

Министерство науки и высшего образования Российской Федерации
Нижекамский химико-технологический институт (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Казанский национальный исследовательский технологический
университет»
(НХТИ ФГБОУ ВО «КНИТУ»)



УТВЕРЖДАЮ
Заместитель директора по УР
Н.И. Никифорова
«30» мая 2022 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине (модулю)

Б1.В.04 Структуры и алгоритмы компьютерной обработки данных
(код и наименование дисциплины (модуля))

09.03.01 Информатика и вычислительная техника
(код и наименование направления подготовки/специальности)

Автоматизированные системы обработки информации и управления
(наименование профиля/специализации)

бакалавр
квалификация

очная, очно-заочная
форма обучения

Нижекамск, 2022

Составитель ФОС:

доцент

(должность)



(подпись)

Л.Р. Вотякова

(Ф.И.О)

ФОС рассмотрен и одобрен на заседании кафедры ИСТ,
протокол от 20.04.2022 г. № 8

Зав. кафедрой



(подпись)

О.В. Матухина

(Ф.И.О.)

Эксперт:

Руководитель ООП



(подпись)

Л.А. Амаева

Перечень компетенций и индикаторов достижения компетенций с указанием этапов формирования в процессе освоения дисциплины

Компетенция:

УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

Индикаторы достижения компетенции:

УК-1.1. Знает методики поиска, сбора и обработки информации; актуальные российские и зарубежные источники информации в сфере профессиональной деятельности; метод системного анализа

УК-1.2. Умеет применять методики поиска, сбора и обработки информации; осуществлять критический анализ и синтез информации, полученной из разных источников; применять системный подход для решения поставленных задач

УК-1.3. Владеет навыками поиска, сбора и обработки, критического анализа и синтеза информации; использования системного подхода для решения поставленных задач

Компетенция:

ПК-1. Способен разрабатывать требования и проектировать программное обеспечение.

Индикаторы достижения компетенции:

ПК-1.1. Знает методологии разработки программного обеспечения, назначение и возможности средств проектирования программного обеспечения.

ПК-1.2. Умеет разрабатывать требования и проектировать программное обеспечение.

ПК-1.3. Владеет навыками разработки требований к программным продуктам, использования методов и средств проектирования программного обеспечения.

Индикаторы достижения компетенции	Этапы формирования в процессе освоения дисциплины				Наименован ие оценочного средства
	Лекции	Практические занятия	Лаборато рные занятия	Курсовой проект (работа)	
УК-1.1	Разделы дисципли ны 1-6.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-6.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
УК-1.2	Разделы дисципли ны 1-6.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-6.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
УК-1.3	Разделы дисципли ны 1-6.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-6.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-1.1	Разделы дисципли ны 1-6.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-6.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-1.2	Разделы дисципли ны 1-6.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-6.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-1.3	Разделы дисципли ны 1-6.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-6.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест

Перечень оценочных средств по дисциплине (модулю)

Оценочные средства	Кол-во	Min, баллов (базовый уровень)	Max, баллов (повышенный уровень)
Расчетно-графические работы	5	36	60
Тест	1	24	40

Шкала оценивания

Цифровое выражение	Выражение в баллах:	Словесное выражение	Критерии оценки индикаторов достижения при форме контроля:
			экзамен / зачет с оценкой
5	87 - 100	Отлично (зачтено)	Оценка «отлично» выставляется студенту, если теоретическое содержание курса освоено полностью, без пробелов; исчерпывающе, последовательно, четко и логически стройно излагает материал; свободно справляется с задачами, вопросами и другими видами применения знаний; использует в ответе дополнительный материал все предусмотренные программой задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному; анализирует полученные результаты; проявляет самостоятельность при выполнении заданий
4	74 - 86	Хорошо (зачтено)	Оценка «хорошо» выставляется студенту, если теоретическое содержание курса освоено полностью, необходимые практические компетенции в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения достаточно высокое. Студент твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос.
3	60 - 73	Удовлетворительно (зачтено)	Оценка «удовлетворительно» выставляется студенту, если теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, большинство предусмотренных программой заданий выполнено, но в них имеются ошибки, при ответе на поставленный вопрос студент допускает неточности, недостаточно правильные формулировки, наблюдаются нарушения логической последовательности в изложении программного материала.
2	Ниже 60	Неудовлетворительно (незачтено)	Оценка «неудовлетворительно» выставляется студенту, если он не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы, большинство предусмотренных программой обучения учебных заданий не выполнено, качество их выполнения оценено числом баллов, близким к минимальному

