двоичной кучи, хранимой в виде одномерного массива? </b></p>**

1. <p><b>i + 1</b></p>
2. <p><b>i + 2</b></p>
3. <p><b>2 \* i + 1</b></p>
4. <p><b>2 \* i + 2</b></p>

Ответ: 3

Комментарий: SD001051

**<p><b>Где может храниться самый большой элемент двоичной кучи? </b></p>**

1. <p><b>на левом краю нижнего ряда</b></p>
2. <p><b>на правом краю нижнего ряда</b></p>
3. <p><b>на верхнем ряду</b></p>
4. <p><b>в любом месте</b></p>

Ответ: 3

Комментарий: SD001052

**<p><b>При добавлении в какую часть динамического массива он показывает наилучшую производительность? </b></p>**

1. <p><b>начало</b></p>
2. <p><b>середина</b></p>
3. <p><b>конец</b></p>
4. <p><b>не имеет значения</b></p>

Ответ: 3

Комментарий: SD001053

**<p><b>Что делает данная программа?<br>while (ptr\_p) {<br>&ensp;cout << p->val << " ";<br>&ensp;p = p->next; <br>}</b></p>**

1. <p><b>Выводит количество элементов списка</b></p>
2. <p><b>Проверяет, нет ли в списке пустых значений</b></p>
3. <p><b>Выводит на экран весь список</b></p>
4. <p><b>Очищает список</b></p>

Ответ: 3

Комментарий: SD001054

**<p><b>Структура FIFO – это…</b></p>**

1. <p><b>очередь</b></p>
2. <p><b>стек</b></p>
3. <p><b>дек</b></p>
4. <p><b>динамический массив</b></p>

Ответ: 1

Комментарий: SD001055

**<p><b>Как называется граф, каждой вершине которого поставлено в соответствие конкретное числовое значение</b></p>**

1. <p><b>Числовой</b></p>
2. <p><b>Определённый</b></p>
3. <p><b>Взвешенный</b></p>
4. <p><b>Нормальный</b></p>

Ответ: 3

Комментарий: SD001056

**<p><b>Как называется граф, каждой вершине которого поставлено в соответствие конкретное числовое значение</b></p>**

1. <p><b>Матрица смежности</b></p>
2. <p><b>Матрица инцидентности</b></p>
3. <p><b>Дерево</b></p>
4. <p><b>Список ребер</b></p>

Ответ: 1, 2, 4

Комментарий: SD001057

**<p><b>Какое минимальное количество путей между каждой парой вершин существует в связном графе?</b></p>**

1. <p><b>0</b></p>
2. <p><b>1</b></p>
3. <p><b>2</b></p>
4. <p><b>3</b></p>

Ответ: 2

Комментарий: SD001058

**<p><b>Можно ли применять алгоритм поиска в ширину для неориентированного графа?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD001059

**<p><b>Какой тип данных эффективнее всего использовать для реализации поиска в ширину в графе?</b></p>**

1. <p><b>Стек</b></p>
2. <p><b>Дерево</b></p>
3. <p><b>Динамический массив</b></p>
4. <p><b>Очередь</b></p>

Ответ: 4

Комментарий: SD001060

02\_C++. Массивы строк типа string

**<p><b>Сколько способов выделения памяти для строк существует? </b></p>**

1. <p><b>1</b></p>
2. <p><b>2</b></p>
3. <p><b>3</b></p>
4. <p><b>4</b></p>

Ответ: 2

Комментарий: SD002001

**<p><b>Что выведет данная программа?<br>string Numbers[] { "Один", "Два", "Три" };<br>cout <&lt;"\nArray of Numbers:"<&lt; endl;<br>for (int i = 0; i &lt; 4; i++)<br>cout <&lt; Numbers[i] <&lt; endl; <br></b></p>**

1. <p><b>Один Два Три</b></p>
2. <p><b>Один<br>

     Два<br>

     Три<br></b></p>

1. <p><b>Три<br>

    Два<br>

    Один<br></b></p>

1. <p><b>Ошибку сборки</b></p>

Ответ: 2

Комментарий: SD002002

**<p><b>Что выведет данная программа?<br>for (int i = 0; i &lt; 7; i++)<br>AS[i] = (char)('1' + i); <br>cout &lt;&lt; "\nArray AS:" &lt;&lt; ::endl; <br>for (int i = 0; i &lt; n; i++)<br>cout &lt;&lt; "AS[" << i << "] = " &lt;&lt; AS[i].c\_str() &lt;&lt; “ ”; <br></b></p>**

1. <p><b>01234567</b></p>
2. <p><b>12345678</b></p>
3. <p><b>1 2 3 4 5 6 7 8, но по одному символу на строку</b></p>
4. <p><b>Ошибку сборки</b></p>

Ответ: 1

Комментарий: SD002003

**<p><b>Как выделяется память под массив строк в данной программе?<br>AS = new string[n]; <br>for (int i = 0; i &lt; n; i++)<br>AS[i] = (char)('1' + i); <br>cout <&lt; "\nArray AS:" <&lt; endl; <br>for (int i = 0; i &lt; n; i++)<br>cout << "AS[" << i << "] = " << AS[i].c\_str() << endl; <br>delete[] AS; <br></b></p>**

1. <p><b>статически</b></p>
2. <p><b>динамически</b></p>

Ответ: 2

Комментарий: SD002004

**<p><b>Сколько времени будет выполняться данная программа?<br>while (s != "");<br>cout <&lt; "\nArray AS is as follows:\n";<br>for (int i = 0; i &lt; count; i++)<br>cout <&lt; "AS[" << i << "] = " <&lt; AS[i] <&lt; endl;<br></b></p>**

1. <p><b>пока не будет нажат Enter</b></p>
2. <p><b>пока не будет введена пустая строка</b></p>
3. <p><b>бесконечно</b></p>
4. <p><b>программа не запустится</b></p>

Ответ: 2

Комментарий: SD002005

**<p><b>Возможна ли работа с типом string без подключения модуля <string>?</b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>

Ответ: 1

Комментарий: SD002006

**<p><b>Сколько элементов будет храниться в данном массиве?<br>string Numbers[]{ "One", "Two", "Three" };</b></p>**

1. <p><b>нисколько</b></p>
2. <p><b>4</b></p>
3. <p><b>неопределенное кол-во</b></p>
4. <p><b>3</b></p>

Ответ: 2

Комментарий: SD002007

**<p><b>Что будет являтся результатом вывода данного участка кода?<br>string family[3] = { 'son', 'mam', 'dad' };<br/ >for (int i = 0; i &lt; 3; i++)<br/ >{<br/ >&ensp;cout<&lt;family[i]<&lt;" "; <br/ >}<br/ ></b></p>**

1. <p><b>son mam dad</b></p>
2. <p><b>dad mam son</b></p>
3. <p><b>man son dad</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 4

Комментарий: SD002008

**<p><b>Можно ли выделять память как динамически, так и статически для массива строк?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD002009

**<p><b> Что будет являтся результатом вывода данного участка кода?<br>string s4(5, 'd');<br/ >cout &lt;** **&lt; s4; </b></p>**

1. <p><b>ddddd</b></p>
2. <p><b>dddd</b></p>
3. <p><b>ошибка</b></p>

Ответ: 1

Комментарий: SD002010

**<p><b>Для работы с типом string нужно подключать заголовочный файл?</b></p>**

1. Да
2. Нет

Ответ: 1

Комментарий: SD002011

**<p><b>Как можно выделять память для массива строк?</b></p>**

1. <p><b>только статически</b></p>
2. <p><b>только динамически</b></p>
3. <p><b>статически и динамически</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 3

Комментарий: SD002012

**<p><b>Что будет выведено в консоль при запуске программы:<br>const int quantity = 3;<br/>string fruits[quantity] = { "apple", "banana", "grape" }; <br/>for (int i = 0; i &lt; quantity; i++) <br/>{<br/>&ensp;std::cout &lt;&lt; "Fruits " &lt;&lt; i &lt;&lt; " = " &lt;&lt; fruits[i]; <br/>}<br/></b></p>**

1. <p><b>Fruits 1 = apple        Fruits 2 = banana       Fruits 3 = grape</b></p>
2. <p><b>Fruits 0 = apple        Fruits 1 = banana       Fruits 2 = grape</b></p>
3. <p><b>ошибка компиляции</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 2

Комментарий: SD002013

**<p><b>Что будет выведено в консоль при запуске программы: <br/>string Numbers[]{"One", "Two", "Three" }; <br/>for (int i = 0; i &lt; 3; i++) <br/>{<br/>&ensp;std::cout &lt; &lt; Numbers[i] &lt; &lt; "\t" ; <br/>}</b></p>**

1. <p><b>1 2 3</b></p>
2. <p><b>0 1 2</b></p>
3. <p><b>One Two Three</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 3

Комментарий: SD002014

**<p><b>Что будет выведено в консоль при запуске программы: <br/>string\* AS; </br>int n=5; </br>AS = new string[n]; </br>for (int i = 0; i &lt; n; i++) </br>{</br>&ensp;AS[i] = (char)('1' + i); </br>}</br>for (int i = 0; i < n; i++) </br>{</br>&ensp;cout &lt; &lt; "AS["&lt; &lt; i &lt; &lt;"] = " &lt; &lt; AS[i].c\_str()&lt; &lt; "\t";</br>}</br>delete[] AS; </b></p>**

1. <p><b>AS[0] = 1       AS[1] = 2       AS[2] = 3</b></p>
2. <p><b>AS[0] = 0       AS[1] = 1      AS[2] = 2</b></p>
3. <p><b>ошибка компиляции</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 1

Комментарий: SD002015

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br/>std::string str1("abcdef");<br/>std::string str2(str1, 3, 5);<br/>std::cout &lt;&lt; str2;</b></p>**

1. <p><b>f</b></p>
2. <p><b>ef</b></p>
3. <p><b>def</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 3

Комментарий: SD002016

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br/>std::string s{ "Hello!" };<br/>std::cout &lt;&lt; s.size() &lt;&lt; s.length();</b></p>**

1. <p><b>66</b></p>
2. <p><b>67</b></p>
3. <p><b>77</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 1

Комментарий: SD002017

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br/>std::string s = "1110111";<br/>s.erase(3);<br/>std::cout &lt;&lt; s;</b></p>**

1. <p><b>111111</b></p>
2. <p><b>110111</b></p>
3. <p><b>1110</b></p>
4. <p><b>111</b></p>

Ответ: 4

Комментарий: SD002018

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br/>std::string s = "abcdef";<br/>int pos = s.find('g');<br/>std::cout &lt;&lt; pos;</b></p>**

