

I-bob. BOI, UAH [1-40] (40)

1. Алгоритмом называется,...

=====

конечная последовательность шагов, которые должны быть выполнены в соответствии с определенным правилом для решения данной проблемы.

=====

алгоритм называется точным в инструкциях, данных исполнителю.

=====

алгоритмы можно разбить на конечные шаги.

=====

говорят, что выполненный алгоритм дает результат за конечные шаги.

+++++

2. Постижимость алгоритма -

=====

инструкции в алгоритме должны быть понятны исполнителю

=====

возможность разбивать алгоритмы на конечные шаги

=====

выполняемый алгоритм приводит к конечным шагам

=====

относиться ко всем задачам одного типа в соответствии с содержанием каждого алгоритма

+++++

3. Дискретное свойство алгоритма -

=====

возможность разбивать алгоритмы на конечные шаги

=====

инструкции в алгоритме должны быть понятны исполнителю

=====

выполняемый алгоритм приводит к конечным шагам

=====

инструкции в алгоритме должны быть понятны исполнителю

+++++

4. Свойство ограниченности алгоритма -

=====

выполняемый алгоритм приводит к конечным шагам

=====

инструкции в алгоритме должны быть понятны исполнителю

=====

возможность разбивать алгоритмы на конечные шаги

=====

относиться ко всем задачам одного типа в соответствии с содержанием каждого алгоритма

+++++

5. Общеизвестная принадлежность алгоритма

=====

относиться ко всем задачам одного типа в соответствии с содержанием каждого алгоритма

=====

инструкции в алгоритме должны быть понятны исполнителю

=====

выполняемый алгоритм приводит к конечным шагам

=====

возможность разбивать алгоритмы на конечные шаги

++++

6. Формальное свойство алгоритма -

=====

возможность механически выполнять команды

=====

инструкции в алгоритме должны быть понятны исполнителю

=====

выполняемый алгоритм приводит к конечным шагам

=====

возможность разбивать алгоритмы на конечные шаги

++++

7. Формальное свойство алгоритма -

=====

обеспечивает выполнение команд на роботах, компьютерах и других устройствах

=====

гарантирует, что инструкции, данные исполнителю в алгоритме, имеют четкое значение

=====

выполняемый алгоритм обеспечивает достижение результата за конечные шаги

=====

позволяет разбивать алгоритмы на конечные шаги

++++

8. Линейный алгоритм ...

=====

безусловно относится только к последовательно выполняемым процессам

=====

относится к процессам, которые происходят, когда условие проверяется или повторяется много раз в алгоритме на основе разных значений параметра.

=====

алгоритм, состоящий из инструкций, которые выполняются при определенных условиях

=====

безусловно относится только к последовательно выполняемым процессам

++++

9. Алгоритм ветвления ...

=====

алгоритм, состоящий из инструкций, которые выполняются при определенных условиях

=====

относится к процессам, которые происходят, когда условие проверяется или повторяется много раз в алгоритме на основе разных значений параметра.

=====

безусловно относится только к последовательно выполняемым процессам

=====

относится к конечной последовательности действий, выполняемых в соответствии с определенным правилом для решения проблемы.

+++++

10. Итерационный алгоритм ...

=====

относится к процессам, которые происходят, когда условие проверяется или повторяется много раз в алгоритме на основе разных значений параметра.

=====

алгоритм, состоящий из инструкций, которые выполняются при определенных условиях

=====

безусловно относится только к последовательно выполняемым процессам

=====

относится к конечной последовательности действий, выполняемых в соответствии с определенным правилом для решения проблемы.

+++++

11. Если итерационные алгоритмы зависят от более чем одного параметра, как они называются?

=====

вложенные циклические алгоритмы

=====

линейные алгоритмы

=====

алгоритмы ветвления

=====

алгоритмы приближения ряда

+++++

12. Как называются алгоритмы саморегулирования?

=====

называются рекурсивными алгоритмами

=====

вложенными циклическими алгоритмами

=====

линейными алгоритмами

=====

алгоритмами ветвления

+++++

13. Определение интервалов при приближенном решении алгебраических и трансцендентных уравнений.

=====

Если в интервале $[a, b]$ функция $y = f(x)$ непрерывна и $f(a) \cdot f(b) < 0$, то в этом интервале есть хотя бы один корень уравнения $f(x) = 0$.

=====

Если уравнение $f(x) = 0$ и $f(a) \cdot f(b) < 0$ в данном интервале $[a, b]$, то существует несколько решений уравнения в этом интервале.

=====

Если в интервале $[a, b]$ функция $y = f(x)$ непрерывна и $f(a) \cdot f(b) > 0$, то есть хотя бы один корень уравнения $f(x) = 0$ в этом интервал.

=====

Если в интервале $[a, b]$ функция $y = f(x)$ непрерывна и $f(a) \cdot f(b) < 0$, то в этом интервале существует единственный корень уравнения $f(x) = 0$.

+++++

14. Формула для нахождения приближенных решений уравнений методом Ньютона.

=====

$$X_{n+1} = X_n - f(X_n) / f'(X_n)$$

=====

$$X_{n+1} = X_n - f(X_n) \cdot (b - X_n) / (f(b) - f(X_n))$$

=====

$$X_{n+1} = X_n - f(X_n) \cdot (X_n - b) / (f(X_n) - f(a))$$

=====

$$X_{n+1} = X_n + f(X_n) \cdot (b - X_n) / (f(b) - f(X_n)).$$

+++++

15. Формула для нахождения приближенного решения уравнений методом хорд.

=====

$$X_{n+1} = X_n - f(X_n) \cdot (b - X_n) / (f(b) - f(X_n))$$

=====

$$X_{n+1} = X_n - f(X_n) \cdot (X_n - b) / (f(X_n) - f(a))$$

=====

$$X_{n+1} = X_n - f(X_n) / f'(X_n)$$

=====

$$X_{n+1} = X_n + f(X_n) \cdot (b - X_n) / (f(X_n) - f(b))$$

+++++

16. Проект - это...

=====

это процесс решения проблемы и достижения цели

=====

это деятельность, которая предполагает достижение поставленной цели с использованием времени, капитала и трудовых ресурсов.

=====

постоянно переходить от простого к сложному и решать одну большую проблему, решая несколько простых задач

=====

состоит в том, чтобы следовать процессу логической последовательности фаз или этапов

++++

17. Проект - это...

=====

набор целевых действий над связанными объектами

=====

это деятельность, которая предполагает достижение поставленной цели с использованием времени, капитала и трудовых ресурсов.

=====

постоянно переходить от простого к сложному и решать одну большую проблему, решая несколько простых задач

=====

состоит в том, чтобы следовать процессу логической последовательности фаз или этапов

++++

18. Проект - это...

=====

ограниченный набор задач, предназначенных для достижения четко определенного результата на основе бюджета, выделенного на определенный период времени

=====

это деятельность, которая предполагает достижение поставленной цели с использованием времени, капитала и трудовых ресурсов.

=====

постоянно переходить от простого к сложному и решать одну большую проблему, решая несколько простых задач

=====

состоит в том, чтобы следовать процессу логической последовательности фаз или этапов

++++

19. Проект...

=====

инициатива, характеризующаяся проблемой, сроками, ценой и условиями качества

=====

деятельность по обеспечению достижения поставленной цели с использованием времени, капитала и трудовых ресурсов

=====

постоянно переходить от простого к сложному и решать одну большую проблему, решая несколько простых задач

=====

состоит в том, чтобы следовать процессу логической последовательности фаз или этапов

++++

20. Управление проектами

=====

это деятельность, которая предполагает достижение поставленной цели с использованием времени, капитала и трудовых ресурсов.