Министерство образования и науки Российской Федерации
Нижекамский химико-технологический институт (филиал) федерального
государственного бюджетного образовательного учреждения
высшего образования
«Казанский национальный исследовательский технологический университет»

Факультет информационных технологий
Кафедра информационных систем и технологий
Направление подготовки 09.03.01 Информатика и вычислительная техника

Курс 2
Семестр 4

Экзаменационный тест
по дисциплине (модулю)

Б1.В.04 Структуры и алгоритмы компьютерной обработки данных

1. Структура данных представляет собой
 - а) **набор правил и ограничений, определяющих связи между отдельными элементами и группами данных**
 - б) набор правил и ограничений, определяющих связи между отдельными элементами данных
 - с) набор правил и ограничений, определяющих связи между отдельными группами данных
 - д) некоторую иерархию данных
2. Линейный список, в котором доступен только последний элемент, называется
 - а) **стеком**
 - б) очередью
 - с) деком
 - д) массивом
 - е) кольцом
3. Структура данных работа с элементами которой организована по принципу FIFO (первый пришел - первый ушел) это –
 - а) Стек
 - б) Дек
 - в) **Очередь**
 - г) Список
4. Линейный последовательный список, в котором включение исключение элементов возможно с обоих концов, называется
 - а) стек
 - б) очередь

- c) **деком**
- d) кольцевой очередью

5. В чём особенности очереди?

- a) **открыта с обеих сторон ;**
- b) открыта с одной стороны на вставку и удаление;
- c) доступен любой элемент.

6. В чём особенности стека?

- a) открыт с обеих сторон на вставку и удаление;
- b) доступен любой элемент;
- c) **открыт с одной стороны на вставку и удаление.**

7. Какую дисциплину обслуживания принято называть FIFO?

- a) стек;
- b) **очередь;**
- c) дек.

8. Какая операция читает верхний элемент стека без удаления?

- a) pop;
- b) push;
- b) **stackpop.**

9. Каково правило выборки элемента из стека?

- a) первый элемент;
- b) **последний элемент;**
- c) любой элемент.

10. Как освободить память от удаленного из списка элемента?

- a) p=getnode;
- b) ptr(p)=nil;
- c) **freenode(p);**
- d) p=lst.

11. Как создать новый элемент списка с информационным полем D?

- a) p=getnode;
- b) **p=getnode; info(p)=D;**
- c) p=getnode; ptr(D)=lst.

12. Как создать пустой элемент с указателем p?

- a) **p=getnode;**
- b) info(p);
- c) freenode(p);
- d) ptr(p)=lst.

13. Сколько указателей используется в односвязных списках?

- a) 1
- b) 2;
- c) сколько угодно.

14. В чём отличительная особенность динамических объектов?

- a) порождаются непосредственно перед выполнением программы;
- b) возникают уже в процессе выполнения программы;**
- c) задаются в процессе выполнения программы.

15. При удалении элемента из кольцевого списка...

- a) список разрывается;
- b) в списке образуется дыра;
- c) список становится короче на один элемент .**

16. Для чего используется указатель в кольцевых списках?

- a) для ссылки на следующий элемент;
- b) для запоминания номера сегмента расположения элемента;
- c) для ссылки на предыдущий элемент ;**
- d) для расположения элемента в списке памяти.

17. Чем отличается кольцевой список от линейного?

- a) в кольцевом списке последний элемент является одновременно и первым;
- b) в кольцевом списке указатель последнего элемента пустой;
- c) в кольцевых списках последнего элемента нет ;**
- d) в кольцевом списке указатель последнего элемента не пустой.

18. Сколько указателей используется в односвязном кольцевом списке?

- a) 1;**
- b) 2;
- c) сколько угодно.

19. В каких направлениях можно перемещаться в кольцевом двунаправленном списке?

- a) в обоих;**
- b) влево;
- c) вправо.

20. С помощью какой структуры данных наиболее рационально реализовать очередь?

- a) стек;
- b) список;**
- c) дек.

21. В памяти ЭВМ бинарное дерево удобно представлять в виде:

- a) связанных линейных списков;
- b) массивов;
- c) связанных нелинейных списков.**

22. Элемент t , на который нет ссылок:

- a) корнем;**
- b) промежуточным;
- c) терминальным (лист).