1. <p><b>0</b></p>
2. <p><b>-1</b></p>
3. <p><b>Ошибка компиляции.</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 2

Комментарий: SD002019

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br/>std::string str;<br/>if (str.clear()) std::cout &lt;&lt; "Str is clear.";</b></p>**

1. <p><b>Ничего не выведется</b></p>
2. <p><b>Str is clear.</b></p>
3. <p><b>Ошибка компиляции.</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 3

Комментарий: SD002020

**<p><b>Со строками можно выполнять следующие операции:</b></p>**

1. <p><b> += </b></p>
2. <p><b> =  ,==, != </b></p>
3. <p><b> + </b></p>
4. <p><b> &lt; , &gt; , &lt;= , &gt;= </b></p>

Ответ: 1, 2, 3, 4

Комментарий: SD002021

**<p><b>Какой  метод очищает всю строку от содержимого?</b></p>**

1. <p><b>clear()</b></p>
2. <p><b>empty()</b></p>
3. <p><b>erase()</b></p>
4. <p><b>resize()</b></p>

Ответ: 1

Комментарий: SD002022

**<p><b>Какие библиотеки следует подключить, чтобы использовать возможности string?</b></p>**

1. <p><b>stdlib</b></p>
2. <p><b>iostream</b></p>
3. <p><b>string </b></p>
4. <p><b>algorithm</b></p>

Ответ: 3

Комментарий: SD002023

**<p><b>На что заканчивается строка?</b></p>**

1. <p><b>/0</b></p>
2. <p><b>\0</b></p>
3. <p><b>0</b></p>
4. <p><b>endl</b></p>

Ответ: 2

Комментарий: SD002024

**<p><b>Выберите правильные способы задания строк:</b></p>**

1. <p><b>С помощью cin</b></p>
2. <p><b>С помощью string(‘value’) </b></p>
3. <p><b>С помощью getline</b></p>
4. <p><b>С помощью бога</b></p>

Ответ: 1, 3

Комментарий: SD002025

**<p><b>Результат программы:<br>string s1 = "hello";string s2 = "4321";s1.insert(3, s2);</b></p>**

1. <p><b>4hello321</b></p>
2. <p><b>hel4321lo</b></p>
3. <p><b>he4321llo</b></p>
4. <p><b>43hello21</b></p>

Ответ: 2

Комментарий: SD002026

**<p><b>Результат программы:<br>string s1 = "hello";<br>string s2 = "4321";<br>int res;<br>res=s1.compare(s2);</b></p>**

1. <p><b>-1</b></p>
2. <p><b>0</b></p>
3. <p><b>1</b></p>
4. <p><b>Затрудняюсь ответить</b></p>

Ответ:3

Комментарий: SD002027

**<p><b>Сколько бит памяти занимает строка СТРУКТУРА ДАННЫХ?</b></p>**

1. <p><b>128</b></p>
2. <p><b>120</b></p>
3. <p><b>15</b></p>
4. <p><b>16</b></p>

Ответ: 1

Комментарий: SD002028

**<p><b>Оператор &gt; &gt; это?</b></p>**

1. <p><b>Функция шаблона, проверяющая, что одна строка намного больше другой</b></p>
2. <p><b>Функция шаблона, выполняющая сдвиг одной строки, и вставляя другую</b></p>
3. <p><b>Ввод новой строки</b></p>
4. <p><b>Функция шаблона, извлекающая строку из входного потока</b></p>

Ответ: 4

Комментарий: SD002029

**<p><b>Содержит ли string конструкторы?</b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>
3. <p><b>Лего? </b></p>

Ответ: 2

Комментарий: SD002030

**<p><b> Какую библиотеку можно подключать при работе  со строками</b></p>**

1. <p><b> <sentence></b></p>
2. <p><b> <widechar></b></p>
3. <p><b> <strung></b></p>
4. <p><b> нет правильного ответа </b></p>

Ответ: 4

Комментарий: SD002031

**<p><b> Выберите верные способы инициализации массива строк </b></p>**

1. <p><b> Инициализация массива строк указанием размера массива</b></p>
2. <p><b> Через служебное слово define</b></p>
3. <p><b> Инициализация без указания размера массива </b></p>
4. <p><b> Все варианты верны</b></p>

Ответ: 1,3

Комментарий: SD002032

**<p><b> Команда delete[]возвращает значение последнего, удалённого <br> элемента массива </b></p>**

1. <p><b> Да</b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD002033

**<p><b> При создании динамического массива строк на 7 элементов, индекс последнего будет 6, <br> под ним будет храниться символ  /0 </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD002034

**<p><b> При методе сортировки вставками на каждом шаге сортировки сравнивается <br> текущий элемент со всеми элементами в неотсортированной части.  </b></p>**

1. <p><b> Это верно </b></p>
2. <p><b> Нет это не верно, текущий элемент сравнивается с отсортированной частью <br> массива </b></p>
3. <p><b> Нет это не верно, текущий элемент сравнивается со всеми  элементами <br> массива</b></p>
4. <p><b> Такого метода не существует </b></p>

Ответ: 2

Комментарий: SD002035

**<p><b>Каким образом можно ввести через консоль строку (str – название строки), содержащую пробелы?</b></p>**

1. <p<b>cin >> str;</b></p>
2. <p<b>ifstream(str);</b></p>
3. <p<b>ofstream(str);</b></p>
4. <p<b>getline(cin, str);</b></p>

Ответ: 4

Комментарий: SD002036

**<p><b>Узнать длину строки (str – название строки) можно при помощи:</b></p>**

1. <p<b>lenth(str);</b></p>
2. <p<b>length(str);</b></p>
3. <p<b>str.length();</b></p>
4. <p<b>length(str);</b></p>

Ответ: 3

Комментарий: SD002037

**<p><b>Результатом выполнения программы будет:<br />string x = "10";<br />string y = "20";<br />string z = x + y;<br />cout &lt;&lt;  z;</b></p>**

1. <p<b>1020</b></p>
2. <p<b>30</b></p>
3. <p<b>120</b></p>
4. <p<b>Выдаст ошибку</b></p>

Ответ: 1

Комментарий: SD002038

**<p><b>Результатом выполнения программы будет:<br />string x = "10";<br />int y = 20;<br />string z = x + y;<br />cout &lt;&lt; z;</b></p>**

1. <p<b>30</b></p>
2. <p<b>1020</b></p>
3. <p<b>10 20</b></p>
4. <p<b>Выдаст ошибку</b></p>

Ответ: 4

Комментарий: SD002039

**<p><b>Каким образом можно обратиться к первому элементу строки(str – название строки)?**

1. <p<b>str[1];</b></p>
2. <p<b>str[0];</b></p>
3. <p<b>\*str.begin();</b></p>
4. <p<b>\*begin(str);</b></p>

Ответ: 2, 3

Комментарий: SD002040

03\_Перечисление. Объединение. STL контейнеры. Массив. Дек. Однодвух связный список. Вектор.

**<p><b>Перечислители могут быть отрицательными?</b></p>**

1. <p><b>нет</b></p>
2. <p><b>да</b></p>

Ответ: 2

Комментарий: SD003001

**<p><b>Перечислителям можно присваивать значения типа с плавающей точкой?</b></p>**

1. <p><b>да </b></p>
2. <p><b>нет</b></p>

Ответ: 2

Комментарий: SD003002

**<p><b>Перечислителям можно не присваивать значения?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 1

Комментарий: SD003003

**<p><b>Что будет выведено в консоль при запуске программы: <br> enum temperature {&ensp;minsk, vitebsk = -16, grodno, gomel = grodno + 10 };<br>&ensp;std::cout <&lt minsk <&lt " " <&lt vitebsk <&lt " " <&lt grodno <&lt " " <&lt gomel;</b></p>**

1. <p><b>1 -16 -15 -5</b></p>
2. <p><b>0 -16 0 10</b></p>
3. <p><b>0 -16 -15 -5</b></p>
4. <p><b>0 -16 -17 -7</b></p>

Ответ: 3

Комментарий: SD003004

**<p><b>Что будет выведено в консоль при запуске программы:<br>enum fruits { apple, orange, grape, pear };<br>int main()<br>{<br> &ensp;int grape = 10;<br>&ensp;std::cout <&lt apple <&lt " " <&lt orange <&lt " " <&lt grape <&lt " " <&lt pear;<br>}</b></p>**

1. <p><b>0 1 10 3</b></p>
2. <p><b>0 1 10 11</b></p>
3. <p><b>0 1 2 3</b></p>
4. <p><b>ошибка компиляции</b></p>

Ответ: 1

Комментарий: SD003005

**<p><b>Что будет выведено в консоль при запуске программы:<br>enum massa{ elephant = 50, lion = 16.9, bear};<br>std::cout <&lt elephant <&lt " " <&lt lion <&lt " " <&lt bear; </b></p>**

1. <p><b>50 16.9 17.9</b></p>
2. <p><b>50 16.9 17</b></p>
3. <p><b>ошибка компиляции</b></p>
4. <p><b>50 16.9 2</b></p>

Ответ: 3

Комментарий: SD003006

**<p><b>Объединение может содержать битовые поля?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 2

Комментарий: SD003007

**<p><b>Объединение может содержать виртуальные методы, конструкторы, деструкторы и операцию присваивания?</b></p>**

1. <p><b>нет</b></p>
2. <p><b>да</b></p>

Ответ: 1

Комментарий: SD003008

**<p><b>Объединение может входить в иерархию классов?</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 2

Комментарий: SD003009

**<p><b>Является ли данное утверждение верным: объединение может инициализироваться только значением его первого элемента.</b></p>**

1. <p><b>да</b></p>
2. <p><b>нет</b></p>

Ответ: 1

Комментарий: SD003010

**<p><b>Что будет выведено в консоль при запуске программы: <br>** **union myunion <br> {<br>&ensp;int x, y; <br>}; <br>myunion a; <br>a.x = 30;<br>a.y = 40; <br>printf("%d\t", a.x); <br> printf("%d", a.y); </b></p>**

1. <p><b>30  40</b></p>
2. <p><b>40  30</b></p>
3. <p><b>30  30</b></p>
4. <p><b>40  40</b></p>

Ответ: 4

Комментарий: SD003011

**<p><b>Какое из утверждений является ложным:<br>1. анонимное объединение -- объединение, у которого нет имени. <br>2.  с помощью анонимного объединения программа может сэкономить память.</b></p>**

1. <p><b>2</b></p>
2. <p><b>1</b></p>
3. <p><b>оба ложные</b></p>
4. <p><b>оба верные</b></p>