=====

это процесс решения проблемы и достижения цели

=====

набор целевых действий над связанными объектами

=====

представляет собой ограниченный набор задач, предназначенных для достижения четко определенного результата на основе бюджета, выделенного на определенный период времени

+++++

21. Управление проектами

=====

решение ряда мелких вопросов на разных этапах проекта

=====

это процесс решения проблемы и достижения цели

=====

набор целевых действий над связанными объектами

=====

представляет собой ограниченный набор задач, предназначенных для достижения четко определенного результата на основе бюджета, выделенного на определенный период времени

+++++

22. В основе управления проектом

=====

лежит краткосрочное или долгосрочное планирование

=====

процесс решения проблемы и достижения цели

=====

лежит набор целевых действий над связанными объектами

=====

лежит ограниченный набор задач, формируемых для достижения четко определенного результата на основе бюджета, выделенного на данный период времени,

+++++

23. Управление проектами

=====

это постоянно переходить от простого к сложному и решать одну большую проблему, решая несколько простых задач

=====

это процесс решения проблемы и достижения цели

=====

набор целевых действий над связанными объектами

=====

данное время

+++++

24. Планирование процессов -

=====

основывается на методах планирования, зависящих от приоритетности поставленной задачи и сроков реализации

=====

основывается на методах формирования целевого набора действий над связанными объектами

=====

это процесс решения проблемы и достижения цели

=====

постоянно переходить от простого к сложному и решать одну большую проблему, решая несколько простых задач

++++

25. Небольшой проект - это

=====

простой и ограниченный проект небольшого размера

=====

отдельный проект разного типа, конструкции и размера

=====

краткосрочный и среднесрочный проект

=====

проект, который взаимосвязан с точки зрения целей, ресурсов и времени

++++

26. Мегапроекты -

=====

включают несколько проектов, связанных между собой общими целями, ресурсами и временем

=====

организационный, экономический, социальный и смешанный проекты

=====

включают в себя отдельные проекты разного типа, конструкции и размера

=====

инвестиции, инновации, исследования, смешанный проект

++++

27. Проекты по срокам внедрения -

=====

разделены на краткосрочные, средние и долгосрочные проекты

=====

разделены на маленькие, средние, большие, очень большие проекты

=====

разделены на простые, сложные, очень сложные проекты

=====

разделены на инвестиционные, инновационные, исследовательские и смешанные проекты

++++

28. По направлениям реализации проекта -

=====

разделены на организационные, экономические, социальные и смешанные проекты

=====

разделены на монопроекты, мультипроекты, мегапроекты

=====

разделены на проекты разного типа, конструкции и размера

=====

разделены на инвестиции, инновации, исследования, смешанные проекты

++++

29. Проекты по структуре и системе -

=====

разделены на монопроекты, мультипроекты, мегапроекты

=====

разделены на маленькие, средние, большие, очень большие проекты

=====

разделены на краткосрочные, среднесрочные и долгосрочные проекты

=====

разделены на инвестиционные, инновационные, исследовательские проекты

++++

30. Монопроект это

=====

это отдельный проект разного типа, конструкции и размера

=====

это маленький, средний, большой, очень большой проект

=====

это краткосрочный, среднесрочный, долгосрочный проект

=====

это простой, сложный, очень сложный проект

++++

31. Проекты по размеру проекта, количеству участников и уровню воздействия на окружающую среду:

=====

разделены на маленькие, средние, большие, очень большие проекты

=====

разделены на краткосрочные, среднесрочные и долгосрочные проекты

=====

монопроект, мультипроект, мегапроект

=====

разделены на простые, сложные, очень сложные проекты

++++

32. По продолжительности проекты:

=====

разделены на краткосрочные и среднесрочные проекты

=====

разделены на малые, средние и большие проекты

=====

монопроект, мультипроект, мегапроект

=====

разделены на простые, сложные, очень сложные проекты

++++

33. По сложности проекты:

=====

разделены на простые, сложные, очень сложные проекты

=====

разделены на маленькие, большие, сложные проекты

=====

разделены на краткосрочные, среднесрочные и долгосрочные проекты

=====

монопроект, мультипроект, мегапроект

++++

34. По характеру отрасли науки проекты делятся на :

=====

инвестиционные, инновационные, научно- исследовательские, смешанные

=====

организационные, экономические, социальные и смешанные

=====

монопроекты, мультипроекты, мегапроекты

=====

это отдельный проект разного типа, конструкции и размера

++++

35. Этапы жизненного цикла проекта:

=====

прединвестиционный, инвестиционный, эксплуатационный

=====

инвестиции, инновации, исследования

=====

организационные, экономические, социальные

=====

монопроекты, мультипроекты, мегапроекты

++++

36. Какая система называется статической?

=====

если наблюдается изменение состояния системы с течением времени

=====

устойчивая к внутреннему и внешнему сопротивлению

=====

если состояние системы меняется во процессе выполнения

=====

если состояние системы не меняется со временем

++++

37. Сколько существует типов блок-схем алгоритмов?

=====

3

=====

2

=====

4

=====

1

++++

38. Какой алгоритм использует блок-схему для поиска поверхности треугольника со сторонами a, b, c.

=====

разветвления

=====

повторяющийся

=====

прямой линии

=====

все ответы верны

++++

39. Понятие проектирования демонстрируют

=====

объекты (новые типы и образцы оборудования и устройств, здания и сооружения, дороги и мосты, машины и оборудование, самолеты и космические корабли, радиоприемники и телевизоры, телефоны и компьютеры и различные другие продукты) процесс составления и составления своих проектов для строительства и Создайте.

=====

автоматизация ввода данных.

=====

ввод и вывод данных

=====

объекты (аппаратура и инструменты, здания и сооружения, дороги и мосты, машины и оборудование, самолеты и космические корабли, радио и телевизоры, телефоны и компьютеры и т. д.)

++++

40. Показать итерационные методы приближенного решения линейных уравнений

=====

расщепление

=====

Лобачевский, McLaren

=====

Лагранж, Ньютон

=====

все ответы верны

++++

II-bob. NSS, AAO [41-80] (40)

41. Задача математического программирования является задачей линейного программирования, если

=====

целевая функция является линейной, а система ограничений - система линейных уравнений или неравенств.

=====

целевая функция является линейной, а система ограничений нелинейная.

=====

система ограничений - это система линейных уравнений или неравенств, а целевая функция нелинейная.

=====

система ограничений - это система линейных уравнений или неравенств, а целевая функция отсутствует.

++++

42. Задача линейного программирования состоит в

=====

отыскание наибольшего (наименьшего) значения линейной функции при наличии линейных ограничений.

=====

создание линейной программы на избранном языке программирования, предназначенной для решения поставленной задачи.

=====

описание линейного алгоритма решения заданной задачи.

=====

отыскании наибольшего (наименьшего) значения линейной функции при наличии нелинейных ограничений.

++++

43. Целевой функцией задачи линейного программирования может являться функция:

=====

$$F = 12x_1 + 20x_2 - 30x_3 \rightarrow \min$$

=====

$$F = \sqrt{x_1^2 + x_2^2} \rightarrow \min$$

=====

$$F = 3x_1 - 4x_2 + \sqrt{x_3} \rightarrow \max$$

=====

$$F = x_1^2 - 2x_2 \rightarrow \max$$

++++

В задаче составления кормовой смеси, или в задаче о диете коэффициент c_i

целевой функции $f = \sum_{i=1}^m c_i x_i$ это

44.

=====

стоимость единицы сырья i .

=====

количество i -го сырья в смеси.

=====

количество ингредиента, содержащегося в единице i -го вида сырья.

=====

количество ингредиентов в i -м сырье.

+++++

45. В задаче составления кормовой смеси, или в задаче о диете x_i это

=====

количество i -го сырья в смеси.

=====

стоимость единицы сырья i .

=====

количество ингредиента, содержащегося в единице i -го вида сырья.

=====

количество ингредиентов в i -м сырье.

+++++

46. В задаче составления кормовой смеси, или в задаче о диете b_j это

=====

минимальное количество ингредиента j , содержащегося в единице смеси.

стоимость единицы сырья i .

=====

количество ингредиента, содержащегося в единице i -го вида сырья.

=====

количество ингредиентов в i -м сырье.

+++++

47. В задаче составления кормовой смеси, или в задаче о диете a_{ij} это

=====

количество ингредиента j , содержащегося в единице i -го вида сырья.

=====

количество видов сырья.

=====

минимальный общий вес смеси.

=====

минимальное количество ингредиента j , содержащегося в единице смеси.

+++++

48. К задачам математического программирования относятся задачи на отыскание

=====

максимума или минимума функции, на неизвестные которой наложены ограничения.

=====

целевой функции.

=====

решения системы уравнений.

=====

решения системы неравенств.

++++

49. В задаче использования сырья a_{ij} это

=====

количество единиц i -го сырья, идущего на изготовление единицы j -й продукции.

=====

запасы сырья i -го вида идущего на изготовление единицы j -й продукции.

=====

величина прибыли, получаемой при реализации единицы j -й продукции.

=====

количество единиц остатков i -го сырья, используемого на изготовление единицы j -й продукции.

++++

50. Если в задаче использования сырья потребовать, чтобы в процессе производства какое-то сырье использовалось полностью, то ограничение для этого сырья

=====

выражается в виде уравнения.

=====

выражается в виде строгого неравенства.

=====

выражается в виде неравенства.

=====

во внимание не применяется.

++++

51. Какое из утверждений является верным?

=====

Безотходный способ раскроя является рациональным.

=====

Безотходный способ раскроя может быть рациональным.

=====

Безотходный способ раскроя не является рациональным.

=====

Рациональный способ раскроя является безотходным.