23. Дерево называется полным бинарным, если степень исходов вершин равна:

- a) 2 или 0;**
- b) 2;
- c) M или 0;
- d) M .

24. Даны три условия окончания просеивания при сортировке прямым включением. Найдите среди них лишнее.

- a) найден элемент $a(i)$ с ключом, меньшим чем ключ u ;
- b) найден элемент $a(i)$ с ключом, большим чем ключ u ;**
- c) достигнут левый конец готовой последовательности.

25. Какой из критериев эффективности сортировки определяется формулой $M = 0,01 * n * n + 10 * n$?

- a) число сравнений;**
- b) время, затраченное на написание программы;
- c) количество перемещений;
- d) время, затраченное на сортировку.

26. Как называется сортировка, происходящая в оперативной памяти?

- a) сортировка таблицы адресов;
- b) полная сортировка;
- c) сортировка прямым включением;
- d) внутренняя сортировка;**
- e) внешняя сортировка.

27. Как можно сократить затраты машинного времени при сортировке большого объема данных?

- a) производить сортировку в таблице адресов ключей;**
- b) производить сортировку на более мощном компьютере;
- c) разбить данные на более мелкие порции и сортировать их.

28. Существуют следующие методы сортировки. Найдите ошибку.

- a) строгие;
- b) улучшенные;
- c) динамические.**

29. Метод сортировки называется устойчивым, если в процессе сортировки...

- a) относительное расположение элементов безразлично;
- b) относительное расположение элементов с равными ключами не меняется;**
- c) относительное расположение элементов с равными ключами изменяется;
- d) относительное расположение элементов не определено.

30. Улучшенные методы имеют значительное преимущество:

- a) при большом количестве сортируемых элементов;**
- b) когда массив обратно упорядочен;
- c) при малых количествах сортируемых элементов;
- d) во всех случаях.

31. Что из перечисленных ниже понятий является одним из типов сортировки?

- a) внутренняя сортировка;**
- b) сортировка по убыванию;
- c) сортировка данных;
- d) сортировка по возрастанию.

32. Сколько сравнений требует улучшенный алгоритм сортировки?

- a) $n \cdot \log(n)$;**
- b) en ;
- c) $n \cdot n/4$.

33. Сколько сравнений и перестановок элементов требуется в пузырьковой сортировке?

- a) $n \cdot \log(n)$;
- b) $(n \cdot n)/4$;**
- c) $(n \cdot n - n)/2$.

34. Сколько дополнительных переменных нужно в пузырьковой сортировке помимо массива, содержащего элементы?

- a) 0 (не нужно);
- b) всего 1 элемент;**
- c) n переменных (ровно столько, сколько элементов в массиве).

35. Как рассортировать массив быстрее, пользуясь пузырьковым методом?

- a) одинаково;**
- b) по возрастанию элементов;
- c) по убыванию элементов.

36. В чём заключается идея метода QuickSort?

- a)выбор 1,2,...n – го элемента для сравнения с остальными;
- b)разделение ключей по отношению к выбранному;**
- c)обмен местами между соседними элементами.

37. Массив сортируется “пузырьковым” методом. За сколько проходов по массиву самый “лёгкий” элемент в массиве окажется вверху?

- a)за 1 проход;**
- b)за n-1 проходов;
- c)за n проходов, где n – число элементов массива.

38. При обходе дерева слева направо получаем последовательность...

- a) отсортированную по убыванию;
- b) неотсортированную;**
- c) отсортированную по возрастанию.

39. При обходе дерева слева направо его элемент заносится в массив...

- a)при втором заходе в элемент;**
- b)при первом заходе в элемент;
- c)при третьем заходе в элемент.

40. Где эффективен линейный поиск?

- a)в списке;
- b)в массиве;
- c)в массиве и в списке.**

41. Какой поиск эффективнее?

- a)линейный;
- b)бинарный;**
- c)без разницы.

42. В чём суть бинарного поиска?

- a)нахождение элемента массива x путём деления массива пополам каждый раз, пока элемент не найден;**
- b)нахождение элемента x путём обхода массива;
- c)нахождение элемента массива x путём деления массива.

43. Как расположены элементы в массиве бинарного поиска?

- a)по возрастанию;**
- b)хаотично;
- c)по убыванию.

44. В чём суть линейного поиска?

