

**Тестовое задание для диагностического тестирования по дисциплине:**

Технологии программирования, 5 семестр

Код, направление подготовки	09.03.02 Информационные системы и технологии
Направленность (профиль)	Безопасность информационных систем и технологий
Форма обучения	Очная
Кафедра разработчик	Информатики и вычислительной техники
Выпускающая кафедра	Информатики и вычислительной техники

<b>Проверяемая компетенция</b>	<b>Задание</b>	<b>Варианты ответов</b>	<b>Тип сложности вопроса</b>	<b>Кол-во баллов за правильный ответ</b>
--------------------------------	----------------	-------------------------	------------------------------	--

<p>ОПК-1.1, ОПК-2.1, ПК-3.1, ПК-5.1, ПК-11.1</p>	<p>При рассмотрении времени работы <math>T(M)</math> и памяти <math>M(N)</math> что нас интересует?</p>	<p>1. точный вид функций <math>T(N)</math> и <math>M(N)</math>  2. приближенный до константы вид функций.  Используется <math>O</math>-символика  3. приближенный вид функций.  Используется <math>o</math>-символика  4. точный вид функций.  Используется <math>O</math>-символика</p>	<p>Низкий</p>	<p>2</p>
<p>ОПК-1.1, ОПК-2.1, ПК-3.1, ПК-5.1, ПК-11.1</p>	<p>Какие существуют метрики, отражающие эффективность алгоритма?</p>	<p>1. процессорное время, память  2. адаптивность, простота реализации  3. надежность, масштабируемость</p>	<p>Низкий</p>	<p>2</p>
<p>ОПК-1.1, ОПК-2.1, ПК-3.1, ПК-5.1, ПК-11.1</p>	<p>Динамические структуры данных – это структуры данных, _____ под которые _____ и _____ по мере _____</p>		<p>Низкий</p>	<p>2</p>

ОПК-1.1, ОПК-2.1, ПК- 3.1, ПК-5.1, ПК-11.1	При размере входных данных N, как рассчитывается время работы алгоритма?	1. как функция от параметра N 2. не зависимо от N 3. в сравнении с N 4. как $O(N)$	Низкий	2
ОПК-1.1, ОПК-2.1, ПК- 3.1, ПК-5.1, ПК-11.1	Какая оценка снизу справедлива для сортировок?	1. $O(N^2)$ 2. $O(N)$ 3. $O(\log N)$ 4. $O(N \cdot \log N)$	Низкий	2
ОПК-2.2, ПК- 3.2, ПК-5.2, ПК-11.2	Какие две операции должен выполнять стек?	1. enqueue, dequeue 2. set, get 3. insert, delete 4. push, pop	Средний	5
ОПК-2.2, ПК- 3.2, ПК-5.2, ПК-11.2	Соотнесите алгоритмы сортировки с их временной сложностью	1. Пузырьком $\Leftrightarrow O(N+K)$ 2. Быстрая $\Leftrightarrow O(N \cdot \log(N))$ 3. Подсчётом $\Leftrightarrow O(N^2)$	Средний	5
ОПК-2.2, ПК- 3.2, ПК-5.2, ПК-11.2	Какое максимальное число потомков может быть у узла бинарного дерева?		Средний	5

<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Для алгоритма сортировки слиянием merge-sort при каком количестве элементов в последовательности и рекурсивное деление должно прерываться, в стандартном виде?</p>	<p>1. 2 2. 1 3. 4 4. 3</p>	<p>Средний</p>	<p>5</p>
<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Сколько дополнительной памяти требуется для работы алгоритма quick-sort?</p>	<p>1. <math>O(N^2)</math> 2. <math>O(N^3)</math> 3. <math>O(N)</math> 4. алгоритм не использует дополнительную память</p>	<p>Средний</p>	<p>5</p>

<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Что означает устойчивость алгоритма сортировки?</p>	<p>1. сортировка происходит на любых данных  2. время работы алгоритма относительно стабильно при различной величине входных данных  3. процент ошибок при сортировке меньше  4. если при работе алгоритма относительный порядок пар с равными ключами не меняется</p>	<p>Средний</p>	<p>5</p>
---	--	--	----------------	----------