++++

52. Задачей линейного программирования в нормальной форме называется следующая задача

=====

$$f = CX \rightarrow \max$$
$$AX \leq A_0, \quad X \geq 0.$$

=====

$$f = CX \rightarrow \max ,$$
$$AX = A_0, X \geq 0.$$

=====

$$f = CX \rightarrow \max ,$$
$$AX \leq A_0$$

=====

$$f = CX \rightarrow \max ,$$
$$AX = A_0$$

+++++

53. Задачей линейного программирования в канонической форме называется следующая задача

=====

$$f = CX \rightarrow \max ,$$
$$AX = A_0, X \geq 0.$$

=====

$$f = CX \rightarrow \max$$
$$AX \leq A_0, \quad X \geq 0.$$

=====

$$f = CX \rightarrow \max ,$$
$$AX \leq A_0$$

=====

$$f = CX \rightarrow \max ,$$
$$AX = A_0$$

+++++

54. Общей задачей линейного программирования называется следующая задача

=====

$$f = CX \rightarrow \max ,$$
$$AX = A_0, X \geq 0.$$

=====

$$f = CX \rightarrow \max$$
$$AX \leq A_0, \quad X \geq 0.$$

=====

$$f = CX \rightarrow \max ,$$

$$AX \leq A_0$$

=====

$$f = CX \rightarrow \max ,$$

$$AX = A_0$$

+++++

55. Планом или допустимым решением общей задачи линейного программирования называется вектор X , удовлетворяющий условиям:

=====

$$AX = A_0, X \geq 0.$$

=====

$$AX = A_0.$$

=====

$$X \geq 0.$$

=====

$$AX \geq A_0.$$

+++++

56. Пусть векторы A_i являются m -мерными, то число его положительных компонент

=====

не может превышать m .

=====

равно m .

=====

равно $m-1$

=====

равно $m+1$

+++++

57. Если ограничение задано со знаком \leq , то дополнительная переменная вводится в это ограничение с коэффициентом

=====

1

=====

-1

=====

0

=====

M

+++++

58. Если ограничение задано со знаком \geq , то дополнительная переменная вводится в это ограничение с коэффициентом

=====

-1

=====

1

=====

0

=====

M

++++

59. В задаче об оптимальном распределении ресурсов дополнительная переменная x_{n+1} имеет экономический смысл:

=====

неиспользованные ресурсы i -го вида.

=====

прибыль от реализации продукции i -го вида.

=====

прибыль от реализации 1 единицы продукции i -го вида.

=====

использованные ресурсы i -го вида.

++++

60. В задаче об оптимальном распределении ресурсов требование не отрицательности накладывается на

=====

на основные и дополнительные переменные.

=====

только основные переменные.

=====

только на дополнительные переменные.

=====

первую и вторую переменные.

++++

61. Для всякого ли многогранника существует задача линейного программирования, допустимым множеством которой он является?

=====

Нет, только для выпуклого многогранника с неотрицательными координатами вершин.

=====

Нет, только для многогранника с положительными координатами вершин.

=====

Нет, только для многогранника, имеющего более трех вершин.

=====

Да, для всякого.

++++

62. Допустимое решение задачи линейного программирования:

=====

Должно одновременно удовлетворять всем ограничениям задачи.

=====

Должно удовлетворять некоторым, не обязательно всем, ограничениям задачи.

=====

Должно быть вершиной множества допустимых решений.

=====

Должно обеспечивать наилучшее значение целевой функции.

+++++

63. Геометрически задача линейного программирования представляет собой отыскание такой точки многогранника решений, координаты которой доставляет целевой функции минимальное значение, причем допустимыми решениями служат

=====

все точки многогранника решений.

=====

только угловые точки многогранника решений.

=====

только точки, лежащие на ребрах многогранника.

=====

только точки, лежащие на гранях многогранника

+++++

64. Малое предприятие производит изделия двух видов. На изготовление одного изделия вида А расходуется 2 кг сырья, на изготовление одного изделия вида В – 1 кг. Всего имеется 60 кг сырья. Требуется составить план производства, обеспечивающий получение наибольшей выручки, если отпускная стоимость одного изделия вида А 3 д.е., вида В - 1 д.е., причем изделий вида А требуется изготовить не более 25, а вида В – не более 30. Целевой функцией данной задачи является функция

=====

$$F(x_1, x_2) = 3x_1 + x_2 \rightarrow \max$$

=====

$$F(x_1, x_2) = 25x_1 + 30x_2 \rightarrow \max$$

=====

$$F(x_1, x_2) = 2x_1 + x_2 \rightarrow \max$$

=====

$$F(x_1, x_2) = 60 - 2x_1 - x_2 \rightarrow \min$$

+++++

65. Если среди оценок оптимального плана нулевые только оценки, соответствующие базисным векторам, то это означает, что

=====

оптимальный план единственный.

=====

линейная функция не ограничена на многограннике решений.

=====

оптимальный план не единственный.

=====

оптимальный план не единственный, и его количество равняется количеству

=====

отрицательных коэффициенты разложения x_{ij} соответствующего вектора.

++++

66. Если нулевая оценка соответствует вектору, не входящему в базис, то в общем случае это означает, что

=====

оптимальный план не единственный.

=====

линейная функция не ограничена на многограннике решений.

=====

оптимальный план единственный.

=====

оптимальный план не единственный, и его количество равняется количеству отрицательных коэффициенты разложения x_{ij} соответствующего вектора.

++++

67. Искусственные переменные в систему ограничений в ЗЛП общем виде вводятся с коэффициентом

=====

1

=====

0

=====

2

=====

-1

++++

68. Для отыскания оптимального плана расширенной задачи в случае, если заранее не задана величина M , применяется симплексный метод, с составлением симплексных таблиц, которые имеют

=====

на одну строку больше, чем обычная симплексная таблица.

=====

на одну строку меньше, чем обычная симплексная таблица.

=====

на две строки больше, чем обычная симплексная таблица.

=====

на две строки меньше, чем обычная симплексная таблица.

++++

69. Пары двойственных задач называются симметричными, если в исходной задаче система ограничений задана в виде

=====

системы неравенств.

=====

системы уравнений.

=====

системы неравенств и уравнений.

=====

системы из одного уравнения и двух неравенств.

+++++

70. Пары двойственных задач называются несимметричными, если в исходной задаче система ограничений задана в виде

=====

системы уравнений.

=====

системы неравенств.

=====

системы неравенств и уравнений.

=====

системы из одного уравнения и двух неравенств.

+++++

71. В симметричной паре двойственных ЗЛП условие не отрицательности

=====

накладывается и на исходные, и на двойственные переменные.

=====

накладываются только на двойственные переменные.

=====

накладывается только на исходные переменные.

=====

не накладывается ни на исходные, ни на двойственные переменные.

+++++

72. Экономический смысл двойственной задачи иногда можно воспринимать как определение оптимальныхцен производственных ресурсов.

=====

цен производственных ресурсов

=====

5-производственных ресурсов

=====

3- производственных ресурсов

=====

4- производственных ресурсов

+++++

73. Для алгоритма симплексного метода начала должно выбрано базисные переменные, число которых должно быть равно—.

=====

m-числу ресурсов

=====

5--числу ресурсов

=====

3--числу ресурсов

=====

10--числу ресурсов

++++

74. Элемент a_{Sl} , расположенный на пересечении разрешающей строки и разрешающего столбца, объявляется

=====

Разрешающим элементом.

=====

Основным элементом

=====

Решающим элементом

=====

Опорным элементом

++++

75. Для алгоритма симплексного метода заполнение таблицы начнем с

=====

разрешающей строки.

=====

разрешающей столбца

=====

с первой разрешающей строки

=====

с второй разрешающей строки

++++

76. Для применения симплексного метода ЗЛП должна быть приведена к....

=====

Каноническому виду.

=====

Квадратному виду.

=====

Комплексному виду.

=====

Тригонометрическому виду.

++++

77. Определить оптимальный план производства, при котором доход предприятия

=====

будет максимальным.

=====

будет минимальным

=====

небудет максимальным

=====

небудет минимальным

++++

78. На основе полученного решения ЗЛП можно провести некоторые исследования по самой постановке задачи, по вопросам дефицита или излишков сырья.

=====

по запасам ресурсов

=====

по дефицитам ресурсов

=====

по вопросам ресурсов

=====

по излишкам ресурсов.

++++

79. Предприятие выпускает m видов изделий на основе n видов ресурсов, запасы которых составляют,

=====

соответственно b_1, b_2, \dots, b_m .

=====

соответственно a_1, a_2, \dots, a_m .

=====

соответственно c_1, c_2, \dots, c_m .

=====

соответственно a_1, b_2, \dots, c_m .

++++

80. ЗЛП Цена одного изделия j -того вида составляет.....

=====

c_j – денежных единиц.

=====

a_j – денежных единиц

=====

b_j – денежных единиц

=====

d_j – денежных единиц

++++

III-bob. MAN, AQU [81-120] (40)

81. В каком виде строится линейная модель для ТЗФ?