- a) производится последовательный просмотр от начала до конца и обратно через 2 элемента;
- b) производится последовательный просмотр элементов от середины

таблицы;

с) производится последовательный просмотр каждого элемента.

45. Где наиболее эффективен метод транспозиций?

- а. в массивах и в списках;**
- б. только в массивах;
- с. только в списках.

46. В чём суть метода транспозиции?

- а. перестановка местами соседних элементов;
- б. нахождение одинаковых элементов;
- с. перестановка найденного элемента на одну позицию в сторону начала списка .**

47. Что такое уникальный ключ?

- а. если разность значений двух данных равна ключу;
- б. если сумма значений двух данных равна ключу;
- с. если в таблице есть только одно данное с таким ключом.**

48. В чём состоит назначение поиска?

- а. среди массива данных найти те данные, которые соответствуют заданному аргументу;**
- б. определить, что данных в массиве нет;
- с. с помощью данных найти аргумент.

49. Элемент дерева, который не ссылается на другие, называется

- а) корнем
- б) листом**
- с) узлом
- д) промежуточным

50. Элемент дерева, на который не ссылаются другие, называется

- а) корнем**
- б) листом
- с) узлом
- д) промежуточным

51. Элемент дерева, который имеет предка и потомков, называется

- а) корнем
- б) листом
- с) узлом
- д) промежуточным**

52. Высотой дерева называется

- а) максимальное количество узлов
- б) максимальное количество связей**

- c) максимальное количество листьев
- d) **максимальная длина пути от корня до листа**

53. Степенью дерева называется

- a) **максимальная степень всех узлов**
- b) максимальное количество уровней его узлов
- c) максимальное количество узлов
- d) максимальное количество связей
- e) максимальное количество листьев

54. Как определяется длина пути дерева

- a) **как сумма длин путей всех его узлов**
- b) как количество ребер от узла до вершины
- c) как количество ребер от листа до вершины
- d) как максимальное количество ребер
- e) как максимальное количество листьев
- f) **как длина самого длинного пути от ближнего узла до какого-либо**

листа

55. Дерево называется бинарным, если

- a) **количество узлов** может быть либо пустым, либо состоять из корня с двумя другими бинарными поддеревьями
- b) каждый узел имеет не менее двух предков
- c) от корня до листа не более двух уровней
- d) от корня до листа не менее двух уровней

56. Бинарное дерево можно представить

- a) **с помощью указателей**
- b) **с помощью массивов**
- c) с помощью индексов
- d) правильного ответа нет

57. Какой метод поиска представлен в следующем фрагменте REPEAT
I:=I+1 UNTIL (A[I]=X) OR (I=N);

- a) **последовательный**
- b) двоичный
- c) восходящий
- d) нисходящий
- e) смешанный

58. Какой метод поиска представлен в следующем фрагменте
REPEAT K:=(I+J)DIV 2; IF X>A[K] THEN I=K+1 ELSE J:=K-1;
UNTIL (A[K]=X) OR (I>J);

- a) последовательный
- b) **бинарный**

- c) восходящий
- d) нисходящий
- e) смешанный

59. Реализация поиска в линейном списке выглядит следующим образом

- a) **WHILE (P<>NIL) AND (P^.KEY<>X) DO P:=P^.NEXT**
- b) WHILE (P<>NIL) DO P:=P^.NEXT
- c) WHILE AND (P^.KEY<>X) DO P:=P^.NEXT
- d) WHILE (P<>NIL) AND (P^.KEY<>X) P:=P^.NEXT
- e) WHILE (P<>NIL P^.KEY<>X) DO P:=P^.NEXT

60. Как называются предки узла, имеющие уровень на единицу меньше уровня самого узла

- a) детьми
- b) **родителями**
- c) братьями
- d)

