

Министерство образования Тверской области
ГБПОУ «Тверской колледж им. А.Н. Коняева»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к контрольной работе по дисциплине:

«Теория алгоритмов»

для студентов заочного отделения специальности

09.02.03 «Программирование в компьютерных системах»

г.Тверь

2018

ОДОБРЕНО

цикловой комиссией

09.02.03

Председатель комиссии

_____ / Л.Л. Петрушенко

подпись

Протокол № _____

от «__» _____ 201__ г.

УТВЕРЖДАЮ

Заместитель директора по

учебной работе

_____ / Н.С.Лукина

«__» _____ 201__ г.

Организация-разработчик: ГБПОУ «Тверской колледж им.А.Н. Коняева»

Разработчик:

Петрушенко Людмила Леонидовна, преподаватель специальных дисциплин

Методические указания к выполнению контрольной работы предназначены для студентов 2 курса заочного отделения специальности 09.02.03 «Программирование в компьютерных системах». В данном пособии указаны основные требования, предъявляемые к оформлению и выполнению домашних контрольных работ. Даны задания, вопросы для подготовки к экзаменам, список рекомендуемой литературы

Содержание

| | |
|--|----|
| 1. Введение | 4 |
| 2. Методические указания к выполнению контрольной работы | 4 |
| 3. Требования к содержанию и оформлению контрольной работы | 5 |
| 4. Тест | 6 |
| 5. Теоретические вопросы | 18 |
| 6. Практические задания | 19 |
| 7. Вопросы для подготовки к экзамену | 21 |
| 8. Список рекомендуемой литературы | 22 |

Введение

Контрольная работа по дисциплине «Теория алгоритмов» составлена в соответствии с федеральными государственными образовательными стандартами по специальности среднего профессионального образования 09.02.03 «Программирование в компьютерных системах» и является формой текущего контроля по данной дисциплине.

Контрольная работа представлена в виде 4 вариантов теоретических заданий в виде теста и 10 вариантов практических заданий по изученным темам.

Данная контрольная работа позволяет оценить полученные навыки и умения:

- Разрабатывать алгоритмы для конкретных задач;
- Основные модели алгоритмов;
- Методы построения алгоритмов;
- Методы вычисления сложности алгоритмов.

Методические указания к выполнению контрольной работы

Цель контрольной работы – закрепление и проверка знаний, полученных студентами заочной формы обучения в процессе самостоятельного изучения учебного материала.

Студент заочной формы обучения должен выполнить контрольную работу в установленные графиком учебного процесса сроки.

Если работа не будет зачтена, необходимо ознакомиться с внесенными в нее поправками и замечаниями, проанализировать ошибки и выполнить ее повторно, устранив недостатки, отмеченные преподавателем в рецензии.

Требования к содержанию и оформлению контрольной работы

1. Номер варианта контрольной работы по последней цифре порядкового номера Ф.И.О. студента в списке журнала группы; цифра «0» означает вариант № 10.
2. Работа представляется в машинописном варианте. К текстам, подготовленным с помощью текстового процессора, предъявляются следующие требования: шрифт Times New Roman, 14; форматирование текста по ширине странице; междустрочный интервал 1,5; поля: верхнее – 2см, нижнее – 2 см., левое – 3 см, правое – 1 см.
3. Обязательно наличие содержания и списка литературы, оформленного в соответствии со стандартами.
4. В начале работы должен быть указан номер варианта задания. Перед ответом на вопрос должно быть приведено задание.
5. Алгоритм в виде блок-схемы оформляется в соответствии с ГОСТ 19.701-90 ЕСПД Схемы алгоритмов, программ, данных и систем.

Распределение заданий по вариантам

| № варианта | Тест | Теоретические вопросы | | Практическое задание |
|------------|------|-----------------------|----|----------------------|
| | | | | |
| 1 | 1 | 2 | 6 | 1 |
| 2 | 2 | 3 | 7 | 2 |
| 3 | 3 | 4 | 8 | 3 |
| 4 | 4 | 5 | 9 | 4 |
| 5 | 1 | 1 | 10 | 5 |
| 6 | 2 | 1 | 9 | 6 |
| 7 | 3 | 2 | 8 | 7 |
| 8 | 4 | 3 | 7 | 8 |
| 9 | 1 | 4 | 6 | 9 |
| 10 | 2 | 5 | 10 | 10 |

Тест

Вариант 1

1. Вычислительный процесс — это

- 1) последовательность действий, выполняемых исполнителем
- 2) порядок выполнения алгоритма в применении к исходным данным
- 3) последовательность шагов алгоритма
- 4) совокупность промежуточных значений переменных