=====

$Y = ax + b$

=====

$Y = ax^2 + b$

=====

$$Y = an - bx$$

=====

$$Y = ax^3 + b$$

++++

82. В каком виде строится квадратичная модель для ТЗФ?

=====

$$Y = ax^2 + bx + c$$

=====

$$Y = ax + b$$

=====

$$Y = an - bx^2$$

=====

$$Y = xa^2 + bx + c$$

+++

83. Укажите тригонометрический ряд Фурье.

=====

$$f(x) \approx a_0 + \sum_{n=1}^{\infty} \left(a_n \cos \frac{2\pi nt}{T} + b_n \sin \frac{2\pi nt}{T} \right)$$

=====

$$f(x) = \sum_{j=1}^n f(x_j) \prod_{i \neq j} \frac{x_j - x_i}{x - x_i}$$

=====

$$a_0 = \frac{1}{T} \int_0^T f(t) dt$$

=====

$$F(x) = f(t) / T$$

++++

84. Укажите формула коэффициента a (ряд Фурье) с нулевым индексом на отрезке $[0; T]$.

=====

$$a_0 = \frac{1}{T} \int_0^T f(t) dt$$

=====

$$a_n = \frac{2}{T} \int_0^T f(t) \cdot \cos \frac{2\pi nt}{T} dt$$

=====

$$b_n = \frac{2}{T} \int_0^T f(t) \sin \frac{2\pi nt}{T} dt.$$

=====

$$a_0 = f(t) / T$$

+++

85. Укажите формула коэффициента a_n (ряд Фурье) по времени на отрезке $[0; T]$.

=====

$$a_n = \frac{2}{T} \int_0^T f(t) \cdot \cos \frac{2\pi nt}{T} dt$$

=====

$$a_0 = \frac{1}{T} \int_0^T f(t) dt$$

=====

$$b_n = \frac{2}{T} \int_0^T f(t) \sin \frac{2\pi nt}{T} dt.$$

=====

$$a_n = n * f(t) / T$$

++++

86. Как называются данные значения X_i в промежутке $[A, B]$?

=====

узловые точки

=====

значения функции

=====

увеличение аргумента

=====

Значения промежутка

++++

87. Для решения каких задач используются линейная и квадратичная модель?

=====

Аппроксимация

=====

Интерполяция

=====

Экстраполяция

=====

ряд Фурье

++++

88. Укажите формула коэффициента b_n (ряд Фурье) по времени на отрезке $[0; T]$.

=====

$$b_n = \frac{2}{T} \int_0^T f(t) \sin \frac{2\pi nt}{T} dt.$$

=====

$$a_0 = \frac{1}{T} \int_0^T f(t) dt$$

=====

$$a_n = \frac{2}{T} \int_0^T f(t) \cdot \cos \frac{2\pi nt}{T} dt$$

=====

$$b_n = n * f(t) / T$$

++++

89. Пусть $y=f(x)$ заданная функция. Как обозначится фиксированную величину приращения аргумента?

=====

$$\Delta x = h$$

=====

$$\Delta x = f(x)$$

=====

$$\Delta x = k$$

=====

$$dx=h$$

++++

90. Аппроксимация– это ...

=====

Приближение

=====

Отдаление

=====

Совместимость

=====

Результат

++++

91. Интерполяция – это ...

=====

определение внутренних значений

=====

определение внешних значений

=====

совместимость

=====

приближение

++++

92. Экстраполяция – это ...

=====

определение внешних значений

=====

определение результатов

=====

определение приближения

=====

определение внутренних значений

++++

93. При проведении экспериментов над модулями связь между входными значениями X и выходными значениями Y изучает ...

=====

Аппроксимация

=====

Интерполяция

=====

Экстраполяция

=====

простая итерация

++++

94. Какие величины используются для построения линейной модели для таблично заданных функций?

=====

$n+1$ и сумма всех x_i , y_i , $(x_i)^2$ и $x_i \cdot y_i$ ($i=0,1,2,\dots,n$) для нахождения a и b .

=====

сумма всех x_i , $(x_i)^2$, $(x_i)^3$ и y_i ($i=0,1,2,\dots,n$) для нахождения a и b .

=====

$2n$ и сумма всех y_i , $(x_i)^2$, $(x_i)^3$ и $x_i \cdot y_i$ ($i=0,1,2,\dots,n$) для нахождения a и b .

=====

$2n+1$ и сумма всех x_i , y_i , $(x_i)^2$, $(x_i)^3$ и $x_i \cdot y_i$ ($i=0,1,2,\dots,n$) для нахождения a и b .

++++

95. Постройте линейную модель для следующей таблично заданной функции?

$x: [-2; 0; 2; 4]$

$y: [0; 2; 1; 3]$

=====

$Y=0.4x+1.1$

=====

$Y=0.5x+1.4$

=====

$Y=0.5x+1$

=====

$Y=0.4x+1$

++++

96. Вычислены коэффициенты ряда Фурье для табличной функции (сигнала). Определит ведущий гармоники их амплитуды, частоты начальный сдвиг фазы.

$i: [0; 1; 2; 3; 4; 5]$

$a_i: [0.003; 0.0001; 0.6; 0.0003; 0.0001; 0.0001]$

$b_i: [-; 0.0002; 0.8; 0.0002; 0.0004; 0.0001]$

$T=0.6; c_1 \ll c_2$ при $i>5$.

=====

$C_2=1; w_2=21; f_2=\arctg(3/4)$

=====

$C_2=1; w_2=6; f_2=\arctg(1/2)$

=====

$C_2=0.8$; $w_2=8$; $f_2=\arctg(0.8)$

=====

$C_2=1.4$; $w_2=12$; $f_2=\arctg(0.75)$

+++++

97. Вычислить коэффициенты ряда Фурье для табличной функции (сигнала). Определит ведущий гармоники их амплитуды и частоты.

i : [0; 1; 2; 3; 4; 5]

a_i : [0.3; 0.0003; 0.0004; 1.2; 0.0002; 0.0007]

b_i : [-; 0.0001; 0.0002; 0.9; 0.0004; 0.0001]

$T=0.5$; $c_i \ll c_3$ при $i > 5$.

=====

3- гармоника $C_3=1,5$; $w_3=37.7$

=====

3- гармоника $C_3=2$; $w_3=37.7$

=====

3- гармоника $C_3=1,5$; $w_3=32$

=====

3- гармоника $C_3=1,2$; $w_3=37.7$

+++++

98. Для таблично заданной функции

t_i : [0; 0.1; 0.2; 0.3; 0.4]

f_i : [1; 1.3; 1.4; 1.2; 1]

Как определяется коэффициенты ряда Фурье?

=====

Представив функцию как кусочно постоянную

=====

Приближенным интегрированием по методам трапеции

=====

Предварительно находим интерполяционный полином

=====

Применяем формулы прямоугольников для интегрирование

+++++

99. Какие базисные функции используются при исследовании и обработке цифровых сигналов?

=====

Тригонометрические функции

=====

Степенные функции

=====

Показательные функции

=====

Функции комплексных переменных

+++++

100. Какой признак наблюдается при разложении четных функций в ряд Фурье?

=====

Отсутствие компоненты \sin и все $b_n=0$

=====

Отсутствие компоненты \cos и все $a_n=0$

=====

Отсутствие четных гармоник $c_{2k}=0$

=====

Отсутствие нечетных гармоник $c_{2k-1}=0$

++++

101. Какой признак наблюдается при разложении нечетных функций в ряд Фурье?

=====

Отсутствие компоненты \cos и все $a_n=0$

=====

Отсутствие компоненты \sin и все $b_n=0$

=====

Отсутствие четных гармоник $c_{2k}=0$

=====

Отсутствие нечетных гармоник $c_{2k-1}=0$

++++

102. Что понимается под ортогональностью в функциональных множествах?

=====

Равенство нулю интеграла от произведения этих функций по заданному отрезку

=====

Перпендикулярность их графиков в точке их пересечения

=====

Равенство нулю интеграла от их отношения по заданному отрезку

=====

Такого свойства для функции нет

++++

103. Как определяется понятие скалярного произведения для функции заданных на отрезке $[a;b]$?

=====

Интеграл от их произведения по данному отрезку.

=====

Интеграл от их разности по данному отрезку.

=====

Интеграл от отношения функций по данному отрезку.

=====

Такого операции для функций нет.

++++

104. Как определяются норма функции в L_2 ?

=====

Через интеграл от квадрата функции по отрезку.

=====

Через интеграл от абсолютного значения функции по отрезку.

=====

Как модуль от интеграла функции по отрезку

=====

Как наибольшее по модулю значение на отрезке.

+++++

105. Какая система базисных функций называется ортонормированной?