Ответ: 4

Комментарий: SD003012

**<p><b>Что будет выведено в консоль при запуске программы: <br>union { <br>&ensp; short int f; <br>&ensp;char ch[2]; <br>}; <br>ch[0] = 'X'; <br>ch[1] = 'Y'; <br>std::cout <&lt ch[0] <&lt " " <&lt ch[1] <&lt " " <&lt f; </b></p>**

1. <p><b>X Y 0</b></p>
2. <p><b>X Y 32767</b></p>
3. <p><b>X Y 22872</b></p>
4. <p><b>Ошибка компиляции</b></p>

Ответ: 3

Комментарий: SD003013

**<p><b>Динамический массив – это:</b></p>**

1. <p><b>набор переменных, имеющих одно и то же базовое имя и отличающихся одна от другой числовым признаком.</b></p>
2. <p><b>массив переменной длины, память под который выделяется в процессе выполнения программы.</b></p>
3. <p><b>массив фиксированной длины, память под который выделяется в процессе выполнения программы.</b></p>
4. <p><b>массив фиксированной длины, память под который выделяется в процессе его объявления. </b></p>

Ответ: 2

Комментарий: SD003014

**<p><b>Выделение динамической памяти может осуществляться при помощи:</b></p>**

1. <p><b>только malloc</b></p>
2. <p><b>только malloc и new</b></p>
3. <p><b>только new</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 4

Комментарий: SD003015

**<p><b>Для чего используется функция malloc?</b></p>**

1. <p><b>выделяет блок памяти и возвращает указатель на начало блока</b></p>
2. <p><b>возвращает указатель на выделенный участок или NULL при невозможности выделить память</b></p>
3. <p><b>освобождает выделенную память</b></p>
4. <p><b>изменяет размер выделенной ранее памяти</b></p>

Ответ: 2

Комментарий: SD003016

**<p><b>Синтаксис array:</b></p>**

1. <p><b>array {тип\_данных, кол-во\_элементов} имя={элементы};</b></p>
2. <p><b>array [тип\_данных, кол-во\_элементов] имя={элементы};</b></p>
3. <p><b>array <тип\_данных, кол-во\_элементов> имя={элементы};</b></p>
4. <p><b>array <тип\_данных: кол-во\_элементов> имя={элементы};</b></p>

Ответ: 3

Комментарий: SD003017

**<p><b>Что будет выведено в консоль при запуске программы: <br> array &lt;int, 5&gt; myarray; <br>myarray = { 1,2,3,4,5 }; <br>myarray[1] = myarray[3]; <br>myarray[4] = myarray[1]; <br>&ensp;for (auto el : myarray)<br>{ <br>&ensp;std::cout <&lt el;<br>}</b></p>**

1. <p><b>32335</b></p>
2. <p><b>14344</b></p>
3. <p><b>14345</b></p>
4. <p><b>ошибка компиляции</b></p>

Ответ: 2

Комментарий: SD003018

**<p><b>Deque – это структура данных, представляющая из себя список элементов, в которой:</b></p>**

1. <p><b>можно добавлять новые и удалять существующие элементы</b></p>
2. <p><b>можно только удалять существующие элементы</b></p>
3. <p><b>можно только добавлять новые элементы</b></p>
4. <p><b>можно добавлять новые и удалять существующие элементы с обоих концов</b></p>

Ответ: 4

Комментарий: SD003019

**<p><b>Синтаксис deque:</b></p>**

1. <p><b>deque <тип\_данных> имя={элементы};</b></p>
2. <p><b>deque имя={элементы} <тип\_данных>;</b></p>
3. <p><b>deque {тип\_данных} имя=[элементы]; </b></p>
4. <p><b>deque [тип\_данных] имя=<элементы>;</b></p>

Ответ: 1

Комментарий: SD003020

**<p><b>Что будет выведено в консоль при запуске программы: <br>deque&lt;int&gt; mydeque(5, 2); <br>for (auto el : mydeque) <br>{<br> &ensp;std::cout <&lt el; <br>} <br>} </b></p>**

1. <p><b>52</b></p>
2. <p><b>55555</b></p>
3. <p><b>22222</b></p>
4. <p><b>ошибка компиляции</b></p>

Ответ: 3

Комментарий: SD003021

**<p><b>Для получения элементов очереди можно использовать:</b></p>**

1. <p><b> [index] и at(index) </b></p>
2. <p><b> [index], at(index), front() и back()</b></p>
3. <p><b>front() и back()</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 2

Комментарий: SD003022

**<p><b>** **При обращении по некорректному индексу с помощью at(index):</b></p>**

1. <p><b>результат будет неопределенным</b></p>
2. <p><b>генерируется исключение out\_of\_range</b></p>
3. <p><b>ошибка компиляции</b></p>
4. <p><b>нет правильного ответа</b></p>

Ответ: 2

Комментарий: SD003023

**<p><b>Для определения размера очереди используется:</b></p>**

1. <p><b>нет правильного ответа</b></p>
2. <p><b>resize()</b></p>
3. <p><b>empty()</b></p>
4. <p><b>size()</b></p>

Ответ: 4

Комментарий: SD003024

**<p><b>Что будет выведено в консоль при запуске программы:<br>deque&lt;int&gt; numbers = { 1, 2, 3, 4, 5, 6 }; <br>numbers.resize(7, 8);<br>for (auto el: numbers) <br>{<br>&ensp;std::cout <&lt el;<br>}</b></p>**

1. <p><b>ошибка компиляции</b></p>
2. <p><b>12345678</b></p>
3. <p><b>1234567</b></p>
4. <p><b>1234568</b></p>

Ответ: 4

Комментарий: SD003025

**<p><b>Что будет выведено в консоль при запуске программы:<br>deque&lt;int&gt; deque1 = { 1, 2, 3, 4, 5 }; <br>deque&lt;int&gt; deque2 = { 6, 7, 8, 9 }; <br> deque&lt;int&gt; deque3 = { 10, 11, 12, 13 };<br>deque1.swap(deque2); <br> deque3.swap(deque1);<br>for (auto el : deque3)<br>{<br>&ensp;std::cout <&lt el; <br>}<br></b></p>**

1. <p><b>ошибка компиляции</b></p>
2. <p><b>1, 2, 3, 4, 5</b></p>
3. <p><b>10, 11, 12, 13</b></p>
4. <p><b>6, 7, 8, 9</b></p>

Ответ: 4

Комментарий: SD003026

**<p><b>Что будет выведено в консоль при запуске программы: <br>deque&lt;int&gt; numbers = { 1, 2, 3, 4, 5 }; <br>auto iter2 = ++numbers.cbegin(); <br> numbers.insert(iter2, 3, 4);<br>auto iter3 = --numbers.cend();<br>numbers.insert(iter3, 3, 4); <br>for (auto el : numbers)<br>{<br>&ensp;std::cout <&lt el; <br>}</b></p>**

1. <p><b>13423345</b></p>
2. <p><b>14442344445</b></p>
3. <p><b>44412345444</b></p>
4. <p><b>341234534</b></p>

Ответ: 2

Комментарий: SD003027

**<p><b>Для удаления элементов из очереди используются:</b></p>**

1. <p><b>нет правильного ответа</b></p>
2. <p><b>только clear()</b></p>
3. <p><b>только pop\_back() и pop\_front()</b></p>
4. <p><b>только clear(p), pop\_back() и pop\_front()</b></p>

Ответ: 1

Комментарий: SD003028

**<p><b>Для чего используется функция erase():</b></p>**

1. <p><b>удаляет последний элемент</b></p>
2. <p><b>удаляет все элементы</b></p>
3. <p><b>удаляет элемент, на который указывает итератор</b></p>
4. <p><b>удаляет первый элемент</b></p>

Ответ: 3

Комментарий: SD003029

**<p><b>Что будет выведено в консоль при запуске программы:<br>deque&lt;int&gt; mydeque = { 5,6,7,8,9,10 };<br>auto begin = mydeque.begin();<br>auto end = mydeque.end();<br>mydeque.erase(begin, --end);<br>for (auto el : mydeque) <br>{<br>&ensp;std::cout <&lt el; <br>}</b></p>**

1. <p><b>510</b></p>
2. <p><b>10</b></p>
3. <p><b>5</b></p>
4. <p><b>910</b></p>

Ответ: 1

Комментарий: SD003030

**<p><b>Какие виды связных списков чаще всего используются в С:</b></p>**

1. <p><b>Односвязные и многосвязные</p></b>
2. <p><b>Связные и несвязные</b></p>
3. <p><b>Односвязные и двусвязные</b></p>
4. <p><b>Связной – это магазин сотовой связи. Это всё, что я знаю.</b></p>

Ответ: 3

Комментарий: SD003031

**<p><b>forward\_list  представляет собой:</b></p>**

1. <p><b>Односвязный линейный список</b></p>
2. <p><b>Односвязный циклический список</b></p>
3. <p><b>Двусвязный линейный список</b></p>
4. <p><b>Двусвязный циклический список</b></p>

Ответ: 1

Комментарий: SD003032

**<p><b>Отличие между односвязным линейным и двусвязным линейным списком заключается в:</b></p>**

1. <p><b>Односвязный линейный содержит 1 поле указателя и последний содержит адрес первого узла, а двусвязный линейный содержит 2 поля и каждое из них из последнего узла содержит адрес первого узла и наоборот.</b></p>
2. <p><b>Односвязный линейный содержит 1 поле указателя и последний содержит нулевое значение, а двусвязный линейный содержит 2 поля и каждое из них из последнего узла содержит адрес первого узла и наоборот.</b></p>
3. <p><b>Односвязный линейный содержит 1 поле указателя и последний содержит адрес первого узла, а двусвязный линейный содержит 2 поля и первый и последний узлы содержат нулевое значение.</b></p>
4. 4.<p><b> Односвязный линейный содержит 1 поле указателя и последний содержит нулевое значение, а двусвязный линейный содержит 2 поля и первый и последний узлы содержат нулевое значение.</p></b>

Ответ: 4

Комментарий: SD003033

**<p><b>Необходимо ли подключать директиву препроцессора для его создания:</b></p>**

1. <p><b>Нет.</b></p>
2. <p><b>Да.</b></p>
3. <p><b>Затрудняюсь ответить.</b></p>

Ответ: 2

Комментарий: SD003034

**<p><b>std::forward\_list<&lt:int> list(5):</b></p>**

1. <p><b>Создаётся список, состоящий из чисел 1,2,3,4,5.</b></p>
2. <p><b>Создаётся список из 5 элементов, чьё значение равно 5.</b></p>
3. <p><b>Не знаю.</b></p>
4. <p><b>Создаётся список из 5 элементов, чьё значение задаётся по умолчанию.</b></p>