<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Какие высказывания относятся к структуре данных связный список?</p>	<ol style="list-style-type: none"> <li>1. в конце структуры нулевой указатель , указатель на первый элемент хранится отдельно</li> <li>2. эта структура используется для реализации стека</li> <li>3. в каждом узле содержатся указатель на следующий узел и данные</li> <li>4. время доступа к элементу константное</li> </ol>	<p>Средний</p>	<p>5</p>
---	--	---	----------------	----------

<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Какое дерево называется разбалансированным?</p>	<p>1. размеры левых и правых поддеревьев в нем сильно различаются  2. если существуют вершины-потомки, ключи которых больше ключей родителей, если в остальных вершинах это свойство не нарушено  3. если значения ключей в левом поддереве намного меньше значений ключей в правом поддереве  4. если в нем нарушен порядок неубывания ключей</p>	<p>Средний</p>	<p>5</p>
<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Бинарное дерево — это структура данных, в которой каждый узел содержит указатель на левого и правого соседа.</p>		<p>Средний</p>	<p>5</p>

<p>ОПК-2.2, ПК-3.2, ПК-5.2, ПК-11.2</p>	<p>Где будет находиться наиболее часто встречающийся символ в дереве кодирования Хаффмана?</p>	<ol style="list-style-type: none"> <li>1. на нижнем уровне дерева</li> <li>2. на верхнем уровне дерева</li> <li>3. в самой крайней правой вершине</li> <li>4. в самой крайней левой вершине</li> <li>5. может находиться в любом месте</li> </ol>	<p>Средний</p>	<p>5</p>
<p>ОПК-1.3, ОПК-2.3, ПК-5.3, ПК-3.3, ПК-11.3</p>	<p>Какие действия включает в себя операция вставки (Insert(x)) в двоичном дереве поиска?</p>	<ol style="list-style-type: none"> <li>1. поиск ключа <math>x</math> в дереве</li> <li>2. вершину <math>w</math> объявим левым сыном <math>v</math>, если <math>key(v) &gt; key(w)</math></li> <li>3. если поиск завершился неудачей, создадим новую вершину <math>w</math> с ключем <math>x</math></li> <li>4. если поиск завершился удачей, создадим новую вершину <math>w</math> с ключем <math>x</math></li> <li>5. вершину <math>w</math> объявим правым сыном <math>v</math>, если <math>key(v) &lt; key(w)</math></li> </ol>	<p>Высокий</p>	<p>8</p>



<p>ОПК-1.3, ОПК-2.3, ПК- 5.3, ПК-3.3, ПК-11.3</p>	<p>Выберите компоненты L- системы</p>	<p>1. теорема 2. набор ограничений 3. правила 4. алфавит 5. аксиома 6. множество целых чисел, называемое ключами</p>	<p>Высокий</p>	<p>8</p>
<p>ОПК-1.3, ОПК-2.3, ПК- 5.3, ПК-3.3, ПК-11.3</p>	<p>Типичный порядок полей триплета для LZ77:</p>	<p>1. next 2. offset 3. length</p>	<p>Высокий</p>	<p>8</p>
<p>ОПК-1.3, ОПК-2.3, ПК- 5.3, ПК-3.3, ПК-11.3</p>	<p>Основные проблемы, которые необходимо решать при реализации алгоритма RLE:</p>	<p>1. способность алгоритма отличать закодированные данные от исходных 2. корректная работа со скользящим окном 3. хранение таблицы символов 4. сохранение закодированных данных на диск</p>	<p>Высокий</p>	<p>8</p>

<p>ОПК-1.3, ОПК-2.3, ПК- 5.3, ПК-3.3, ПК-11.3</p>	<p>Что можно сделать для алгоритма Quick-sort, чтобы дерево рекурсии было всегда сбалансированным ?</p>	<ol style="list-style-type: none"> <li>1. увеличить количество рекурсивных вызовов для функции</li> <li>2. уменьшить число рекурсий в рекурсивной функции</li> <li>3. заменить рекурсию на цикл</li> <li>4. выбирать правильный опорный элемент (pivot)</li> </ol>	<p>Высокий</p>	<p>8</p>
---	---	--	----------------	----------