=====

Если для всех $(f_i, f_j) = \{0, \text{при } i \neq j; 1, \text{при } i=j\}$

=====

Если для всех $(f_i, f_j) = 0, \text{при } i \neq j.$

=====

Если для всех $f_i, \|f_i\|=1.$

=====

Для всех $f_i(t)$ существует норма.

+++++

106. Что называется гармоникой в ряде Фурье и как определяются их амплитуда?

=====

Слагаемое ряда Фурье при конкретном $n=k$, а амплитуда $c_k = \sqrt{(a_k)^2 + (b_k)^2}$

=====

Слагаемое ряда Фурье будет гармоникой если $a_k=0$, а амплитуда $c_k=|b_k|$

=====

Слагаемое ряда Фурье будет гармоникой если $b_k=0$, а амплитуда $c_k=|a_k|$

=====

Слагаемое ряда Фурье будет гармоникой если $a_k \neq 0$ и $b_k \neq 0$, а амплитуда $c_k = 2\pi k/T$

+++++

107. Что называется гармоникой в ряде Фурье и как определяются их частота?

=====

Слагаемое ряда Фурье при конкретном $n=k$, а частота $w_k = 2\pi k/T$

=====

Слагаемое ряда Фурье будет гармоникой если $a_k=0$, а частота $w_k=|b_k|$

=====

Слагаемое ряда Фурье будет гармоникой если $b_k=0$, а частота $w_k=|a_k|$

=====

Слагаемое ряда Фурье будет гармоникой если $a_k \neq 0$ и $b_k \neq 0$, а частота $w_k = \sqrt{(a_k)^2 + (b_k)^2}$

+++++

108. Как определяются ведущие гармоники в ряде Фурье?

=====

Если $c_k \gg c_i$ при $i \neq k$, то k - тая гармоника ведущая.

=====

Если $c_k > c_i$ при $i \neq k$, то k - тая гармоника ведущая.

=====

Если $c_k \neq 0$, то k - тая гармоника ведущая.

=====

Если $c_k > 1$, то k -тая гармоника ведущая.

++++

109. Для чего используется выделение ведущих гармоник?

=====

Для отделения основной части сигнала.

=====

Для построения графика сигнала.

=====

Для определения амплитуды сигнала.

=====

Для определения частоты сигнала.

++++

110. Как определяются число необходимых гармоник в ряде Фурье?

=====

По порядку требуемой точности оставляются только те гармоники амплитуды которых больше этой точности.

=====

Это заранее задаётся.

=====

По заданным частотам.

=====

Из принципа чем больше, тем лучше.

++++

111. Для чего нужен спектральный анализ?

1. Для определения ведущих частот;
2. Определение доли ведущих гармоник;
3. Очищение сигнала;
4. Для построения графика.

=====

1, 2, 3

=====

1, 2, 4

=====

2, 3, 4

=====

1, 3

++++

112. Где применяется метод разложения в ряд Фурье цифровых сигналов?

1. В современных средствах передачи и приёма цифровых сигналов;
2. При геологических исследованиях в целях нахождения полезных ископаемых;
3. При определении химического состава далеких планет и звезд.

=====

1, 2, 3

=====

1, 2

=====

2, 3

=====

1, 3

+++++

113. Какая мера близости используется при определении коэффициентов ряда Фурье для функций заданных таблично?

=====

Минимальность интегральной нормы квадрата разности табличной функции и ряд Фурье.

=====

Минимальность модуля разности табличной функции и ряд Фурье.

=====

Равенство значений табличной функции и ряд Фурье.

=====

Минимальность любой нормы разности табличной функции и ряд Фурье.

+++++

114. В каком виде представляется табличная функция при разложении в ряд Фурье?

=====

В виде кусочно постоянной функции на всем отрезке $[0; T]$, с учетом табличных значений.

=====

В виде табличной функции в заданном виде.

=====

В виде ломанной функции соединяющей табличные точки на координатной плоскости.

=====

В виде интерполяционного полинома, построенного по табличным значениям.

+++++

115. Укажите правило перевода табличной функции к кусочно постоянному виду на $[0; T]$ если $t_i = ih$, $f(t_i) = f_i$, $T = Nh$.

=====

$f(t) = f_i$ при $t \in [t_i - h/2; t_i + h/2)$, $i = 1, 2, 3, \dots, N-1$; $f(t) = f_0$ при $t \in [0; h/2)$; $f(t) = f_N$ при $t \in (T - h/2; T]$.

=====

$f(t) = f_{i-1} + (t - t_{i-1})(f_i - f_{i-1})/h$, при $t \in (t_{i-1}; t_i)$, $i = 1, 2, 3, \dots, N$.

=====

$f(t) = f_i$, при $t \in (t_{i-1}; t_i)$, $i = 1, 2, 3, \dots, N$.

=====

$f(t) = f_i$, при $t \in (t_i; t_{i+1})$, $i = 0, 1, 2, \dots, N-1$.

+++++

116. В каких случаях считается уместной линейная модель зависимости найденная по методу наименьших квадратов?

=====

Когда сумма квадратов разностей линейной модели $y = ax + b$ и табличных значений меньше требуемой точности.

=====

Когда разность значений табличной функции и линейной модели $y=ax+b$ по модулю меньше требуемой точности.

=====

Когда другие модели не применимы.

=====

Когда число наблюдений слишком велико.

++++

117. Какие из следующих моделей зависимости относятся к двухпараметрическим?

1. Линейная модель;
2. Квадратичная модель;
3. Обратно-пропорциональная зависимость;
4. Показательная модель зависимости.

=====

1, 3, 4

=====

1, 2, 3

=====

2, 3, 4

=====

1, 2, 4

++++

118. Каким преобразованием пользуются при построении показательной модели зависимости $y = a_1 (a_0)^x$?

=====

$$\ln y = \ln a_1 + x \ln a_0$$

=====

$$y - a_1 = (a_0)^x$$

=====

$$y = a_1 + a_0 \ln x$$

=====

$$\ln y = \ln a_1 + a_0 \ln x$$

++++

119. Почему при аппроксимации табличных функций пользуются двух параметрическими моделями зависимости?

=====

Так как в природе и в технике редко встречаются сложные модели зависимости.

=====

Для простоты вычислений.

=====

Так как другие модели зависимости не встречаются в природе и в технике.

=====

Так как эти модели считаются наилучшими.

++++

120. Почему при аппроксимации табличных функций редко пользуется интерполяционными полиномами?

1. Так как степень полинома повышается с увеличением числа наблюдений.
2. Так как объем вычислений резко возрастает.
3. Наличие неустраняемых погрешностей в табличных данных.
4. Отсутствие в природе и технике сложных моделей зависимости.

=====

1, 3, 4

=====

1, 2, 3

=====

2, 3, 4

=====

1, 2, 4

++++

IV-bob. SKK, AOK [121-160] (40)

121. Какая парадигма объединяет алгоритмы, в которых исходную сложную задачу разбивают на два или более относительно простых задач и на основе решения этих задач далее строится решение исходной задачи?

=====

Разделяй и властвуй

=====

Грубая сила

=====

Жадные алгоритмы

=====

Пузырьковая сортировка

++++

122. Почему термин “грубая сила” вошла в обиход?

=====

из-за того, что при делении задачи на части, часто и не задумываются над правилом деления

=====

из-за того, что при делении задачи на части

=====

из-за того, что при делении задачи на части по определенному принципу

=====

из-за того, что при объединении не задумываются над правилом деления

++++

123. Составьте алгоритм по методу грубой силы для вычисления суммы
 $S=1+2+3+4+5+6+7+8$.

=====

$S1=1+2$; $S2=3+4$; $S3=5+6$; $S4=7+8$; $S5=S1+S2$; $S6=S3+S4$; $S=S5+S6$

=====

S=0; for i=1:8 S=S+i end

=====

S=0; i=8:1 S=S+i end

=====

S=1+2+3+4+5+6+7+8

+++++

124. Найдите алгоритм Евклида для нахождения НОД(x;y), где x=15, y=20.

=====

1-ый шаг: $z=y-x=5$, $x=5$, $y=15$; 2-ой шаг: $z=y-x=10$, $x=5$, $y=10$; 3-ый шаг: $z=y-x=5$, $z=x=5$, НОД(x;y)=5

=====

1-ый шаг: $z=y-x=5$; 2-ой шаг: НОД(x;y)=z=5

=====

1-ый шаг: $x=3*5$, $y=2^2*5$; 2-ой шаг: НОД(x;y)=5

=====

1-ый шаг: $z=y-x=5$, $y=5$; 2-ой шаг: $z=x-y=10$, $y=10$; 3-ый шаг: $z=x-y=5$, $y=5$, НОД(x;y)=5

+++++

125. Заданные массивы в каких порядках заданы? A=(1,2,3,4,5,6), B=(6,5,4,3,2,1), C=(2,1,4,3,6,5).

