

Входной контроль

1. Какой из документов является алгоритмом?
 - а) Правила техники безопасности.
 - б) Инструкция по получению денег в банкомате.
 - в) Расписание уроков.

2. Программы, которые содержат команду повторения, называются ...
 - а) линейными
 - б) разветвляющимися;
 - в) циклическими;
 - г) вспомогательными?

3. Программы, в которых команды выполняются последовательно друг за другом, называются ...
 - а) линейными;
 - б) разветвляющимися;
 - в) циклическими;
 - г) вспомогательными?

4. Какой тип алгоритмической структуры необходимо применить, если последовательность команд выполняется или не выполняется в зависимости от условия
 - а) цикл
 - б) ветвление
 - в) линейный.

5. Графический способ описания алгоритма – это ...
 - а) программа;
 - б) блок-схема;
 - в) алгоритм;
 - г) словесно-пошаговая запись?

Контрольная работа

Вариант 1

1. Опишите развитие понятия алгоритм
2. Что означает свойство алгоритма: Дискретность
3. Приведите классификацию алгоритмов по сложности
4. Что такое полиномиальный алгоритм
5. Назовите критерии выбора конкретного численного метода решения задачи

Вариант 2

1. Опишите классическую теорию алгоритмов
2. Что такое вычислительные алгоритмы
3. Что такое быстрые алгоритмы
4. Что такое структурное программирование
5. Опишите нисходящее проектирование

Вариант 3

1. Опишите три класса алгоритмов
2. Что означает свойство алгоритма: Конечность (результативность)
3. Что такое Информационные алгоритмы
4. Общие требования к алгоритмическим машинам
5. Цель структурного программирования

Вариант 4

1. Напишите три определения алгоритма
2. Что означает свойство алгоритма: Детерминированность (определенность)
3. Опишите теорию практического анализа вычислительных алгоритмов
4. Что такое Недетерминированные алгоритмы
5. Три основных направления в подходах к определению понятия алгоритм

Вариант 5

1. Что означает свойство алгоритма: Понятность
2. Опишите Теорию асимптотического анализа алгоритмов
3. Что такое Управляющие алгоритмы
4. Что такое Модульное программирование
5. Опишите восходящее проектирование

Вариант 6

1. Свойства алгоритма (перечислить)
2. Приведите примеры трудных задач
3. Что такое Детерминированные алгоритмы
4. Опишите метод пошагового уточнения
5. Направления в теории алгоритмов (перечислить)

Тест
по темам «Машина Поста», "Машина Тьюринга", "Нормальные алгоритмы Маркова"

1. Команда машины Поста имеет структуру nKm , где:
 - n - действие, выполняемое головкой; K - номер следующей команды, подлежащей выполнению; m - порядковый номер команды
 - n - порядковый номер команды; K - действие, выполняемое головкой; m - номер следующей команды, подлежащей выполнению
 - n - порядковый номер команды; K - номер следующей команды, подлежащей выполнению; m - действие, выполняемое головкой
 - n - порядковый номер команды; K - действие, выполняемое головкой; m - номер клетки, с которой данную команду надо произвести

2. Сколько существует команд у машины Поста?
 - 2
 - 4
 - 6
 - 8

3. В машине Поста останов будет результативным:
 - При выполнении недопустимой команды
 - Если машина не останавливается никогда
 - Если результат выполнения программы такой, какой и ожидался
 - По команде "Стоп"

4. В машине Поста некорректным алгоритм будет в следующем случае:
 - При выполнении недопустимой команды
 - Результат выполнения программы такой, какой и ожидался
 - Машина не останавливается никогда
 - По команде "Стоп"

5. В машине Тьюринга предписание L для лентопротяжного механизма означает:
 - Переместить ленту вправо
 - Переместить ленту влево
 - Остановить машину
 - Занести в ячейку символ

6. В машине Тьюринга предписание R для лентопротяжного механизма означает:
 - Переместить ленту вправо
 - Переместить ленту влево
 - Остановить машину
 - Занести в ячейку символ

7. В машине Тьюринга предписание S для лентопротяжного механизма означает:
 - Переместить ленту вправо
 - Переместить ленту влево
 - Остановить машину
 - Занести в ячейку символ

8. В алгоритме Маркова ассоциативным исчислением называется:
 - Совокупность всех слов в данном алфавите
 - Совокупность всех допустимых подстановок

Совокупность всех слов в данном алфавите вместе с допустимой системой подстановок