2. Какие из перечисленных свойств алгоритма являются необходимыми

- 1) дискретность
- 2) результативность
- 3) массовость
- 4) детерминированность
- 5) рекурсивность
- 6) формальность и простота
- 7) эффективность

3. Алгоритм – это

- 1) понятное и точное описание конечной последовательности команд, приводящей от исходных данных к искомому результату
- 2) пошаговое описание процесса решения какой-либо задачи
- 3) последовательность действий, применяемая к некоторым исходным данным

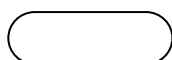
4. У машины Поста существует команд

- 1) 3
- 2) 5
- 3) 6
- 4) 9

5. Если ребра графа определяются упорядоченными парами вершин, то граф называется

- 1) семантическим
- 2) ориентированным
- 3) простым
- 4) циклом

6. Конструкция на рисунке называется



- 1) выполнение операций
- 2) начало-конец алгоритма
- 3) вызов вспомогательного алгоритма
- 4) ввод-вывод данных

7. Информационной моделью является:

- 1) модель автомобиля
- 2) сборник правил дорожного движения
- 3) формула закона всемирного тяготения
- 4) номенклатура списка товаров на складе

8. Рекурсия в алгоритме будет прямой, когда:

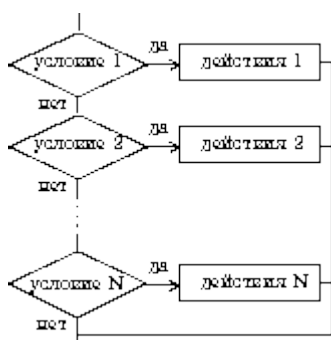
- 1) рекурсивный вызов данного алгоритма происходит из вспомогательного алгоритма, к которому в данном алгоритме имеется обращение
- 2) порядок следования команд определяется в зависимости от результатов проверки некоторых условий
- 3) команда обращения алгоритма к самому себе находится в самом алгоритме
- 4) один вызов алгоритма прямо следует за другим

9. Фрагмент алгоритма:

$s := x[1]$; *нц* для i от 1 до n ; *если* ($s > x[i]$) *то* $s := x[i]$; *все*; *кц*;

определяет

- 1) минимальный элемент массива чисел $X = \{x[i], i=1, 2, \dots, n\}$
- 2) сумму отрицательных элементов X
- 3) индекс минимального элемента X
- 4) максимальный элемент X

10. На рисунке представлена часть блок-схемы

- 1) если-то
- 2) выбор-иначе
- 3) если-то-иначе
- 4) выбор

Вариант 2

1. Свойство, означающее, что алгоритм всегда приводит к результату через конечное число шагов, соответствует

- 1) результативности
- 2) дискретности
- 3) определенности
- 4) массовости

2. Исходные данные — это

- 1) точно определенное множество значений, с которых начинается выполнение алгоритма
- 2) переменные и константы, которые используются в алгоритме
- 3) множество возможных значений переменных
- 4) набор всех переменных алгоритма и их значений

3. Алгоритм называется циклическим,

- 1) если он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий
- 2) если ход его выполнения зависит от истинности тех или иных условий
- 3) если его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий
- 4) если он представим в табличной форме
- 5) если он включает в себя вспомогательный алгоритм

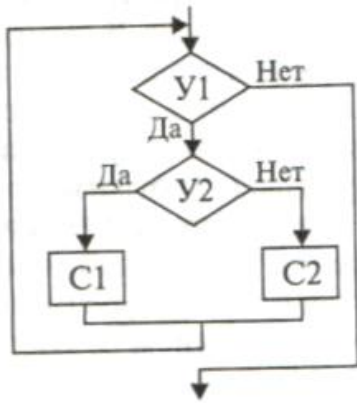
4. Цепь графа – это

- 1) если все определяемые маршрутом ребра смежные
- 2) если ребра в маршруте не образуют петель
- 3) маршрут, в котором все определяемые им ребра различны
- 4) если граф простой

5. В машине Поста останов будет результативным

- 1) при выполнении недопустимой команды
- 2) если машина не останавливается никогда
- 3) если результат выполнения программы такой, какой и ожидался
- 4) по команде «Стоп»

6. Определите название блок-схемы:



- 1) вложенные ветвления
- 2) цикл-пока с вложенным ветвлением
- 3) вложенные циклы-пока
- 4) следование ветвления и цикла-до
- 5) вложенные циклы: внешний – цикл-пока, внутренний – цикл-до