=====

по возрастанию, по убыванию, по неупорядоченному виду

=====

по взаимно упорядоченному виду

=====

по взаимно неупорядоченному виду

=====

по почти упорядоченному виду

+++++

126. Для того чтобы сохранить исходного массива что можно делать?

=====

Ввести новый массив, при этом $b_i = a_i$

=====

Определить минимальное значение элементов исходного массива

=====

Определить максимальное значение элементов массива

=====

Определить значения соседних элементов массива

+++++

127. В методе постепенной перестановки соседних элементов в порядке возрастания имеет сколько сравнений?

=====

$(n-1)*(n-1)$

=====

$n \cdot \lg(n)$

=====

$n \cdot \ln(n)$

=====

$n \cdot \ln(n) + n$

+++++

128. По принципу «разделяй и властвуй» весь массив делится на две (равные) части, эти части тоже в свое очередь делятся на две части, до скольки элементов массива проводится деления

=====

До одного элемента

=====

До два элемента

=====

До числа $n \cdot \lg(n)$ элементов массива

=====

До числа $n \cdot \ln(n)$ элементов массива

+++++

129. Как можно условно писать рекурсивную функцию по принципу «разделяй и властвуй»?

=====

$y = f(f(f \dots (x) \dots))$

=====

$y = f(x) * f(x) * \dots * f(x)$

=====

$y = f(x)$

=====

$y = 1/f(x)$

+++++

130. Как можно написать факториала в рекурсивном виде?

=====

если $f(n) = n!$, то $f(n+1) = (n+1) * f(n)$

=====

$f(n) = 1 * 2 * 3 * \dots * n$

=====

$y = f(f \dots (n) \dots)$

=====

$y = f(n)$

+++++

131. Каким образом, можно свести определителя порядка n к вычислению определителя порядка $n-1$?

=====

По разложению минорам

=====

По вычислению определением

=====

По заданному соотношению для определителя

=====

По виду определителя

+++++

132. При математическом моделировании задачи, связанная со стоимостью передачи информационных или материальных ресурсов для проектирования дорог между населенными пунктами, каналов связи между определенными пунктами или коммуникационных сетей водопроводных, газоносных и канализационных труб с какой теории можно воспользоваться?

=====

Теорией графов

=====

Теорией множеств

=====

Теорией булевых алгебр

=====

Элементами комбинаторики

+++++

133. Нахождение маршрута минимальной стоимости из маршрутов между вершинами заданного графа – это задача называется

=====

Задачей коммивояжера

=====

Задачей размены монет

=====

Задачей выбор заявок

=====

Задачей выделения максимальной клики

+++++

134. Остовое дерево – это

=====

Дерево минимальной стоимости на базе заданного графа

=====

Цикл минимальной стоимости на базе заданного графа

=====

Все маршруты на базе заданного графа

=====

Дерево максимальной стоимости на базе заданного графа

+++++

135. Найдите обобщенный жадный алгоритм из нижеследующих алгоритмов

=====

алгоритм Радо — Эдмондса

=====

Алгоритм Хаффмана

=====

Алгоритм Краскала

=====

Алгоритм Прима

+++++

136. Для того чтобы найти кратчайший путь по алгоритму Краскала в начале множество ребер дерева объявляется

=====

Пустым

=====

Максимальным

=====

Минимальным

=====

Все ребра графа

+++++

137. Когда заканчивается алгоритм Краскала?

=====

Когда не остается ациклические ребра

=====

Когда не остается циклические ребра

=====

Когда обходит по всем ребрам

=====

Когда проходим по всем вершинам

+++++

138. Чему равно число гамильтоновых циклов на полном графе?

=====

$n!$

=====

$(n-1)!$

=====

$n*(n-1)$

=====

$(n-1)*(n-1)$

+++++

139. Чтобы определить гамильтонов цикл минимальной стоимости, необходимо выполнить сколько операций сложения?

=====

$(n-1)n!$

=====

$(n-1)!$

=====

$n!$

=====

$(n-1)*(n-1)$

++++

140. Если число вершин равно шести, то сколько чисел ребер гамильтонова цикла?

=====

5

=====

6

=====

4

=====

7

++++

141. Из скольких шагов состоят алгоритмы разделения и правила и как они называются?

=====

Он состоит из 3 этапов 1) Этап разделения 2) Этап господства 3) Фаза слияния

=====

Состоит из 4 стадий 1) Стадия разделения 2) Стадия доминирования 3) Стадия подчинения 4) Стадия разделения

=====

Состоит из 2 стадий 1) Стадия разделения 2) Стадия подчинения

=====

Рекурсия - это конструкция, в которой функция вызывает сама себя.

++++

142. Что такое рекурсия?

=====

Рекурсия - это конструкция, в которой функция вызывает сама себя.

=====

Все элементы связаны со следующим элементом

=====

Сортировщик на основе сравнения несортированного массива

=====

Сравните каждый элемент в массиве один за другим с искомым элементом.

++++

143. Что такое Quicksort - алгоритм быстрой сортировки?

=====

Этот алгоритм является рекурсивным и сортирует, сравнивая среднее значение $N * \log_2 N$.

=====

Этот алгоритм представляет собой алгоритм сортировки, основанный на сравнении несортированного массива

=====

Этот алгоритм сравнивает каждый элемент в массиве один за другим с искомым элементом.

=====

Данный алгоритм разбивает основную задачу на части и решает каждую из них отдельно.

++++

144. Стек на основе массива это?

=====

Возвращает во времени элемент в массиве через индекс

=====

Массив, который может изменять свой размер самостоятельно

=====

Все элементы в стеке связаны со следующим элементом

=====

Массив, который не может изменить свой размер сам по себе

++++

145. Какой алгоритм называется алгоритмом сортировки слиянием?

=====

Этот алгоритм представляет собой алгоритм сортировки, основанный на сравнении несортированного массива

=====

Этот алгоритм является рекурсивным и сортирует, сравнивая среднее значение $N * \log_2 N$.

=====

Этот алгоритм сравнивает каждый элемент в массиве один за другим с искомым элементом.

=====

Этот алгоритм основан на делении последовательности на две, т.е. сравнении заданного x со средним элементом массива, если он большой, он занимает массив между концом и серединой, если маленький - это начало и o ' принимает массив между rta , и этот процесс повторяется каждый раз, пока элемент x не станет равным элементу сравниваемого массива, или пока не останутся элементы массива.

++++

146. Из скольких частей состоит алгоритм сортировки слиянием?

=====

2

=====

3

=====

4

=====

5

++++

147. Алгоритм двоичного поиска (алгоритм двоичного поиска) это?

=====

Серия основана на разделении на две части, то есть сравнении заданного X со средним элементом массива, если он большой, он занимает массив между концом и серединой, если маленький - это начало, а середина занимает массив между, и этот процесс повторяется каждый раз, пока элемент x не станет равным элементу сравниваемого массива, или пока не останутся элементы массива.

=====

$n - 1$ раз в массиве,двигающемся снизу вверх по сравнению с парой ключей. Если значение нижнего ключа меньше значения верхней пары ключей, то они заменяются

=====

Этот алгоритм сравнивает каждый элемент в массиве один за другим с искомым элементом.

=====

Этот алгоритм является рекурсивным и сортирует, сравнивая среднее значение $N * \log_2 N$.

++++

148. Что такое сортировка?

=====

Постоянное размещение ключей в памяти машины.

=====

Стирание памяти машины клавишей

=====

При обработке данных на компьютере информационное поле элемента

=====

Его размещение в машинной памяти

++++

149. Сколько существует видов сортировки?

=====

2

=====

3

=====

4

=====

только 1

++++

150. На каких принципах основан алгоритм выбора?

=====

Этот метод основан на следующих принципах: 1. Выбирается элемент с наименьшей клавишей. 2. Этот элемент заменяется первым элементом. Затем этот процесс повторяется с оставшимися $n-1$, $n-2$ элементами, пока не останется один «самый большой» элемент.

=====

Этот принцип широко используется в карточных играх. Элементы (карточки) делятся на воображаемые «готовые» $a(1)$, ..., $a(i-1)$ и исходные последовательности.

=====

На каждом шаге (начиная с $i = 2$ и увеличиваясь на одну единицу на каждом шаге) i -й элемент отделяется от исходной последовательности и помещается в желаемую позицию в готовой последовательности.

=====

$n - 1$ раз в массиве,двигающемся снизу вверх по сравнению с парой ключей. Если значение нижнего ключа меньше значения верхней пары ключей, то они заменяются

++++

151. Предположим, что $N = 0,01n^2 + 10n$ - количество сравнений. Если $n < 1000$, то второй член больше, в противном случае, если $n > 1000$, первый член больше. Итак, количество сравнений в малом n равно n , а какое число в большом n ?