61. В графах общая идея поиска в глубину состоит в следующем:

a) **Поиск начинаем с некоторой фиксированной вершины v_0 . Затем выбираем произвольную вершину u , смежную с v_0 , и повторяем просмотр от u . Предположим, что находимся в некоторой вершине v . Если существует ещё не просмотренная вершина u , $u-v$, то рассматриваем её, затем продолжаем поиск с нее. Если не просмотренной вершины, смежной с v , не существует, то возвращаемся в вершину, из которой попали в v , и продолжаем поиск (если $v=v_0$, то поиск закончен);**

b) Поиск начинаем с некоторой фиксированной вершины v_0 . Затем выбираем произвольную вершину u , смежную с v_0 , и повторяем просмотр от u . Предположим, что находимся в некоторой вершине v . Если существует ещё не просмотренная вершина u , $u-v$, то рассматриваем её, затем продолжаем поиск с нее. Если не просмотренной вершины, смежной с v , не существует, то возвращаемся в вершину, из которой попали в v , и продолжаем поиск (если $v=u$, то поиск закончен);

c) Поиск начинаем с некоторой фиксированной вершины v_0 . Затем выбираем произвольную вершину u , смежную с v_0 , и повторяем просмотр от u . Предположим, что находимся в некоторой вершине v . Если существует ещё не просмотренная вершина u , то рассматриваем её, затем продолжаем поиск с нее. Если не просмотренной вершины, смежной с v , не существует, то возвращаемся в вершину, из которой попали в v , и продолжаем поиск (если $v=v_0$, то поиск закончен).

62. Стандартным способом устранения рекурсии при поиске в глубину является использование:

- a) массива;
- b) очереди;
- c) **стека;**
- d) циклического списка.

63. При поиске в ширину используется:

- a) массив;
- b) **очередь;**
- c) стек;
- d) циклический список

64. В последовательном файле доступ к информации может быть

- a) **только последовательным**
- b) как последовательным, так и произвольным
- c) произвольным
- d) прямым

65. Граф – это

- a) **Нелинейная структура данных, реализующая отношение «многие ко многим»;**
- b) Линейная структура данных, реализующая отношение «многие ко многим»;
- c) Нелинейная структура данных, реализующая отношение «многие к одному»;
- d) Нелинейная структура данных, реализующая отношение «один ко многим»;
- e) Линейная структура данных, реализующая отношение «один ко многим».

66. Узлам (или вершинам) графа можно сопоставить:

- a) отношения между объектами;
- b) **объекты;**
- c) связи
- d) типы отношений
- e) множества

67. Рёбрам графа можно сопоставить:

- a) связи
- b) типы отношений
- c) множества
- d) объекты;
- e) **отношения между объектами;**

68. Граф, содержащий только ребра, называется.

- a) ориентированным
- b) **неориентированным**
- c) простым
- d) смешанным

69. Граф, содержащий только дуги, называется.

- a) **ориентированным**
- b) неориентированным
- c) простым
- d) смешанным

70. Граф, содержащий дуги и ребра, называется.

- a) ориентированным
- b) неориентированным
- c) простым
- d) **смешанным**

71. Есть несколько способов представления графа в ЭВМ. Какой из способов приведенных ниже не относится к ним.

- a) матрица инциденций;
- b) матрица смежности;
- c) список ребер;
- d) **массив инцидентности.**

72. Если последовательность вершин v_0, v_1, \dots, v_p определяет путь в графе G , то его длина определяется:

- a) $\sum_{i=1}^p a(v_{i-1}, v_i)$; **правильный ответ**
- b) $\sum_{i=1}^p a(v_{i+1}, v_i)$;
- c) $\sum_{i=2}^p a(v_{i-1}, v_i)$;
- d) $\sum_{i=0}^p a(v_{i-1}, v_i)$.

73. Каким образом осуществляется алгоритм нахождения кратчайшего пути от вершины s до вершины t

- a) **нахождение пути от вершины s до всех вершин графа**
- b) нахождение пути от вершины s до заданной вершины графа
- c) нахождение кратчайших путей от вершины s до всех вершин графа
- d) нахождение кратчайшего пути от вершины s до вершины t графа
- e) нахождение всех путей от каждой вершины до всех вершин графа

74. Суть алгоритма Дейкстры - нахождения кратчайшего пути от вершины s до вершины t заключается

- a) **вычислении верхних ограничений $d[v]$ в матрице весов дуг $a[u,v]$ для u, v**
- b) вычислении верхних ограничений $d[v]$
- c) вычислении верхних ограничений в матрице весов дуг $a[u,v]$
- d) вычислении нижних ограничений $d[v]$ в матрице весов дуг $a[u,v]$ для u, v

75. Улучшение $d[v]$ в алгоритме Форда- Беллмана производится по формуле

- a) $D[v] := D[u] + a[u, v]$
- b) $D[v] := D[u] - a[u, v]$
- c) $D[v] := a[u, v]$
- d) $D[v] := D[u]$