7. Математической моделью является:

- 1) модель автомобиля
- 2) сборник правил дорожного движения
- 3) формула закона всемирного тяготения
- 4) номенклатура списка товаров на складе

8. Рекурсия в алгоритме будет косвенной, когда:

- 1) рекурсивный вызов данного алгоритма происходит из вспомогательного алгоритма, к которому в данном алгоритме имеется обращение

- 2) порядок следования команд определяется в зависимости от результатов проверки некоторых условий
- 3) команда обращения алгоритма к самому себе находится в самом алгоритме
- 4) один вызов алгоритма прямо следует за другим

9. Фрагмент алгоритма:

$s:=0$; *нц* для i от 1 до n ; $s:=s+i$; *кц*

находит сумму

- 1) натуральных последовательных чисел
- 2) четных последовательных натуральных чисел
- 3) соседних вещественных чисел
- 4) нечетных последовательных натуральных чисел

10. Идея алгоритма сортировки массива прямым выбором заключается в том, что

- 1) на каждом шаге просмотру подвергается несортированная правая часть массива. В ней ищется очередной максимальный элемент. Если он найден, то производится его перемещение на место крайнего левого элемента несортированной правой части массива
- 2) на каждом шаге просмотру подвергается несортированная правая часть массива. В ней ищется очередной минимальный элемент. Если он не найден, то производится его перемещение на место крайнего левого элемента несортированной правой части массива
- 3) на каждом шаге просмотру подвергается несортированная правая часть массива. В ней ищется очередной минимальный элемент. Если он найден, то производится его перемещение на место крайнего правого элемента несортированной левой части массива
- 4) на каждом шаге просмотру подвергается несортированная правая часть массива. В ней ищется очередной минимальный элемент. Если он найден, то производится его перемещение на место крайнего левого элемента несортированной правой части массива

Вариант 3

1. Алгоритм называется линейным,

- 1) если он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий
- 2) если ход его выполнения зависит от истинности тех или иных условий
- 3) если его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий
- 4) если он представим в табличной форме
- 5) если он включает в себя вспомогательный алгоритм

2. Петля в графе будет, если

- 1) начала и концы ребер совпадают
- 2) цепь из этих ребер замкнута
- 3) их вершины соединены двумя и более ребрами
- 4) они концевые

3. Команда машины Поста имеет структуру $n K m$, где:

- 1) n — действие, выполняемое головкой; K — номер следующей команды, подлежащей выполнению; m — порядковый номер команды
- 2) n — порядковый номер команды; K — действие, выполняемое головкой; m — номер следующей команды, подлежащей выполнению
- 3) n — порядковый номер команды; K — номер следующей команды, подлежащей выполнению; m — действие, выполняемое головкой
- 4) n — порядковый номер команды; K — действие, выполняемое головкой; m — номер клетки, с которой данную команду надо произвести

4. Свойство алгоритма, заключающееся в том, что один и тот же алгоритм можно использовать с разными исходными данными, называется:

- 1) результативность
- 2) детерминированность

- 3) конечность
- 4) массовость
- 5) дискретность

5. К стохастическим моделям относятся:

- 1) модель движения тела, брошенного под углом к горизонту
- 2) модель броуновского движения
- 3) модель таяния кусочка льда в стакане
- 4) модель обтекания газом крыла самолета

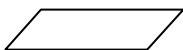
6. Компьютерное моделирование — это:

- 1) процесс построения модели компьютерными средствами
- 2) процесс исследования объекта с помощью его компьютерной модели
- 3) построение модели на экране компьютера
- 4) решение конкретной задачи с помощью компьютера

7. Что называют служебными словами в алгоритмическом языке:

- 1) слова, употребляемые для записи команд, входящих в СКИ
- 2) слова, смысл и способ употребления которых задан раз и навсегда
- 3) вспомогательные алгоритмы, которые используются в составе других алгоритмов
- 4) константы с постоянным значением

8. Конструкция на рисунке называется



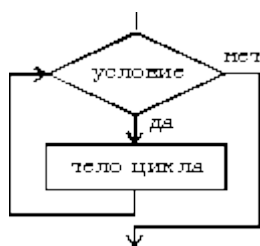
- 1) выполнение операций
- 2) начало-конец алгоритма
- 3) вызов вспомогательного алгоритма
- 4) ввод-вывод данных

9. Фрагмент алгоритма:

$h:=1$; *нц* для i от 1 до n ; $h:=h*i$; *кц*

находит

- 1) произведение натуральных последовательных чисел
- 2) произведение четных натуральных чисел
- 3) произведение попарно соседних натуральных чисел
- 4) произведение нечетных натуральных чисел

10. На рисунке представлена часть блок-схемы

- 1) если-то
- 2) цикл-для
- 3) цикл-пока
- 4) если-то-иначе

Вариант 4

1. Алгоритм включает в себя ветвление,

- 1) если он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий
- 2) если ход его выполнения зависит от истинности тех или иных условий
- 3) если его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий
- 4) если он представим в табличной форме
- 5) если он включает в себя вспомогательный алгоритм

2. Дерево – это

- 1) неориентированный связанный граф
- 2) ориентированный несвязный граф
- 3) граф со смежными вершинами
- 4) ориентированный связный граф

3. В машине Поста некорректным алгоритм будет в случае

- 1) при выполнении недопустимой команды
- 2) если машина не останавливается никогда
- 3) если результат выполнения программы такой, какой и ожидался
- 4) по команде «Стоп»

4. Суть такого свойства алгоритма, как *понятность*, заключается в том, что:

- 1) алгоритм всегда состоит из последовательности дискретных шагов
- 2) для записи алгоритма используются команды, которые входят в систему команд исполнителя
- 3) алгоритм обеспечивает решение не одной конкретной задачи, а некоторого класса задач

- 4) при точном исполнении всех команд алгоритма процесс должен прекратиться за конечное число шагов и привести к определенному результату
- 5) алгоритм должен состоять из команд, однозначно понимаемых исполнителем

5. К детерминированным моделям относится:

- 1) модель случайного блуждания частицы
- 2) модель формирования очереди
- 3) модель свободного падения тела в среде с сопротивлением
- 4) модель игры «орел—решка»

6. Компьютерный эксперимент — это:

- 1) решение задачи на компьютере
- 2) исследование модели с помощью компьютерной программы
- 3) подключение компьютера для обработки физических экспериментов
- 4) автоматизированное управление физическим экспериментом

7. Конструкция на рисунке называется



- 1) выполнение операций
- 2) начало-конец алгоритма
- 3) вызов вспомогательного алгоритма
- 4) ввод-вывод данных

8. Значение s после выполнения фрагмента алгоритма:

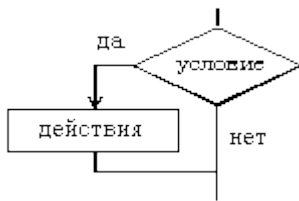
$s:=0; i:=5; \text{нц пока } (i>2); i:=i-1; s:=s+i; \text{кц}$

равно

- 1) 0
- 2) 5
- 3) 14

4) 9

9. На рисунке представлена часть блок-схемы



- 1) цикл-пока
- 2) цикл-для
- 3) если-то
- 4) если-то-иначе

10.Идея алгоритма сортировки массива прямым обменом заключается в том, что

- 1) если номер позиции большего из элементов больше номера позиции меньшего элемента, то меняем их местами
- 2) если номер позиции меньшего из элементов больше номера позиции большего элемента, то не меняем их местами
- 3) если номер позиции меньшего из элементов больше номера позиции большего элемента, то оставляем их на месте
- 4) если номер позиции меньшего из элементов больше номера позиции большего элемента, то меняем их местами

Теоретические вопросы

Ответ на вопрос должен содержать от 2 до 5 страниц текста

1. Неформальное понятие алгоритма. Алгоритм как формальная математическая система. Основные требования к алгоритмам. Формы представления алгоритмов.
2. Эффективность, сходимость, сложность, надежность алгоритмов.
3. Абстрактные машины. Система команд. Примеры схем машины Поста и Тьюринга. Вычислимые по Тьюрингу функции. Основная гипотеза теории алгоритмов
4. Рекурсивные функции, примитивно-рекурсивные функции и операторы, схемная интерпретация примитивной рекурсии, частично рекурсивные и общерекурсивные функции. Тезис Черча.
5. Нормальные алгоритмы Маркова. Способы композиции нормальных алгоритмов Маркова;
6. Основные понятия структурного программирования. Использование метода пошаговой детализации при проектировании структуры программного обеспечения
7. Понятие рекурсии. Глубина рекурсии. Рекурсивные методы.
8. Задачи поиска по критерию. Полный перебор. Перебор с возвратом. Понятие эвристики. Эвристические методы в программировании.
9. Сортировка данных. Алгоритмы сортировки элементов массива.
10. Сложность алгоритмов. Временная сложность алгоритма. Объемная сложность алгоритма. Оценка порядка. Определение сложности алгоритмов.