=====

n^2

=====

$n1$

=====

n

=====

$n >$

++++

152. Какой алгоритм называется алгоритмом сортировки методом прямого сложения?

=====

Этот метод широко используется в карточных играх. Элементы (карточки) делятся на воображаемые «готовые» $a(1), \dots, a(i-1)$ и исходные последовательности. На каждом шаге (начиная с $i = 2$ и увеличиваясь на одну единицу на каждом шаге) i -й элемент отделяется от исходной последовательности и помещается в желаемую позицию в готовой последовательности.

=====

$n - 1$ раз в массиве,двигающемся снизу вверх по сравнению с парой ключей. Если значение нижнего ключа меньше значения верхней пары ключей, то они заменяются

=====

1. Выбирается элемент с наименьшим ключом. 2. Этот элемент заменяется первым элементом. 3. Затем этот процесс повторяется с оставшимися $n-1, n-2$ элементами, пока не останется один «самый большой» элемент.

=====

Позволяет легко решать сложные проблемы

++++

153. Алгоритм пузырьковой сортировки это?

=====

$n - 1$ раз в массиве,перемещающемся снизу вверх по сравнению с парой ключей. Если значение нижнего ключа меньше значения верхней пары ключей, то они заменяются

=====

Этот алгоритм является рекурсивным и сортирует, сравнивая среднее значение $N * \log_2 N$.

=====

Этот алгоритм сравнивает каждый элемент в массиве один за другим с искомым элементом.

=====

1. Выбирается элемент с наименьшим ключом. 2. Этот элемент заменяется первым элементом. 3. Затем этот процесс повторяется с оставшимися $n-1, n-2$ элементами, пока не останется один «самый большой» элемент.

++++

154. Как можно улучшить метод пузырей?

=====

Одновременный переход снизу вверх и сверху вниз в элементах массива можно улучшить.

=====

Перемещая элементы массива сверху вниз

=====

Путем одновременного перемещения

=====

Элементов массива сверху вниз и снизу вверх

++++

155. Объясните структуру стека и приведите пример.

=====

Все элементы в стеке связаны со следующим элементом, и, используя эту последовательность, мы можем идентифицировать «верхний» элемент в стеке.

=====

Возвращает элемент в массиве с течением времени, используя индекс

=====

Все элементы в стеке не связаны со следующим элементом,

=====

Элементы в стеке идентифицируются с использованием этой последовательности.

++++

156. Сколько байтов составляет размер первичной кэш-памяти?

=====

несколько 10кбайт

=====

80 Кбайт

=====

100 кбайт

=====

150 Кбайт

++++

157. Сколько байтов занимает вторичная кэш-память?

=====

128/256/512 Кбайт и т. д.

=====

1 МБ

=====

6 МБ

=====

10кбайт

++++

158. Какой массив называется динамическим массивом?

=====

Массив, который может изменять свой размер самостоятельно.

=====

Возвращает элемент в массиве с течением времени, используя индекс

=====

Массив, который сам по себе не может изменить свой размер

=====

На его элемент можно ссылаться с помощью индекса

+++++

159. В какие интервалы количество сравнений в идеале отбора:

=====

от 1 до n каждый

=====

по 2 и по n

=====

n и n каждый

=====

между n и n^2

+++++

160. На сколько частей разделяет алгоритм быстрого сортировки Quicksort для сортировки заданного массива?

=====

2

=====

1

=====

3

=====

4

+++++

V-bob. NOA, AYoQ [161-200] (40)

161. Эвристический алгоритм?

=====

Это алгоритм решения проблемы, которая не была доказана во всех возможных случаях, но, как известно, в большинстве случаев находит очень хорошее решение.

=====

Алгоритм решения проблемы, которая, как известно, находит хорошее решение.

=====

Не гарантирует лучшего решения.

=====

Нет правильного ответа.

++++

162. Каковы пути решения NP-сложных задач?

=====

Ясный, эвристический, метаввристический

=====

Неопределенные методы

=====

Стохастические методы

=====

Нет правильного ответа.

++++

163. NP-решение сложных задач конкретными методами-...

=====

все возможные решения основаны на полном рассмотрении (полный перебор), что, в свою очередь, снижает их привлекательность

=====

Это приводит к относительно ограниченному поиску решений и обычно находит очень хорошее решение в течение оптимального времени. Но и у этих методов есть недостаток, то есть они приближительны.

=====

Наиболее эффективны, но в этих методах есть параметр, который напрямую влияет на результат, на основе входных данных, на практике каждый раз этот параметр приходится пересчитывать.

=====

Нет правильного ответа.

++++

164. NP-эвристические методы решения сложных задач-...

=====

Это приводит к относительно ограниченному поиску решений и обычно находит очень хорошее решение в течение оптимального времени. Но и у этих методов есть недостаток, то есть они приближительны.

=====

все возможные решения основаны на полном просмотре (полный перебор), а это, в свою очередь, снижает их привлекательность.

=====

Наиболее эффективны, но в этих методах есть параметр, который напрямую влияет на результат, на основе входных данных, на практике каждый раз этот параметр приходится пересчитывать.

=====

Нет правильного ответа.

++++

165. NP-метаэвристические методы решения сложных задач-...

=====

Наиболее эффективны, но в этих методах есть параметр, который напрямую влияет на результат, на основе входных данных, на практике каждый раз этот параметр приходится пересчитывать.

=====

Все возможные решения основаны на полном рассмотрении (полный перебор), что, в свою очередь, снижает их привлекательность

=====

Это приводит к относительно ограниченному поиску решений и обычно находит очень хорошее решение в течение оптимального времени. Но и у этих методов есть недостаток, то есть они приближительны.

=====

Нет правильного ответа.

+++++

166. Какую задачу решают эквивалентные алгоритмы?

=====

Для двух натуральных чисел m и n вычисляется $EKUK$.

=====

Вычисление квадрата и корня натурального числа.

=====

Находит $EKUB$ для двух натуральных чисел m и n .

=====

Находит среднее арифметическое заданного числа n .

+++++

167. Когда данный массив $A [0..2 * N]$ изначально состоит из 2 элементов, затем 4 элементов и т. д., все элементы сортируются одновременно.

=====

Сортировка по Shell

=====

Сортировка по выбору

=====

Сортировка пузырьковым методом

=====

Сортировка по методу пирамиды

+++++

168. Какой из приведенных ниже алгоритмов является алгоритмом? 1) Посмотрите на числа M и N как на первое и второе число; 2) сравните первое и второе число как на 5 пунктов, если они равны; 3) поменяйте местами, если первое число меньше второго; 4) умножьте второе число на первое число, вычитая первое число и получите второе число; 5) посмотрите на первое число как на ответ.

=====

Алгоритм Евклида

=====

Алгоритм машины Тьюринга

=====

Нормальные алгоритмы Маркова

=====

Пост алгоритм

++++

169. ... сам человек служит алгоритму своей машины

=====

Машина Тьюринга

=====

Электронная вычислительная машина

=====

Механическая машина

=====

Электронная механическая машина

++++

170. Последний элемент называется ... стека.

=====

Самый высокий

=====

Точка

=====

конец

=====

глава

++++

171. Это называется..., если количество элементов в стеке ограничено.

=====

Ограниченный стек

=====

Пустой стек

=====

Неограниченный стек

=====

Бесконечный стек

++++

172. Стек, который не имеет элемента, называется ...

=====

Пустой стек

=====

Ограниченный стек

=====

Неограниченный стек

=====

Бесконечный стек

++++

173. Неориентированные непересекающиеся строки называются ...

=====

деревья

=====

поклони

=====

урмани

=====

Графики

+++++

174. Кто опубликовал первые научные работы по теории алгоритмов, не зная друг друга?

=====

Алан Тьюринг, Алонзо Чёрч и Эмиль Пост

=====

Алонзо Чёрч, Эмиль Пост и Дональд Кнут

=====

Алан Тьюринг, Альфред Ахо и Эмиль Пост

=====

Алонзо Чёрч, Джон Хопкрофт и Алан Тьюринг

+++++

175. Структура машины Post имеет структуру $p \times K \times m$, где ...

=====

p -порядковый номер команды; Действие выполняется головой \underline{K} ; \underline{m} - номер следующей исполняемой команды;

=====

операция, выполняемая с p -головкой; \underline{K} - номер следующей пополняемой команды; порядковый номер \underline{m} -команды;

=====

p - серийный номер команды; \underline{K} - номер следующей размещаемой команды; \underline{m} - действие головы;

=====

\underline{N} -порядковый номер группы; \underline{K} -действие, которое должно быть выполнено с помощью галочки; \underline{m} -номер ячейки, в которой должен быть выполнен данный элемент;

+++++

176. В теории полноты NP рассматриваются только вопросы решения!

