

Тестовое задание для диагностического тестирования по дисциплине:

Технологии программирования, 6 семестр

Код, направление подготовки	09.03.01 Информатика и вычислительная техника
Направленность (профиль)	Автоматизированные системы обработки информации и управления
Форма обучения	Очная
Кафедра разработчик	АСОИУ
Выпускающая кафедра	АСОИУ

Проверяемая компетенция	Задание	Варианты ответов	Тип сложности вопроса	Кол-во баллов за правильный ответ
--------------------------------	----------------	-------------------------	------------------------------	--

<p>ОПК-1.1, ОПК-2.1, ПК-3.1, ПК-5.1, ПК-11.1</p>	<p>При рассмотрении времени работы $T(M)$ и памяти $M(N)$ что нас интересует?</p>	<p>1. точный вид функций $T(N)$ и $M(N)$ 2. приближенный до константы вид функций. Используется O-символика 3. приближенный вид функций. Используется o-символика 4. точный вид функций. Используется O-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. $O(N^2)$ 2. $O(N^3)$ 3. $O(N)$ 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>	<ol style="list-style-type: none"> 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>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"> 1. в конце структуры нулевой указатель , указатель на первый элемент хранится отдельно 2. эта структура используется для реализации стека 3. в каждом узле содержатся указатель на следующий узел и данные 4. время доступа к элементу константное 	<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"> 1. на нижнем уровне дерева 2. на верхнем уровне дерева 3. в самой крайней правой вершине 4. в самой крайней левой вершине 5. может находиться в любом месте 	<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"> 1. поиск ключа x в дереве 2. вершину w объявим левым сыном v, если $key(v) > key(w)$ 3. если поиск завершился неудачей, создадим новую вершину w с ключем x 4. если поиск завершился удачей, создадим новую вершину w с ключем x 5. вершину w объявим правым сыном v, если $key(v) < key(w)$ 	<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 системы</p>	<p>L-</p> <ol style="list-style-type: none"> 1. теорема 2. набор ограничений 3. правила 4. алфавит 5. аксиома 6. множество целых чисел, называемое ключами 	<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>	<ol style="list-style-type: none"> 1. next 2. offset 3. length 	<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>	<ol style="list-style-type: none"> 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>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"> 1. увеличить количество рекурсивных вызовов для функции 2. уменьшить число рекурсий в рекурсивной функции 3. заменить рекурсию на цикл 4. выбирать правильный опорный элемент (pivot) 	<p>Высокий</p>	<p>8</p>
---	---	--	----------------	----------