Ответ: 2

Комментарий: SD003035

**<p><b>Можно ли узнать размер списка forward\_list(). Как?</b></p>**

1. <p><b>Да. Это легко реализуется с помощью функции size()</b></p>
2. <p><b>Затрудняюсь ответить.</b></p>
3. <p><b>Нет. Можно узнать только максимальный размер контейнера max\_size()</b></p>

Ответ: 3

Комментарий: SD003036

**<p><b>Обе функции before\_begin() и cbefore\_begin() возвращают итератор на несуществующий элемент списка непосредственно перед его началом:</p></b>**

1. <p><b>Верно</b></p>
2. <p><b>Неверно, только before\_begin() </b></p>
3. <p><b>Неверно, только cbefore\_begin() </b></p>
4. <p><b>Затрудняюсь ответить.</b></p>

Ответ: 1

Комментарий: SD003037

**<p><b>Функция resize(n) позволяет:</b></p>**

1. <p><b>Изменить размер списка до n</b></p>
2. <p><b>Увеличить число элементов списка на n</b></p>
3. <p><b>Оставить n элементов списка</b></p>

Ответ: 3

Комментарий: SD003038

**<p><b>Результат программы:<br>List &lt;int&gt;numbers = { 1,2,3,4,5 };<br>numbers.resize(12);<br>for (int i:numbers)<br> {<br>&ensp;cout <&lt; i <&lt;"."<&lt; endl;<br>}<br></b></p>**

1. <p><b>1.,2.,3.,4.,5.,5.,5.,5.,5.,5.,5.,5.</b></p>
2. <p><b>1.,2.,3.,4.,5.,6.,7.,8.,9.,10.,11.,12. </b></p>
3. <p><b>1.,2.3.,4.,5.,0.,0.,0.,0.,0.,0.,0. </b></p>
4. <p><b>Выдаст ошибку</b></p>

Ответ: 3

Комментарий: SD003039

**<p><b>В функции resize(n, value) итератор value: </b></p>**

1. <p><b>Заменяет все оставшиеся элементы на его значение. </b></p>
2. <p><b>В случае, если n>значения элементов, добавляет элементы со значением value. </b></p><
3. <p><b>Затрудняюсь ответить</b></p>
4. <p><b>Мдааа… Надо было всё-таки на лекции ходить. </b></p>

Ответ: 2

Комментарий: SD003040

**<p><b>Какая функция изменяет значение элементов: </b></p>**

1. <p><b>Swap()</b></p>
2. <p><b>Push\_front()</b></p>
3. <p><b>Pop\_front()</b></p>
4. <p><b>Assign()</b></p>

Ответ: 4

Комментарий: SD003041

**<p><b>Какой из предложенных вариантов assign() позволяет заменить содержимое контейнера элемента элементами инициализации: </b></p>**

1. <p><b>Assign(n, value) </b></p>
2. <p><b>Assign(begin,end) </b></p>
3. <p><b>Затрудняюсь ответить</b></p>
4. <p><b>Assign(il) </b></p>

Ответ: 4

Комментарий: SD003042

**<p><b>Результат программы:<br>forward\_list&lt;int&gt;numbers = { 1, 2, 3, 4, 5 };<br>forward\_list&lt;int&gt; values = { 6, 7, 8, 9, 10, 11 };<br>auto start = ++values.begin();<br>auto end = values.end();<br>numbers.assign(start, end);<br>for (int i : numbers) {<br>cout <&lt; i <&lt; "." <&lt; endl;<br>}<br></b></p>**

1. <p><b>2.,3.,4.,5. </b></p>
2. <p><b>1.,2.,3.,4.,5. </b></p>
3. <p><b>6.,7.,8.,9.,10.,11. </b></p>
4. <p><b>7.,8.,9.,10.,11. </b></p>

Ответ: 4

Комментарий: SD003043

**<p><b>Что делает функция insert\_after(p, n, val): </b></p>**

1. <p><b>Вставка после элемента, на который указывает итератор p n элементов со значением val</b></p>
2. <p><b>Вставка после элемента, на который указывает итератор p набор из другого контейнера от начала до конца, при этом возвращая итератор на последний несуществующий элемент</b></p>
3. <p><b>Вставка списка инициализации после итератора p.</b></p>
4. <p><b>Затрудняюсь ответить</b></p>

Ответ: 1

Комментарий: SD003044

**<p><b>Какая функция выполняет вставку после элемента, на который указывает итератор p набор из другого контейнера от начала до конца, при этом возвращая итератор на последний несуществующий элемент? </b></p>**

1. <p><b>emplace\_front(val) </b></p>
2. <p><b>push\_front(val**)** </b></p>
3. <p><b> insert\_after(p, begin, end) </b></p>
4. <p><b>insert\_after(p, val) </b></p>

Ответ: 3

Комментарий: SD003045

**<p><b>Результат программы:<br>std::forward\_list&lt;int&gt;numbers = { 1, 2 };<br>numbers.push\_front(0);<br>numbers.emplace\_front(-1);<br>auto iter = numbers.begin();<br>iter = numbers.insert\_after(iter, 5, 4);<br>iter = numbers.insert\_after(iter, -3);<br>std::forward\_list&lt;int&gt; values = { 3, 6 };<br>iter = numbers.insert\_after(iter, values.begin(), values.end());<br>numbers.insert\_after(iter, { 7, 8,5 });<br>iter =numbers.emplace\_after(iter,-2); <br>for (int n : numbers)<br>std::cout << n << "\t";<br></b></p>**

1. <p><b> -1 -3 4 4 4 4 4 3 6 -2 -3 7 8 5 0 1 2</b></p>
2. <p><b>-1 4 4 4 4 4 3 -3 6 -2 7 8 5 0 1 2</b></p>
3. <p><b>0 1 2 -1 -3 4 4 4 4 4 3 6 -2 7 8 5</b></p>
4. <p><b>1 2 0 -1 4 4 4 4 4 -3 3 6 7 8 5 -2</b></p>

Ответ: 1

Комментарий: SD003046

**<p><b>Какая функция удаляет все элементы списка: </b></p>**

1. <p><b>erase\_after(p) </b></p>
2. <p><b>pop\_front()</b></p>
3. <p><b>clean()</b></p>
4. <p><b>erase\_after(begin, end) </b></p>

Ответ: 3

Комментарий: SD003047

**<p><b>Какой элемент удалится?<br>std::forward\_list&lt;int&gt; numbers = { 1, 2, 3, 4, 5, 6, 7 };<br>auto iter = numbers.erase\_after(++numbers.begin());<br></b></p>**

1. <p><b> 3</b></p>
2. <p><b>2</b></p>
3. <p><b>Затрудняюсь ответить </b></p>

Ответ: 2

Комментарий: SD003048

**<p><b>Результат программы:<br>std::forward\_list&lt;int&gt; numbers = { 1, 2, 3, 4, 5, 6, 7 };<br>auto iter=numbers.erase\_after(++numbers.begin());<br>numbers.erase\_after(++iter,numbers.end());<br>for (int i : numbers) {<br>cout <&lt; i <&lt; "." <&lt; endl;<br>}<br> </b></p>**

1. <p><b> 1 2 4 5 6</b></p>
2. <p><b>1 2 4 5</b></p>
3. <p><b> 1 2 4 </b></p>
4. <p><b> 2 4 5 6</b></p>

Ответ: 2

Комментарий: SD003049

**<p><b>List является: </b></p>**

1. <p><b>Односвязным линейным список</b></p>
2. <p><b>Односвязным циклическим список</b></p>
3. <p><b>Двусвязным линейным список</b></p>
4. <p><b>Двусвязным циклическим список</b></p>

Ответ: 3

Комментарий: SD003050

**<p><b>Можно ли определить размер списка list? Как? </b></p>**

1. <p><b>Да. Это легко реализуется с помощью функции size()</b></p>
2. <p><b>Затрудняюсь ответить. </b></p>
3. <p><b>Уже ж было, нет? </b></p>
4. <p><b>Нет. Можно узнать только максимальный размер контейнера max\_size()</b></p>

Ответ: 1

Комментарий: SD003051

**<p><b>Результат программы: <br>std::list<int>numbers = { 1,2,3,4,5 };<br>std::list<int> values = { 6, 7, 8, 9, 10, 11 };<br>auto start = ++values.begin(); <br> auto end = values.end();<br>numbers.assign(++start, --end);<br>for (int i : numbers) {<br>cout << i << "." << endl;<br>}</b></p>**

1. <p><b>7 8 9 10</b></p>
2. <p><b>8 9 10</b></p>
3. <p><b>8 9 10 11</b></p>
4. <p><b>7 8 9</b></p>

Ответ: 2

Комментарий: SD003052

**<p><b>Результат программы:<br>std::list<int>numbers = { 1,2,3,4,5 };<br>numbers.assign({ 21, 22, 23, 24, 25 });<br>numbers.assign(2, 3);<br>for (int i : numbers) {<br>cout << i << "." << endl;<br>}</b></p>**

1. <p><b>21 3 3 24 25</b></p>
2. <p><b>21 3 3 22 23 24 25</b></p>
3. <p><b>3 3 </b></p>
4. <p><b>1 3 3 3 5</b></p>

Ответ: 3

Комментарий: SD003053

**<p><b>Стек – это …</b></p>**

1. <p><b>упорядоченный набор элементов, где размещение новых и удаление существующих происходит с одного конца. </b></p>
2. <p><b>набор элементов, где размещение новых и удаление существующих происходит с одного конца. </b></p>
3. <p><b>упорядоченный набор элементов одного типа. </b></p>
4. <p><b>упорядоченный набор элементов, которые могут удаляться с её начала и помещаться в её конец</b></p>

Ответ: 1

Комментарий: SD003054

**<p><b>std::map – это</b></p>**

1. <p><b>упорядоченный последовательный контейнер</b></p>
2. <p><b>неупорядоченный последовательный контейнер</b></p>
3. <p><b>упорядоченный ассоциативный контейнер</b></p>
4. <p><b>неупорядоченный ассоциативный контейнер</b></p>

Ответ: 3

Комментарий: SD003055

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp{ {4, 1}, {2, 3} };<br>mp[0] = 5;<br>std::map&lt;int, int&gt;::iterator it = mp.begin();<br>std::cout &lt;&lt; it++-&gt;second;</b></p>**