=====

вопросы, на которые нужно ответить “да” или “нет”

=====

вопрос, на который нужно ответить “да”

=====

вопрос, на который нужно ответить “нет”

=====

правильный ответ-В и С.

+++++

177. NP-полные задачи в некотором смысле образуют набор “типичных” задач класса NP:

=====

любая другая задача класса NP может быть решена таким же образом, если для некоторых из них найден алгоритм “быстрого” решения.

=====

если для некоторых из них не найден алгоритм “быстрого” решения, то любая другая задача в классе NP может быть решена таким же образом.

=====

Не подходит для любой другой проблемы в классе NP.

=====

все ответы правильные

++++

178. Укажите конкретные методы решения NP-полных задач.

=====

Полный перебор; Динамическое программирование; Сети и границы.

=====

Методы типа FF

=====

Жадные и градиентные методы

=====

Случайные методы

++++

179. Укажите вероятностные методы решения NP-полных задач.

=====

Жадный и градиентный методы; Случайные методы; Методы типа FF

=====

Полный повторный выбор

=====

Динамическое программирование

=====

Сети и границы

++++

180. Суть метода полного повторного отбора заключается в следующем:

=====

1) рассмотрение всех возможных случаев; 2) найти решения, удовлетворяющие условию данной задачи; 3) показать, что других решений нет.

=====

1) рассмотрение всех возможных случаев; 2) поиск решений, не соответствующих условиям поставленной задачи;

=====

1) разбор всех возможных случаев; 2) найти решения согласно условиям данной задачи; 3) показать, что есть другие решения.

=====

правильный ответ В и С

++++

181. Что такое класс задач P?

=====

P-класс задач, решаемых во времени на полиноме (от входной величины).

=====

P-класс задач, которые можно проверить во время полинома.

=====

NP-класс задач, решаемых за время в полиноме (от входной величины).

=====

NP-класс задач, которые можно проверить во время полинома.

+++++

182. Что такое класс задач NP?

=====

P-класс задач, решаемых во времени на полиноме (от входной величины).

=====

P-класс задач, которые можно проверить во время полинома.

=====

NP-класс задач, решаемых за время в полиноме (от входной величины).

=====

NP-класс задач, которые можно проверить во время полинома.

+++++

183. Как доказать, что задача принадлежит классу NP?

=====

Приведя алгоритм решения задачи с использованием полиномиального времени в детерминированной машине Тьюринга.

=====

Приведя алгоритм решения задачи с использованием полиномиального времени в недетерминированной машине Тьюринга.

=====

Приведя алгоритм решения задачи без использования полиномиального времени в недетерминированной машине Тьюринга.

=====

Приведя алгоритм решения задачи без использования полиномиального времени в детерминированной машине Тьюринга.

+++++

184. Вопросы NP относятся к классу самых сложных задач?

=====

Ха. Есть более сложные задачи, решения которых можно проверить даже во время полинома.

=====

Да. Есть более сложные задачи, которые нельзя решить даже с помощью многочленов.

=====

Нет. Есть более сложные задачи, которые нельзя решить даже при многочленах..

=====

Нет. Есть более сложные задачи, которые можно решить даже во время полиномов.

++++

185. NP-укажите ответ на сложную задачу с правильным определением.

=====

Все проблемы в классе NP являются уменьшаемыми проблемами. Кроме того, сама проблема не обязательно должна принадлежать классу NP, она может быть более сложной.

=====

Все проблемы в классе NP являются уменьшаемыми проблемами. Кроме того, сама проблема не обязательно должна принадлежать классу NP, она может быть более сложной.

=====

Все проблемы в классе NP являются уменьшаемыми проблемами. Кроме того, сама проблема не обязательно должна принадлежать классу NP, она может быть более сложной.

=====

Все проблемы в классе NP являются уменьшаемыми проблемами. Кроме того, сама проблема не обязательно должна принадлежать классу NP, она может быть более сложной.

++++

186. NP-показать полную проблему.

=====

P-сложные задачи, принадлежащие классу NP.

=====

NP-сложные задачи, принадлежащие классу NP.

=====

NP-сложные задачи, принадлежащие классу P.

=====

P-сложные задачи, относящиеся к классу P.

++++

187. Как доказать NP-трудную задачу?

=====

Доказано, что некоторые задачи класса NP сводятся к заданной, и таким образом определяется первая NP-сложная (и-полная) задача.

=====

Доказано, что не все задачи класса NP сводятся к заданному. Таким образом, определяется первая NP-сложная (и-полная) проблема.

=====

Доказано, что все задачи класса NP сводятся к заданному. Таким образом, определяется первая NP-сложная (и-полная) проблема.

=====

Доказывается, что все задачи класса NP сводятся к заданному, и таким образом определяется первая NP-сложная (и-полная) задача.

++++

188. Как доказать NP-комплексную задачу?

=====

Доказано, что NP-сложная задача.

=====

Доказано, что P-сложная задача.

=====

Доказано, что P-это задача.

=====

Доказано, что NP-сложная задача.

++++

189. Одной из центральных проблем проектирования алгоритмов является... .

=====

Вопрос о равенстве классов P и NP.

=====

Задача о неравенстве классов P и NP.

=====

Проблема класса P.

=====

Проблема класса NP.

++++

190. Класс P относится к классу NP?

=====

Да, это. Класс P является дополнением к классу NP.

=====

Да, это. Класс P является частью класса NP.

=====

Нет, это не так. Класс P и класс NP - отдельные предметные классы.

=====

Нет, это не так. Но класс P и класс NP являются дополнительными классами.

++++

191. Какие типы задач относятся к классу NP?

=====

Задачи с неопределенной полиномиальной сложностью

=====

Задачи с полиномиальной сложностью

=====

Детерминированные проблемы

=====

Проблемы, решение которых легко найти

++++

192. Какой из приведенных вопросов относится к разряду NP-полных вопросов?

=====

Машина Тьюринга

=====

Задача нахождения максимума

=====

Вопрос сортировки

=====

Вопрос поиска ближайшего пути

+++++

193. Какая из приведенных ниже алгоритмических оценок выполняется в наименьшее время?

=====

$O(N)$

=====

$O(N^3)$

=====

$O(N^2)$

=====

$O(N \log N)$

+++++

194. Вопрос коммивояжера-это....

=====

Речь идет об объезде заранее заданных пунктов за минимальное время или при достижении минимального пробега

=====

Речь идет о доставке различных грузов из многих источников по разным адресам

=====

Вопрос минимизации затрат на продукцию

=====

Вопрос построения модели увеличения прибыли

+++++

195. Каковы последствия использования рекурсии большой глубины в программе?

=====

Долго работает и стек переполняется

=====

Ошибка работает

=====

При повторном расчете будет погрешность

=====

Функция сводится к бесконечности

+++++

196. Если алгоритм тратит 64 С на выполнение со сложностью $O(N \log N)$, сколько времени этот же алгоритм тратит со сложностью $O(N^2)$?

=====

256

=====

100

=====

512

=====
500

++++

197. Если алгоритм тратит 256с на выполнение со сложностью $O(N)$, сколько времени этот же алгоритм тратит со сложностью $O(N\log N)$?

=====
2048

=====
1024

=====
100

=====
500

++++

198. Если алгоритм тратит 1024 с на выполнение со сложностью $O(N)$, сколько времени этот же алгоритм тратит со сложностью $O(N\log N)$?

=====
10240

=====
1024

=====
2048

=====
720

++++

199. От каких параметров зависит точность расчета суммы накопительных остатков?

=====
N-число элементов в множестве, P-число двоичных чисел чисел, составляющих множество

=====
N-число элементов множества, P-сумма элементов множества

=====
M-количество элементов, K-количество пар элементов

=====
N-число подмножеств топологии, P-число элементов подмножества множества

=====
M-количество элементов, K-количество пар элементов

=====
N-число подмножеств топологии, P-число элементов подмножества множества

=====
N-число подмножеств топологии, P-число элементов подмножества множества

=====
N-число подмножеств топологии, P-число элементов подмножества множества

++++

200. Показать базовую формулу динамического программирования:

=====
$$f(C_1, \{C_k\}) = \min_{C_j \in \{C_k\}} (C_{ij} + f(C_j; \{C_k\} - \{C_j\}))$$

=====
$$C(X, \bar{X}) = \sum_{ij} b_{ij}$$

=====
$$C(X, \bar{X}) = \sum_{ij} b_{ij}$$

=====
$$C(X, \bar{X}) = \sum_{ij} b_{ij}$$

=====

$$f(C_1, \{C_k\}) = \max_{C_j \in \{C_k\}} \left(f(C_j; \{C_k\} - \{C_j\}) \right)$$

=====

$$\sum_{j=1}^S R_j(y_j) \rightarrow \max, \min$$