Когда все слова в алфавите являются смежными

9. В ассоциативном исчислении два слова называются смежными:

Если одно из них может быть преобразовано в другое применением подстановок

Когда существует цепочка от одного слова к другому и обратно

Когда они дедуктивны

Если одно из них может быть преобразовано в другое однократным применением допустимой подстановки

10. В алгоритме Маркова дана цепочка $P = P_1, P_2, \dots, P_n$. Если слова P_1, P_2, \dots, P_n смежные, то цепочка называется:

Ассоциативной

Эквивалентной

Индуктивной

Дедуктивной

11. В алгоритме Меркова дана цепочка $P = P_1, P_2, \dots, P_k$. Если слова P_1, P_2, \dots, P_k смежные и цепочка существует и в обратную сторону, то слова P_1 и P_k называют:

Ассоциативными

Эквивалентными

Индуктивными

Дедуктивными

12. В алгоритмах Маркова дана система подстановок в алфавите $L = \{a, b, c\}$: $abc - c$; $ba - cb$; $ca - ab$. Преобразуйте с помощью этой системы слово $bacaabc$

cbc

$ccbcbbc$

$cbacba$

$cbabc$

13. В алгоритмах Маркова дана система подстановок в алфавите $A = \{a, b, c\}$: $cb - abc$; $bac - ac$; $sab - b$. Преобразуйте с помощью этой системы слово $bcaab$:

ccb

sab

cbc

$bcaab$

14. Способ композиции нормальных алгоритмов будет суперпозицией, если:

Существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B

Выходное слово первого алгоритма является входным для второго

Алгоритм D будет суперпозицией трех алгоритмов ABC , причем область определения D является пересечением областей определения алгоритмов A , B и C , а для любого слова p из этого пересечения $D(p) = A(p)$, $C(p) = e$, $D(p) = B(p)$, если $C(p) = e$, где e - пустая строка

Существует алгоритм C , являющийся суперпозицией алгоритмов A и D такой, что для любого входного слова p $C(p)$ получается в результате последовательного многократного применения алгоритма A до тех пор, пока не получится слово, преобразуемое алгоритмом B

15. Способ композиции нормальных алгоритмов будет объединением, если:

Входное слово первого алгоритма является входным для второго

Существует алгоритм C , преобразующий любое слово p , содержащееся в пересечении областей определения алгоритмов A и B

Алгоритм В будет суперпозицией трех алгоритмов АВС, причем область определения Д является пересечением областей определения алгоритмов А В и С, а для любого слова р из этого пересечения $D(p)=A(p)$, $C(p)=e$, $D(p)=B(p)$, если $C(p)=e$, где е - пустая строка

Существует алгоритм С, являющийся суперпозицией алгоритмов А и Д такой, что для любого входного слова р $C(p)$ получается в результате последовательного многократного применения алгоритма А до тех пор, пока не получится слово, преобразуемое алгоритмом В

16. Способ композиции нормальных алгоритмов будет разветвлением, если:

Выходное слово первого алгоритма является входным для второго

Существует алгоритм С, преобразующий любое слово р, содержащееся в пересечении областей определения алгоритмов А и В

Алгоритм Д будет суперпозицией трех алгоритмов АВС, причем область определения Д является пересечением областей определения алгоритмов А В и С, а для любого слова р из этого пересечения $D(p)=A(p)$, если $C(p)=e$, $D(p)=B(p)$, если $C(p)=e$, где е - пустая строка

Существует алгоритм С, являющийся суперпозицией алгоритмов А и В, такой, что для любого входного слова р $C(p)$ получается в результате последовательного многократного применения алгоритма А до тех пор, пока не получится слово, преобразуемое алгоритмом В

17. Способ композиции нормальных алгоритмов будет итерацией, если:

Выходное слово первого алгоритма является входным для второго

Существует алгоритм С, преобразующий любое слово р, содержащееся в пересечении областей определения алгоритмов А и В

Алгоритм Д будет суперпозицией трех алгоритмов АВС, причем область определения Д является пересечением областей определения алгоритмов А В С, а для любого слова р из этого пересечения $D(p)=A(p)$, если $C(p)=e$, $D(p)=B(p)$, если $C(p)=e$, где е - пустая строка

Существует алгоритм С, являющийся суперпозицией алгоритмов А и В, такой, что для любого входного слова р $C(p)$ получается в результате последовательного многократного применения алгоритма А до тех пор, пока не получится слово, преобразуемое алгоритмом В