Практические задания

Разработать алгоритм задачи в виде блок-схемы. Указать описание входных и выходных данных. Выполнить трассировку.

1. Сформировать массив $A[1..10]$, элементы которого выбираются случайным образом из интервала $[10,90]$. Определить методом последовательного поиска, содержит ли он заданное число. Если элемент найден, то удалить его из массива.

2. Сформировать массив $A[1..18]$, элементы которого выбираются случайным образом из интервала $[-20, 40]$. Определить методом последовательного поиска, содержит ли он заданное число. Если элемент не найден, то вставить его на последнее место.

3. Сформировать массив $A[1..12]$, упорядоченный по возрастанию. Методом бинарного поиска определить, содержит ли он заданное число. Если элемент найден, то удалить его из массива.

4. Сформировать массив $A[1..10]$, упорядоченный по убыванию. Методом бинарного поиска определить, содержит ли он заданное число. Если элемент не найден, то вставить его в массив на второе место.

5. Задан массив $B[1..20]$. Отсортировать все элементы, стоящие на нечетных местах по невозрастанию.

6. Задан массив $B[1..30]$. Отсортировать все элементы с n -го по k -ый по неубыванию.

7. Задан массив $B[1..20]$, элементы которого выбираются случайным образом из отрезка $[30, 100]$. Отсортировать элементы с 1-го по 10-ый по невозрастанию, а с 11-го по 20-й - по неубыванию.

8. Сформировать массив $A[1..12]$, упорядоченный по возрастанию. Методом бинарного поиска определить, содержит ли он заданное число. Если элемент найден, то удалить его и следующий за ним элемент из массива.

9. Задан массив $X[1..20]$, элементы которого выбираются случайным образом из отрезка $[-30, 80]$. Отсортировать все элементы с 5-го по 15-й по невозрастанию.

10. Сформировать массив $a[1..10]$, упорядоченный по возрастанию. Методом бинарного поиска определить, содержит ли он заданное число. Если элемент не найден, то вставить его в массив на первое место.

Вопросы для подготовки к экзамену

по дисциплине «Теория алгоритмов»

1. Неформальное понятие алгоритма. Алгоритм как формальная математическая система.
2. Основные требования к алгоритмам.
3. Формы представления алгоритмов.
4. Эффективность, сходимость, сложность, надежность алгоритмов
5. Машина Поста. Примеры схем машины Поста.
6. Рекурсивные функции, примитивно-рекурсивные функции и операторы.
7. Нормальные алгорифмы Маркова. Способы композиции нормальных алгорифмов Маркова;
8. Основные понятия структурного программирования. Использование метода пошаговой детализации при проектировании структуры программного обеспечения
9. Понятие рекурсии. Глубина рекурсии. Рекурсивные методы.
10. Задачи поиска по критерию. Полный перебор. Перебор с возвратом.
11. Базовые алгоритмические структуры.
12. Сортировка данных Алгоритмы сортировки.
13. Сложность алгоритмов. Временная сложность алгоритма. Объемная сложность алгоритма.
14. Определение сложности алгоритмов.

Список рекомендуемой литературы

Основные источники:

1. Кормен, Т. Х. Алгоритмы: построение и анализ / Т. Х. Кормен, Ч. И. Лейзерсон, Р. Л. Ривест, К. Штайн — М.: Вильямс, 2012. – 1296 с..
2. Кнут, Д. Э. Искусство программирования, том 1. Основные алгоритмы / Д. Э. Кнут. — М.: «Вильямс», 2010. – 720 с.
3. Семакин, И. Г. Основы программирования / И. Г. Семакин, А. П. Шестаков. – М.: Издательский центр «Академия», 2006. – 432 с.
4. Игошин В.И. Теория алгоритмов: учеб.пособие для студентов учреждений СПО / В.И.Игошин. – М.:Издательский центр «Академия», 2013 – 320с.

Дополнительные источники:

1. Андреева, Е. В. Математические основы информатики. Элективный курс: учебное пособие / Е. Л. Андреева, Л. Л. Босова, И. Н. Фалина. – М.: БИНОМ. Лаборатория знаний, 2012. – 312 с..
2. Грэхем, Р. Л., Конкретная математика. Математические основы информатики / Р. Л. Грэхем, Д. Э. Кнут, О. Паташник. – М.: Вильямс, 2010. – 784 с.