76. Строка представляет собой

- a) **конечную линейно-упорядоченную последовательность простых данных символьного типа**
- b) конечную последовательность простых данных символьного типа
- c) конечную последовательность простых данных
- d) последовательность данных символьного типа

77. Граф, содержащий только ребра, называется

- a) ориентированным
- b) **неориентированным**
- c) простым
- d) связным

78. Граф, содержащий только дуги, называется

- a) **ориентированным**
- b) неориентированным
- c) простым
- d) связным

79. Граф, содержащий ребра и дуги, называется

- a) неориентированным
- b) простым
- c) **смешанным**
- d) связным

80. Путь(цикл), который содержит все ребра графа только один раз, называется

- a) **Эйлеровым**
- b) Гамильтоновым
- c) декартовым
- d) замкнутым

Министерство образования и науки Российской Федерации
Нижекамский химико-технологический институт (филиал) федерального
государственного бюджетного образовательного учреждения
высшего образования
«Казанский национальный исследовательский технологический университет»

Направление подготовки 09.03.01 Информатика и вычислительная
техника

Темы расчетно-графических работ
по дисциплине Структуры и алгоритмы компьютерной обработки данных

Расчетно-графическая работа №1

Каждую из представленных ниже задач выполнить используя

- *перебор с возвратом;*
- *перебор с барьером;*
- *бинарный поиск;*

*Путем численных экспериментов произвести сравнительный анализ
временной трудоемкости каждого алгоритма*

Вариант 1. Имеется массив окружностей. Написать функцию, определяющую первую окружность в массиве, находящуюся на расстоянии заданной величины.

Входные параметры функции:

- точка на плоскости,
- массив окружностей,
- количество элементов в массиве.
- Расстояние до искомого элемента

Выходные параметры: -нет.

Выдаваемое значение: -номер искомого элемента

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Рекомендуется написать дополнительно функцию, вычисляющую расстояние между двумя точками.

Вариант 2. Имеется массив прямоугольников со сторонами параллельными осям координат. Написать функцию, определяющую первый прямоугольник с заданной площадью.

Входные параметры функции:

- массив прямоугольников,
- количество элементов в массиве.

Выходные параметры: -нет.

Выдаваемое значение: -номер искомого элемента

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую площадь прямоугольника.

Вариант 3. Имеется массив прямоугольников со сторонами параллельными осям координат. Написать функцию, определяющую первый прямоугольник с заданным периметром.

Входные параметры функции:

- массив прямоугольников,
- количество элементов в массиве,
- периметр искомого прямоугольника

Выходные параметры: -нет.

Выдаваемое значение: -номер искомого элемента

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую периметр прямоугольника

Вариант 4. Имеется массив окружностей. Написать функцию, определяющую первую окружность с заданной площадью.

Входные параметры функции:

- массив окружностей,
- количество элементов в массиве,
- площадь искомой окружности

Выходные параметры: -нет.

Выдаваемое значение: -номер искомого элемента

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Вариант 5. Имеется массив векторов на плоскости. Написать функцию, определяющую первый вектор с заданной длиной.

Входные параметры функции:

- массив векторов,
- количество элементов в массиве,
- длина искомого вектора

Выходные параметры: -нет.

Выдаваемое значение: -номер искомого элемента

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую длину вектора

Вариант 6. Имеется массив векторов на плоскости. Написать функцию, определяющую первый вектор с заданным углом наклона к оси ОХ.

Входные параметры функции:

- массив векторов,
- количество элементов в массиве,
- угол наклона искомого вектора

Выходные параметры: -нет.

Выдаваемое значение: -номер искомого элемента

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую угол наклона вектора

Расчетно-графическая работа №2

Программно реализовать алгоритмы поиска образа в тексте:

- прямой поиск;
- алгоритм Кнута, Мориса, Пратта;
- алгоритм Боуера-Мура

Путем численных экспериментов произвести анализ временной сложности этих алгоритмов.

Расчетно-графическая работа №3

Каждую из представленных ниже задач выполнить используя

- **сортировку прямым включением;**
- **сортировку прямым выбором;**
- **сортировку прямым обменом;**
- **быструю сортировку.**

Путем численных экспериментов произвести сравнительный анализ временной трудоемкости каждого алгоритма

Вариант 1. Имеется массив окружностей. Написать функцию, упорядочивающую массив в порядке удаленности их от некоторой точки на плоскости.