1. <p><b>1</b></p>
2. <p><b>3</b></p>
3. <p><b>5</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 3

Комментарий: SD003056

**<p><b>Какая инициализация содержит ошибку?<br>std::map&lt;int, int&gt; mp1{ { 1, 2.5 }, { 'a', 4 } };<br>std::map&lt;unsigned int, int&gt; mp2 = { { -5, -15 } };<br>std::map&lt;char, char&gt; mp3 = { 79, 89 };</b></p>**

1. <p><b>Инициализация mp1.</b></p>
2. <p><b>Инициализация mp2.</b></p>
3. <p><b>Инициализация mp3.</b></p>
4. <p><b>Ошибок нет.</b></p>

Ответ: 3

Комментарий: SD003057

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp{ {4, 1}, {2, 3} };<br>std::map&lt;int, int&gt;::iterator it = mp.begin();<br>it += 1;<br>std::cout &lt;&lt; it-&gt;first;</b></p>**

1. <p><b>1</b></p>
2. <p><b>2</b></p>
3. <p><b>4</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 4

Комментарий: SD003058

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;char, int&gt; mp2 = { {97.0f, true}, {97.99f, false} };<br>auto it = mp2.begin();<br>std::cout &lt;&lt; (it++)-&gt;second &lt;&lt; ' ';<br>std::cout &lt;&lt; it-&gt;second;</b></p>**

1. <p><b>1 0</b></p>
2. <p><b>true false</b></p>
3. <p><b>Ошибка компиляции.</b></p>
4. <p><b>Ошибка времени выполнения</b></p>

Ответ: 4

Комментарий: SD003059

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp = { {'0', false}, {'1', true} };<br>for (auto it = mp.begin(); \*it; it++) {<br>&ensp;if (it-&gt;second) std::cout &lt;&lt; it-&gt;first;<br>}</b></p>**

1. <p><b>1</b></p>
2. <p><b>49</b></p>
3. <p><b>Ошибка компиляции.</b></p>
4. <p><b>Ошибка времени выполнения</b></p>

Ответ: 3

Комментарий: SD003060

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp = { {3, 1}, {4, 1}, {5, 9}, {2, 6} };<br>for (auto it : mp) {<br>&ensp;std::cout &lt;&lt; it.second &lt;&lt; " ";<br>}</b></p>**

1. <p><b>1 1 9 6 </b></p>
2. <p><b>6 1 1 9 </b></p>
3. <p><b>Программа содержит бесконечный цикл.</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 2

Комментарий: SD003061

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp = { {1, 8}, {2, 7}, {3, 6}, {4, 5} };<br>for (auto it = mp.end(); it != mp.begin(); ) {<br>&ensp;std::cout &lt;&lt; it---&gt;second &lt;&lt; ' ';<br>}</b></p>**

1. <p><b>5 6 7 </b></p>
2. <p><b>6 7 8 </b></p>
3. <p><b>5 6 7 8 </b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 4

Комментарий: SD003062

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp = { {1, 2}, {3, 4}, {5, 6} };<br>mp.insert(std::pair&lt;int, int&gt;(1, 3));<br>std::cout &lt;&lt; mp[1] &lt;&lt; ' ' &lt;&lt; mp.count(1) &lt;&lt; ' ';<br>std::cout &lt;&lt; mp.find(mp.size())-&gt;second;</b></p>**

1. <p><b>2 1 4</b></p>
2. <p><b>2 2 4</b></p>
3. <p><b>3 1 4</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 1

Комментарий: SD003063

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp = { {1, 2}, {3, 4}, {5, 6} };<br>std::cout &lt;&lt; (--mp.rend())-&gt;first &lt;&lt; ' ';<br>if (mp.emplace(std::pair&lt;int, int&gt;(3, 7)).second)<br>&ensp;std::cout &lt;&lt; mp[3];</b></p>**

1. <p><b>1</b></p>
2. <p><b>5</b></p>
3. <p><b>1 7</b></p>
4. <p><b>5 7</b></p>

Ответ: 1

Комментарий: SD003064

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;int, int&gt; mp = { {1, 3}, {2, 4}, {3, 5}, {4, 2} };<br>std::cout &lt;&lt; mp.at(3) &lt;&lt; ' ';<br>mp.erase(2);<br>std::cout &lt;&lt; mp.lower\_bound(2)-&gt;first;</b></p>**

1. <p><b>1 1</b></p>
2. <p><b>1 2</b></p>
3. <p><b>5 2</b></p>
4. <p><b>5 3</b></p>

Ответ: 4

Комментарий: SD003065

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::map&lt;long, short&gt; mp = { {0, 1}, {0, 2}, {0, 3} };<br>std::cout &lt;&lt; mp.size() &lt;&lt; ' ' &lt;&lt; sizeof(mp[0]) &lt;&lt; ' ';<br>mp.clear;<br>std::cout &lt;&lt; mp.size();</b></p>**

1. <p><b>1 2 0</b></p>
2. <p><b>1 4 0</b></p>
3. <p><b>3 8 0</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 4

Комментарий: SD003066

**<p><b>std::vector – это</b></p>**

1. <p><b>упорядоченный последовательный контейнер</b></p>
2. <p><b>неупорядоченный последовательный контейнер</b></p>
3. <p><b>упорядоченный ассоциативный контейнер</b></p>
4. <p><b>неупорядоченный ассоциативный контейнер</b></p>

Ответ: 2

Комментарий: SD003067

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;char&gt; v1; v1 = { 'a' };<br>std::vector&lt;int&gt; v2 = { 1, 2 }; v2 = { 3 };<br>std::vector&lt;int&gt; v3(3, 5);<br>std::cout &lt;&lt; v1[0] &lt;&lt; v2[0] &lt;&lt; v3[0];</b></p>**

1. <p><b>a13</b></p>
2. <p><b>a33</b></p>
3. <p><b>a35</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 3

Комментарий: SD003068

**<p><b>Какая строка вызовет ошибку компиляции?<br>std::vector&lt;int&gt; v1(10);std::cout &lt;&lt; v1[0];<br>std::vector&lt;int&gt; v2[10]; std::cout &lt;&lt; v2[0];<br>std::vector&lt;int&gt; v3;<br>v3.reserve(10); std::cout &lt;&lt; v3[0];</b></p>**

1. <p><b>Первая.</b></p>
2. <p><b>Вторая.</b></p>
3. <p><b>Четвёртая.</b></p>
4. <p><b>Ни одна.</b></p>

Ответ: 2

Комментарий: SD003069

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v = { 20, 15, 14, 13 };<br>std::vector&lt;int&gt;::iterator it = v.begin();<br>it += 2;<br>std::cout &lt;&lt; v.at(3) &lt;&lt; \*--it;</b></p>**

1. <p><b>1314</b></p>
2. <p><b>1315</b></p>
3. <p><b>2014</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 2

Комментарий: SD003070

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v1(3, 4);<br>std::vector&lt;int&gt; v2; v2.reserve(3);<br>std::vector&lt;int&gt; v3(v1); v3 = v2;<br>std::cout &lt;&lt; v1.at(0) &lt;&lt; ' ' &lt;&lt; v3[0];</b></p>**

1. <p><b>3 0</b></p>
2. <p><b>3 3</b></p>
3. <p><b>4 0</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 4

Комментарий: SD003071

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v(3, 1);<br>v.resize(5);<br>std::cout &lt;&lt; v[3];</b></p>**

1. <p><b>0</b></p>
2. <p><b>1</b></p>
3. <p><b>3</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 1

Комментарий: SD003072

**<p><b>Какие операции сравнения векторов можно использовать?</b></p>**

1. <p><b>Никакие.</b></p>
2. <p><b>==  !=</b></p>
3. <p><b>==  !=  &gt;  &lt;</b></p>
4. <p><b>==  !=  &gt;  &lt;  &gt;=  &lt;=</b></p>

Ответ: 4

Комментарий: SD003073

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v1(3, 1);<br>std::vector&lt;int&gt; v2{ 1, 3 };<br>std::vector&lt;int&gt; v3;<br>std::cout &lt;&lt; (v1 &gt; v2) &lt;&lt; (v1[0] == v2[0]);</b></p>**

1. <p><b>01</b></p>
2. <p><b>10</b></p>
3. <p><b>11</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 1

Комментарий: SD003074

**<p><b>Какая строка вызовет ошибку компиляции?<br>std::vector&lt;std::vector&lt;int&gt;&gt; v{ { 1 }, { 2, 3 } };<br>v = { { 1, 2, 3, 4, 5, 6 } };<br>v = { 1, 2, 3, 4, 5, 6 };</b></p>**

1. <p><b>Первая.</b></p>
2. <p><b>Вторая.</b></p>
3. <p><b>Третья.</b></p>
4. <p><b>Ни одна.</b></p>

Ответ: 3

Комментарий: SD003075

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;std::vector&lt;int&gt;&gt; v2;<br>v2 = { { 1, 2, 3 }, { 4, 5, 6 }, {(2, 7)} };<br>for (int i = 0; i &lt; 9; i += 3)<br>&ensp;std::cout &lt;&lt; v2[i / 3][i % 3];</b></p>**

1. <p><b>147</b></p>
2. <p><b>157</b></p>
3. <p><b>Ошибка компиляции.</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 1

Комментарий: SD003076

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;std::vector&lt;int&gt;&gt; v{ { 0, 1, 2, 3, }, { 4, 5, 6 } };<br>v[1].resize(0);<br>if (v[1].empty()) v[1].assign(1, 9);<br>std::cout &lt;&lt; v.size() &lt;&lt; v[1].size();</b></p>**

1. <p><b>21</b></p>
2. <p><b>23</b></p>
3. <p><b>29</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 1

Комментарий: SD003077

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;std::vector&lt;int&gt;&gt; v{ { 0, 1, 2, 3 }, { } };<br>std::cout &lt;&lt; v.size();<br>if (v[1].empty()) v[1].resize(5);<br>std::cout &lt;&lt; v.size() &lt;&lt; v[1].size();</b></p>**

1. <p><b>121</b></p>
2. <p><b>125</b></p>
3. <p><b>221</b></p>
4. <p><b>225</b></p>

Ответ: 4

Комментарий: SD003078

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v1(3, 1), v2{ 1, 3 };<br>v2.push\_back(3);<br>v2.push\_back(v1[1]);<br>for (int i : v2) std::cout &lt;&lt; i;</b></p>**

1. <p><b>130001</b></p>
2. <p><b>1331</b></p>
3. <p><b>1333</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 2

Комментарий: SD003079

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v1(3, 1), v2{ 1, 3 };<br>v1.pop\_back();<br>if (v1.empty()) v2.pop\_back();<br>for (int i : v2) std::cout &lt;&lt; i;</b></p>**

1. <p><b>1</b></p>
2. <p><b>13</b></p>
3. <p><b>Ошибка компиляции.</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 1

Комментарий: SD003080

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v1{ 1 }; auto it = v1.begin();<br>v1.insert(it, 4);<br>v1.insert(it, 5);<br>for (int i : v1) std::cout &lt;&lt; i;</b></p>**

1. <p><b>154</b></p>
2. <p><b>451</b></p>
3. <p><b>541</b></p>
4. <p><b>Ошибка времени выполнения.</b></p>

Ответ: 4

Комментарий: SD003081

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v1{ 1, 2, 3 };<br>auto it = --v1.end();<br>v1.insert(it, { 1, 2 });<br>for (int i : v1) std::cout &lt;&lt; i;</b></p>**

1. <p><b>12312</b></p>
2. <p><b>12123</b></p>
3. <p><b>11223</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 2

Комментарий: SD003082

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v{ 4, 6, 6, 7 };<br>v.front() = v.back();<br>v.back() = (v.front() + 1);<br>for (int i : v) std::cout &lt;&lt; i;</b></p>**

1. <p><b>4667</b></p>
2. <p><b>4668</b></p>
3. <p><b>7668</b></p>
4. <p><b>Ошибка компиляции.</b></p>

Ответ: 3

Комментарий: SD003083

**<p><b>Что будет выведено на консоль (если только нет ошибок)?<br>std::vector&lt;int&gt; v{ 5, 6 };<br>std::cout &lt;&lt; (bool)(v.front() = v.back());</b></p>**

1. <p><b>0</b></p>
2. <p><b>1</b></p>
3. <p><b>5</b></p>
4. <p><b>6</b></p>

Ответ: 2

Комментарий: SD003084

04\_База данных на файлах

**<p><b>При помощи какой функции можно определить конец файла?</b></p>**

1. <p><b>eof()</b></p>
2. <p><b>CountLinesFile()</b></p>
3. <p><b>GetStringsFromFile()</b></p>
4. <p><b>eaf()</b></p>

Ответ: 1

Комментарий: SD004001

**<p><b>Что происходит на данном участке кода?<br>while (!file.eof)<br/ >{<br/ >&ensp;file >&gt; c;<br/ >cout <&lt; c;<br/ >}</b></p>**

1. <p><b>В тот момент, когда в файле будет происходить считывание последнего символа, будет осуществлен вывод абсолютно всех символов сразу</b></p>
2. <p><b>Пока файл не пуст, происходит считывание из файла по одному символу и вывод на экран</b></p>
3. <p><b>Присутствует ошибка</b></p>
4. <p><b>Происходит чтение и вывод сразу всех символов из файла</b></p>

Ответ: 2

Комментарий: SD004002

**<p><b>В большинстве случаев функция eof() используется в цикле while при чтении строк файла.</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004003

**<p><b>За что отвечает функция CountLinesInFile в данной части кода?<br>void main()<br/ >{<br/ >&ensp;int nLines = CountLinesInFile((char\*)"TextFile1.txt");<br/ >&ensp;if (nLines == - <br/ >&ensp;cout <&lt; "Error opening file";<br/ >&ensp;else<br/ >&ensp;cout <&lt; "nLines = " <&lt; nLines <&lt; endl; <br/ >}<br/ ></b></p>**

1. <p><b>За определение конца файла</b></p>
2. <p><b>За получение массива строк из текстового файла</b></p>
3. <p><b>За обмен местами двух строк в файле</b></p>
4. <p><b>За подсчет кол-ва строк в текстовом файле</b></p>

Ответ: 4

Комментарий: SD004004

**<p><b>Правда ли, что функция GetStringsFromFileS() получает массив строк(список) типа char\* из текстового файла, а функция GetStingsFileC() получает массив строк типа string из текстового файла?</b></p>**

1. <p><b>ДА</b></p>
2. <p><b> НЕТ</b></p>

Ответ: 2

Комментарий: SD004005

**<p><b>Библиотека fstream используется для: </b></p>**

1. <p><b>Применения модификатора setw</b></p>
2. <p><b>Работы с простейшими математическими операциями</b></p>
3. <p><b>Работы с файлами: чтение/запись из/в файл(а). </b></p>
4. <p><b>Работы с со строками: strlen, strcpy, strcat, strcmp</b></p>

Ответ: 3

Комментарий: SD004006

**<p><b>Функция SetStringsToFileS() используется для: </b></p>**

1. <p><b>Сортировки строк файла</b></p>
2. <p><b>Записи массива(списка) строк из файла по его номеру</b></p>
3. <p><b>Вставки в заданную позицию в файле</b></p>
4. <p><b>Замены строки в текстовом файле</b></p>

Ответ: 2

Комментарий: SD004007

**<p><b> Функция ChangeStringInFileC() предназначена для замены строки в файле в заданной позиции. </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004008

**<p><b> Функция ChangeStringInFileC() использует следующие функции: </b></p>**

1. <p><b>CountLinesInFile()</b></p>
2. <p><b>GetStringsFromFileC() </b></p>
3. <p><b>SetStringsToFileS()</b></p>
4. <p><b>GetStringsFromFileS()</b></p>

Ответ: 1, 2

Комментарий: SD004009

**<p><b> Функция RemoveStringFormFileByIndex() предназначена для удаления строки из файла по ее номеру </b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>

Ответ: 2

Комментарий: SD004010

**<p><b> Для чего используется функция InsertStringToFile(), вызванная в 6 строчке кода? </b></p>**

1. <p><b>Для вставки строки в заданную позицию в файле</b></p>
2. <p><b>Для обмена местами двух строк в файле</b></p>
3. <p><b>Для реверсии строк файла</b></p>
4. <p><b>Для сортировки строк файла</b></p>

Ответ: 1

Комментарий: SD004011

**<p><b> При помощи функции SwapStringsFile() может быть осуществлен обмен местами двух строк в файле</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004012

**<p><b>Для чего используется функция ReversStringsInFile()?</b></p>**

1. <p><b>Для реверсии строк файла</b></p>
2. <p><b>Для вставки в заданную позицию в вайле</b></p>
3. <p><b>Для определения конца файла</b></p>
4. <p><b>Для реверсирования строк файла</b></p>

Ответ: 4

Комментарий: SD004013

**<p><b> Составьте последовательность алгоритма метода действия функции ReversStringsInFile().</b></p>**

1. <p><b> обменять местами элементы массива так, чтобы строки массива размещались в обратном порядке</b></p>
2. <p><b> записать измененный массив снова в файл</b></p>
3. <p><b> считать строки из файла и записать их в массив</b></p>
4. <p><b> Поменять строки в списке</b></p>

Ответ: 2, 3, 1

Комментарий: SD004014

**<p><b> Функция SortStringsInFile() нужна для сортировки строк в файле? </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004015

**<p><b> Если в файле 6 строк, то выводом будет слово:<br>void main()<br/ >{<br/ >&ensp;string filename = "TextFile1.txt"; <br/ >&ensp;int pos1 = 2; <br/ >&ensp;int pos2 = 7; <br/ >&ensp;if (SwapStringsInFile(filename,pos1, pos) <br/ >&ensp;{<br/ >&ensp;&ensp;cout << "OK!!!"; <br/ >&ensp;}<br/ >&ensp;else<br/ >&ensp;{<br/ >&ensp;&ensp;cout <&lt; "Error";<br/ >&ensp;}<br/ >}</b></p>**

1. <p><b>  OK!!! </b></p>
2. <p><b>  Error</b></p>

Ответ: 2

Комментарий: SD004016

**<p><b> Что является результатом вывода?<br>void main()<br/ >{&ensp;<br/ >&ensp;string filename = "TextFile4.txt"; <br/ >&ensp;if (SortStringsInFile(filename) <br/ >&ensp;{<br/ >&ensp;** **&ensp;cout <&lt; "OK!!!"; <br/ >&ensp;}<br/ >&ensp;else<br/ >&ensp;{<br/ >&ensp;** **&ensp;cout <&lt; "Error";<br/ >&ensp;}<br/ >}</b></p>**

1. <p><b>0K!!!</b></p>
2. <p><b>Error</b></p>

Ответ: 2

Комментарий: SD004017

**<p><b> Что является результатом вывода?<br>void main()<br/ >{<br/ >&ensp;string filename = "TextFile3.txt"; <br/ >&ensp;if (ReverseStringsInFile(filename)) <br/ >&ensp;{<br/ >&ensp;&ensp;cout <&lt; "OK!!!"; <br/ >&ensp;}<br/ >&ensp;else<br/ >&ensp;{<br/ >&ensp;&ensp;cout <&lt; "Error";<br/ >&ensp;}<br/ >}</b></p>**

1. <p><b>OK!!! </b></p>
2. <p><b>Error</b></p>

Ответ: 1

Комментарий: SD004018

**<p><b> SwapStringsInFile(string filename, int pos1, int pos2) Нумерация позиции строки pos1 или pos2 начинается с: </b></p>**

1. <p><b>0</b></p>
2. <p><b>1</b></p>
3. <p><b>2</b></p>
4. <p><b>3</b></p>

Ответ: 1

Комментарий: SD004019

**<p><b>Схема базы данных-это:</b></p>**

1. <p><b>Схема, описывающий форму данных, как они могут быть связаны с другими таблицами или моделями. </b></p>
2. <p><b>план того, как будут выглядеть наши данные. Причем он содержит сами данные и непосредственно работает с ними. </b></p>
3. <p><b>План того, как будут обрабатываться наши данные, с последующей записью их в таблицы или модели. </b></p>

Ответ: 1

Комментарий: SD004020

**<p><b> Под файлом понимается:</b></p>**

1. <p><b>Некоторая безымянная последовательность данных </b></p>
2. <p><b>Некоторая последовательность байтов, но не больше 100, которая имеет свое уникальное имя</b></p>
3. <p><b>Некоторая последовательность байтов, которая имеет свое уникальное имя</b></p>
4. <p><b>Затрудняюсь ответить</b></p>

Ответ: 3

Комментарий: SD004021

**<p><b>С помощью класса open(),  мы можем создать файл в текущей директории с программой, с именем, записанным в круглые скобки. </b></p>**

1. <p><b>Да</b></p>
2. <p><b> Нет</b></p>

Ответ: 1

Комментарий: SD004022

**<p><b>Примером какой функции служит данный код?<br>#include <iostream> <br/ >using namespace std; <br/ >#include <fstream><br/ >int CountLinesInFile(char\* filename) <br/ >{ <br>&ensp;ifstream F(filename, ios::in); <br/ >&ensp;if (!F) <br/ >&ensp;{<br/ >&ensp;&ensp;return -1; <br/ >&ensp;}<br/ >&ensp;int count = 0; <br/ >&ensp;char buffer[1000]; <br/ >&ensp;while (!F.eof())<br/ >&ensp;{<br/ >&ensp;&ensp;count++;<br/ >&ensp;&ensp;F.getline(buffer, 1000); <br/ >&ensp;}<br/ >&ensp;F.close();<br/ >&ensp;return count; <br/ >}</b></p>**

1. <p><b> GetStringsFromFileS()</b></p>
2. <p><b> GetStringsFromFileC()</b></p>
3. <p><b> CountLinesInFile()</b></p>
4. <p><b> SetStringsToFileS()</b></p>

Ответ: 3

Комментарий: SD004023

**<p><b>функция GetStringsFromFile()предназначена для получить массива (списка) строк типа char\* из текстового файла</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004024

**<p><b>Функция GetstringsFromFileS() предназначена для получения строк типа string из текстового файла</b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>

Ответ: 2

Комментарий: SD004025

**<p><b>Функция SetStringsToFileS()предназначена для записи массива(списка)строк типа string в текстовом файле</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004026

**<p><b>Функция ChangeStringInFileC()предназначена для замены строки в текстовом файле </b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>

Ответ: 2

Комментарий: SD004027

**<p><b>Функция RemoveStringFromFileByIndex()нужна для удаления строки из файла по его номеру</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004028

**<p><b>SwapStringInFile()совершает обмен местами двух строк в файле</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004029

**<p><b>ReversStringInFile()совершает реверсирование строк файла </b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD004030

**<p><b> Расшифровка STL </b></p>**

1. <p><b> Библиотека стандартных шаблонов</b></p>
2. <p><b> Библиотека сторонних шаблонов </b></p>
3. <p><b> Библиотека стандартных контейнеров</b></p>
4. <p><b> Фреймворк стандартных шаблонов </b></p>

Ответ: 1

Комментарий: SD004031

**<p><b> Выберите верные утверждения для итератора </b></p>**

1. <p><b> Итератор — объект, предоставляющий доступ к элементам контейнера и позволяющий их <br>перебирать </b></p>
2. <p><b> Основные требования к итераторам — наличие операторов разыменования и инкремента </b></p>
3. <p> Итератор является свойством контейнера <b> </b></p>
4. <p><b> Итератор, указывающий на первый элемент, можно получить при помощи метода<br> iterator first(); </b></p>

Ответ: 1,2,3

Комментарий: SD004032

**<p><b>  Контейнер (container)  управляет набором объектов в памяти?  </b></p>**

1. <p><b> Да</b></p>
2. <p><b> Нет</b></p>

Ответ: 1

Комментарий: SD004033

**<p><b> Для доступа к элементам вектора можно использовать квадратные скобки []? </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 1

Комментарий: SD004034

**<p><b> list::insert() используется для </b></p>**

1. <p><b> нахождения минимального элемента </b></p>
2. <p><b> того чтобы найти определенное значение в списке </b></p>
3. <p><b> добавления нового значения в список </b></p>
4. <p><b> возвращает ссылку на последний элемент </b></p>

Ответ: 3

Комментарий: SD004035

**<p><b> Что делает данная строчка кода<br>std::list&lt;int&gt;::const iterator it;</b></p>**

1. <p><b> Выполняем сортировку при помощи итератора</b></p>
2. <p><b> Возвращаем значение итератора </b></p>
3. <p><b> Объявляем итератор </b></p>
4. <p><b> Эта строчка не имеет смысла </b></p>

Ответ: 3

Комментарий: SD004036

**<p><b> В очереди 4 элемента. Какое згачение вернёт функция empty() </b></p>**

1. <p><b> false </b></p>
2. <p><b> true</b></p>
3. <p><b> 4</b></p>
4. <p><b> -NaN </b></p>

Ответ: 1

Комментарий: SD004037

**<p><b> Выберите верные утверждения </b></p>**

1. <p><b> pop\_back() — удалить последний элемент </b></p>
2. <p><b> clear() — удалить все элементы вектора </b></p>
3. <p><b> empty() — проверить вектор на пустоту </b></p>
4. <p><b> ничего не верно </b></p>

Ответ: 1, 2, 3

Комментарий: SD004038

**<p><b> Какого типа очереди не существует </b></p>**

1. <p><b> кольцевая очередь </b></p>
2. <p><b> простая очередь </b></p>
3. <p><b> очередь с приоритетами </b></p>
4. <p><b> очередь с запросом </b></p>

Ответ: 4

Комментарий: SD004039

**<p><b> Какой лишний способ реализации очереди </b></p>**

1. <p><b> Статический массив</b></p>
2. <p><b>Динамический массив </b></p>
3. <p><b> Односвязный список</b></p>
4. <p><b> Условный массив</b></p>

Ответ: 4

Комментарий: SD004040

**<p><b> Какое значение возвращает функция Get()</b></p>**

1. <p><b> Получить какой-то элемент</b></p>
2. <p><b> Проверить наличие на элемент </b></p>
3. <p><b> Выводит очередь</b></p>
4. <p><b> Возвращает значение количества элементов очереди </b></p>

Ответ: 4

Комментарий: SD004041

**<p><b> Чем отличаются методы  pop() и GetItem() </b></p>**

1. <p><b> Они идентичны</b></p>
2. 2<p><b> Они выполняют совершенно разные действия </b></p>
3. <p><b> Рор() вытягивает первый элемент из очереди а GetItem() читает  из очереди не вытягивая <br>его</b></p>
4. <p><b> Один из методов не существует </b></p>

Ответ: 3

Комментарий: SD004042

**<p><b> Вершины бинарного дерева может иметь 3 и более вершин </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD004043

**<p><b> узел дерева, не имеющий потомков называется лист </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 1

Комментарий: SD004044

**<p><b> Что делает данная строчка   sort (vec.begin(), vec.end());</b></p>**

1. <p><b> Сортирует строку</b></p>
2. <p><b> Сортирует вектор с конца к началу</b></p>
3. <p><b> Ничего не делает</b></p>
4. <p><b> Сортирует вектор с начала к концу</b></p>

Ответ: 4

Комментарий: SD004045

**<p><b> Сортировка слиянием (англ. merge sort) — алгоритм сортировки, который<br>упорядочивает списки (или другие структуры данных, доступ к элементам которых можно <br>получать только последовательно, например — потоки) в определённом порядке. Верно? </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 1

Комментарий: SD004046

**<p><b> Выберите верное утверждение b></p>**

1. <p><b> размер памяти, дерева, заранее известен, потому что известно, сколько узлов будет в <br>  него входить </b></p>
2. <p><b> размер памяти, дерева, заранее известен, потому что количество узлов не влияет на память <br>  дерева </b></p>
3. <p><b> размер памяти, дерева, заранее неизвестен, потому что неизвестно, сколько узлов будет <br>   в него входить.</b></p>
4. <p><b> Все варианты не верны</b></p>

Ответ: 3

Комментарий: SD004047

**<p><b> Что выпонит этот код <br>struct Node<br>{<br>&ensp;int key; <br>Node  \*left, \*right; <br>};<br>typedef Node \*PNode; <br></b></p>**

1. <p><b> Полностью описывает дерево </b></p>
2. <p><b> Описывает только вершину дерева </b></p>
3. <p><b> Описание правого и левого поlдеревьев.</b></p>
4. <p><b> Описание корня дерева </b></p>

Ответ: 2

Комментарий: SD004048

**<p><b> Выберете правильные варианты обхода дерева</b></p>**

1. <p><b> ЛПК </b></p>
2. <p><b> ЛНК </b></p>
3. <p><b> ПКЛ </b></p>
4. <p><b> КЛП </b></p>

Ответ: 1,3,4

Комментарий: SD004049

**<p><b Верно ли что, указатель на корень дерева надо передавать именем указателя, так как он не <br>может измениться при создании новой вершины</b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD004050

**<p><b> Построение дерева поиска. Выберете необходимы пункты </b></p>**

1. <p><b> Если текущее дерево пустое, игнорировать его и не создавать новую вершину .</b></p>
2. <p><b> Сравнить ключ очередного элемента массива с ключом корня.</b></p>
3. <p><b> Если ключ нового элемента меньше, включить его в левое поддерево, если больше <br>или равен, то в правое.</b></p>
4. <p><b> Если текущее дерево пустое, создать новую вершину и включить в дерево.</b></p>

Ответ: 2,3,4

Комментарий: SD004051

**<p><b>Можно ли представить игру крестики нолики как бинарное дерево</b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 1

Комментарий: SD004052

**<p><b> Верно ли что, двоичная куча (binary heap) –структура данных, позволяющая быстро <br> (за логарифмическое время) добавлять элементы и извлекать элемент с максимальным приоритетом </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD004053

**<p><b> Двоичная куча представляет собой полное бинарное дерево, для которого не обязательно<br>  выполняется основное свойство кучи. Верно ли утверждение </b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD004054

**<p><b> При добавлении нового элемента в кучу, он занимает место :</b></p>**

1. <p><b> На последнее место в массиве </b></p>
2. <p><b> На позицию с максимальным индексом </b></p>
3. <p><b> На первое место в массиве</b></p>
4. <p><b> На позицию с минимальным индексом</b></p>

Ответ: 1,2

Комментарий: SD004055

**<p><b> Метод heapify. Выберите верные утверждения</b></p>**

1. <p><b> Восстанавливает основное свойство кучи для дерева с корнем в i-ой вершине </b></p>
2. <p><b> Восстанавливает основное свойство кучи для дерева с корнем в i-ой <br>  вершине при условии, что оба поддерева ему удовлетворяют </b></p>
3. <p><b> сложность этого алгоритма также равна O(log2 N)</b></p>
4. <p><b> сложность этого алгоритма также равна O(log2 2\*N)</b></p>

Ответ: 2,3

Комментарий: SD004056

**<p><b> Метод heapify. Выберите верные утверждения</b></p>**

1. <p><b> Восстанавливает основное свойство кучи для дерева с корнем в i-ой вершине </b></p>
2. <p><b> Восстанавливает основное свойство кучи для дерева с корнем в i-ой <br>  вершине при условии, что оба поддерева ему удовлетворяют </b></p>
3. <p><b> сложность этого алгоритма также равна O(log2 N)</b></p>
4. <p><b> сложность этого алгоритма также равна O(log2 2\*N)</b></p>

Ответ: 2,3

Комментарий: SD004057

**<p><b>Что описывается этой строчкой кода <br>template<typename ITERATOR, typename RESULT><br>RESULT an\_algorithm(ITERATOR first, ITERATOR last, ...);</b></p>**

1. <p><b> Введение алгоритма</b></p>
2. <p><b> Введения итератора</b></p>
3. <p><b> Ввод фйала </b></p>
4. <p><b> Создание нового класса </b></p>

Ответ: 1

Комментарий: SD004058

**<p><b>Что описывается этой строчкой кода <br>Имеется vector [1, 4,7,9,2,3,] <br>sort (vec.begin(), vec.end());  // сортировка<br>cout << endl << "После сортировки: ";<br> <br> for (int i = 0; i < n; i++) {<br> cout << vec[i] << " ";<br> }<br>Что выведется</b></p>**

1. <p><b> 1 2 3 4 7 9</b></p>
2. <p><b> 9 7 4 3 2 1</b></p>
3. <p><b> 329741</b></p>
4. <p><b> В коде есть ошибка </b></p>

Ответ: 1

Комментарий: SD004059

**<p><b>Верна ли следующая строчка кода<br> vector new<int> v1</b></p>**

1. <p><b>Да </b></p>
2. <p><b> Нет</b></p>

Ответ: 2

Комментарий: SD004060

05\_Манипуляции со структурами данных

**<p><b>Сортировка – это..</b></p>**

1. <p><b> ..упорядочивание данных по некоторым признакам</b></p>
2. <p><b>..постановка объектов массива в порядки</b></p>
3. <p><b>..перестановка объектов местами с целью изменения результата</b></p>

Ответ: 1

Комментарий: SD005001

**<p><b>Верно ли утверждение, что сортировки бывают разных видов?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005002

**<p><b>Можно ли сортировать строки?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005003

**<p><b>С какой позиции начинается массив?</b></p>**

1. <p><b>1</b></p>
2. <p><b>0</b></p>
3. <p><b>это задаётся с клавиатуры</b></p>
4. <p><b>с того, что мы указываем в квадратных скобках</b></p>

Ответ: 2

Комментарий: SD005004

**<p><b>Верно ли утверждение, что имя массива - это указатель на адрес его элементов?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD005005

**<p><b>Правильно ли задан массив в данном случае?<br>int main() <br>{<br>int &ensp; b = 4;<br>int &ensp; a[b];<br>}</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD005006

**<p><b>Однонаправленный связанный список – это..</b></p>**

1. <p><b>..динамическая структура данных, состоящая из узлов, которые имеют какое-то значение.</b></p>
2. <p><b>..структура данных, которая имеет большое количество элементов, имеющих заданный порядок.</b></p>
3. <p><b>..массив, размер которого может изменяться во время исполнения программы.</b></p>

Ответ: 1

Комментарий: SD005007

**<p><b>Верно ли, что связные списки медленнее динамических массивов в изменении/удалении элементов?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD005008

**<p><b>Правильно ли утверждение, что односвязный список не способен иметь направление в несколько сторон?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005009

**<p><b>Двунаправленный связный список – это..</b></p>**

1. <p><b>..динамическая структура данных, состоящая из узлов, каждый из которых содержит данные и ссылку.</b></p>
2. <p><b>..динамическая структура данных, которая имеет большое количество элементов, изменяющихся двунаправленно</b></p>
3. <p><b>..массив, размер которого может изменяться в 2 стороны.</b></p>

Ответ: 1

Комментарий: SD005010

**<p><b>Верно ли утверждение, что двунаправленный связный список может содержать более одной ссылки?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005011

**<p><b>Нагружает ли двунаправленный связный список систему больше, чем однонаправленный?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD005012

**<p><b>Стек – это..</b></p>**

1. <p><b>..абстрактный тип данных, представляющий собой массив элементов.</b></p>
2. <p><b>..абстрактный тип данных, представляющий собой список элементов.</b></p>

Ответ: 2

Комментарий: SD005013

**<p><b>По какому принципу работает стек?</b></p>**

1. <p><b>XD</b></p>
2. <p><b>LILO</b></p>
3. <p><b>LIFO</b></p>
4. <p><b>QW</b></p>

Ответ: 3

Комментарий: SD005014

**<p><b>Требует ли stack подключение дополнительных библиотек?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005015

**<p><b>Очередь – это..</b></p>**

1. <p><b>..структура данных, которая работает по принципу LILO</b></p>
2. <p><b>..структура данных, которая работает по принципу LIFO</b></p>
3. <p><b>Не знаю(неправильный ответ кстати)</b></p>

Ответ: 1

Комментарий: SD005016

**<p><b>Возможно ли реализовать queue без библиотеки?</b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>

Ответ: 2

Комментарий: SD005017

**<p><b>Отличается ли чем-то очередь от стека?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005018

**<p><b>Дерево – это..</b></p>**

1. <p><b>..структура данных, представляющая собой древовидную структуру в виде набора связанных узлов.</b></p>
2. <p><b>..структура данных, представляющая собой набор параметров и элементов, реализующих эти параметры.</b></p>
3. <p><b>..структура данных, представляющая собой гибрид queue.</b></p>

Ответ: 1

Комментарий: SD005019

**<p><b>Как называется элемент, который стоит на вершине дерева?</b></p>**

1. <p><b>Крона</b></p>
2. <p><b>Лист</b></p>
3. <p><b>Корень</b></p>

Ответ: 3

Комментарий: SD005020

**<p><b>Имеются узлы Q и W, причём они расположены относительно друг друга в соответствующем порядке(от Q к W). Кем является Q для W?</b></p>**

1. <p><b>Предком</b></p>
2. <p><b>Потомком</b></p>
3. <p><b>Родителем</b></p>
4. <p><b>Ребёнком</b></p>

Ответ: 1

Комментарий: SD005021

**<p><b>Отличаются ли чем-то структуры данных куча и дерево?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b>

Ответ: 1

Комментарий: SD005022

**<p><b>По какому(-им) принципу(-ам) работают кучи?(какому(-им) св-ву они удовлетворяют) [несколько вариантов ответа]</b></p>**

1. <p><b>Существует 2 типа куч, max-куча и min-куча.</b></p>
2. <p><b>Существует лишь один тип куч.</b></p>
3. <p><b>Модель кучи похожа на модель дерева</b></p>
4. <p><b>min и max кучи работают от наибольшего к наименьшему и от наименьшего к наибольшему соответственно</b></p>

Ответ: 1, 3

Комментарий: SD005023

**<p><b>Может ли создать подобная структура данных утечку памяти?</b></p>**

1. <p><b>Нет</b></p>
2. <p><b>Да</b></p>

Ответ: 2

Комментарий: SD005024

**<p><b>Поиск имеет несколько алгоритмов?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005025

**<p><b>Линейный поиск является самым эффективным?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD005026

**<p><b>Бинарный поиск – .. </b></p>**

1. <p><b>..очень сложный алгоритм с быстрой реализацией, который находит элемент в уже отсортированном массиве.</b></p>
2. <p><b>..очень простой алгоритм с быстрой реализацией, который находит элемент в уже отсортированном массиве.</b></p>
3. <p><b>дешёвая подделка линейного поиска.</b></p>
4. <p><b>..метод сортировки данных с последующим нахождением требуемого элемента.</b></p>

Ответ: 2

Комментарий: SD005027

**<p><b>Правда ли, что фильтрация и сортировка являются одним и тем же действием, но выполняемым по-разному?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 2

Комментарий: SD005028

**<p><b>Фильтрация – это..</b></p>**

1. <p><b>..операция по ограничению значений в результирующем наборе только элементами, соответствующими указанному условию.</b></p>
2. <p><b>..операция по избавлению от элементов в результирующем наборе, соответствующих условию.</b></p>
3. <p><b>..операция по перераспределению элементов в определённый порядок по определённому условию.</b></p>
4. <p><b>..тип данных, меняющий значение выбранных элементов на противоположное.</b></p>

Ответ: 1

Комментарий: SD005029

**<p><b>Можно ли с помощью сортировки имитировать фильтрацию?</b></p>**

1. <p><b>Да</b></p>
2. <p><b>Нет</b></p>

Ответ: 1

Комментарий: SD005030

**<p><b>Виды стека: </b></p>**

1. <p><b>Программный и непрограммный</b></p>
2. <p><b>Аппаратный и программный</b></p>
3. <p><b>Аппаратный и неаппаратный</b></p>
4. <p><b>Затрудняюсь ответить</b></p>

Ответ: 2

Комментарий: SD005031

**<p><b>Как можно задать стек(несколько вариантов ответа): </b></p>**

1. <p><b>Одномерный массив</b></p>
2. <p><b>Функция</b></p>
3. <p><b>ООП - класс</b></p>
4. <p><b>Связной список</b></p>

Ответ: 1, 3, 4

Комментарий: SD005032

**<p><b>Как создать стек: </b></p>**

1. <p><b>stack &lt;тип данных&gt; &lt;имя&gt;</b></p>
2. <p><b>&lt;имя функции&gt;(struct stack &ltимя стека&gt;)</b></p>
3. <p><b>stack &lt;имя&gt;</b></p>
4. <p><b>&lt;имя функции&gt;(struct stack\* &lt;имя стека&gt;)</b></p>

Ответ: 1, 4

Комментарий: SD005033

**<p><b>Примеры работы со стеком: </b></p>**

1. <p><b>Касса - покупатели</b></p>
2. <p><b>В корзинке лежат свежие яблоки, и мы выбираем первое попавшее нам. </b></p>
3. <p><b>Стопка тарелок</b></p>
4. <p><b>Чтобы взять второй лист бумаги сверху, нужно снять нижний</b></p>

Ответ: 3, 4

Комментарий: SD005034

**<p><b>Как посмотреть на верхний элемент стека(несколько вариантов): </b></p>**

1. <p><b>Функция first()</b></p>
2. <p><b>Функция top()</b></p>
3. <p><b>Функция peek()</b></p>
4. <p><b>Функция start()</b></p>

Ответ: 2, 3

Комментарий: SD0050535

**<p><b>Что делает стек вызовов: </b></p>**

1. <p><b>отслеживает все используемые нами функции</b></p>
2. <p><b>отслеживает все реализованные функции</b></p>
3. <p><b>отслеживает все вызванные, но не реализованные функции</b></p>
4. <p><b>Вызывает помощь друга на экзамене. </b></p>

Ответ: 3

Комментарий: SD005036