Входные параметры функции:

- точка на плоскости,
- массив окружностей,

- количество элементов в массиве.

Выходные параметры: -нет.

Выдаваемое значение: -нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Рекомендуется написать дополнительно функцию, вычисляющую расстояние между двумя точками.

Вариант 2. Имеется массив прямоугольников со сторонами параллельными осям координат. Написать функцию, упорядочивающую массив в порядке возрастания площадей прямоугольников.

Входные параметры функции:

- массив прямоугольников,
- количество элементов в массиве.

Выходные параметры: -нет.

Выдаваемое значение: -нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую площадь прямоугольника.

Вариант 3. Имеется массив прямоугольников со сторонами параллельными осям координат. Написать функцию, упорядочивающую массив по убыванию их периметров.

Входные параметры функции:

- массив прямоугольников,
- количество элементов в массиве,

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую периметр прямоугольника

Вариант 4. Имеется массив окружностей. Написать функцию, упорядочивающую массив по возрастанию радиусов окружностей.

Входные параметры функции:

- массив окружностей,
- количество элементов в массиве,

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Вариант 5. Имеется массив векторов на плоскости. Написать функцию, упорядочивающую массив по возрастанию длин этих векторов.

Входные параметры функции:

- массив векторов,
- количество элементов в массиве,

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую длину вектора

Вариант 6. Имеется массив векторов на плоскости. Написать функцию, упорядочивающую массив по возрастанию угла наклона вектора к оси ОХ.

Входные параметры функции:

- массив векторов,
- количество элементов в массиве,

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую угол наклона вектора

Расчетно-графическая работа №4

Вариант 1. Имеется массив окружностей, размещенный в файле. Написать функцию, упорядочивающую массив в порядке удаленности их от некоторой точки на плоскости, используя алгоритмы сортировки слиянием.

Входные параметры функции:

- имя входного файла
- имя выходного файла

Выходные параметры: -нет.

Выдаваемое значение: -нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
 - окружность (в виде структуры) с заданным центром (точкой) и радиусом.
- Рекомендуется написать дополнительно функцию, вычисляющую расстояние между двумя точками.

Вариант 2. Имеется массив прямоугольников со сторонами параллельными осям координат, размещенный в файле. Написать функцию, упорядочивающую массив в порядке возрастания площадей прямоугольников, используя алгоритмы сортировки слиянием.

Входные параметры функции:

- имя входного файла
- имя выходного файла

Выходные параметры: -нет.

Выдаваемое значение: -нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую площадь прямоугольника.

Вариант 3. Имеется массив прямоугольников со сторонами параллельными осям координат, размещенный в файле. Написать функцию, упорядочивающую массив по убыванию их периметров, используя алгоритмы сортировки слиянием.

Входные параметры функции:

- имя входного файла
- имя выходного файла

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую периметр прямоугольника

Вариант 4. Имеется массив окружностей, размещенный в файле. Написать функцию, упорядочивающую массив по возрастанию радиусов окружностей, используя алгоритмы сортировки слиянием.

Входные параметры функции:

- имя входного файла
- имя выходного файла

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Вариант 5. Имеется массив векторов на плоскости, размещенный в файле. Написать функцию, упорядочивающую массив по возрастанию длин этих векторов, используя алгоритмы сортировки слиянием.

Входные параметры функции:

- имя входного файла
- имя выходного файла

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую длину вектора

Вариант 6. Имеется массив векторов на плоскости, размещенный в файле. Написать функцию, упорядочивающую массив по возрастанию угла наклона вектора к оси ОХ, используя алгоритмы сортировки слиянием.

Входные параметры функции:

- имя входного файла
- имя выходного файла

Выходные параметры: -нет.

Выдаваемое значение: нет.

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую угол наклона вектора

Расчетно-графическая работа №5

Вариант 1. Имеется граф, вершинами которого являются окружности. Написать функцию, определяющую окружность в графе, находящуюся на расстоянии заданной величины.

Входные параметры функции:

- точка на плоскости,
- граф,
- Расстояние до искомого элемента

Выходные параметры: -нет.

Выдаваемое значение: -искомая вершина

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Рекомендуется написать дополнительно функцию, вычисляющую расстояние между двумя точками.

Вариант 2. Имеется граф, вершинами которого являются прямоугольники со сторонами параллельными осям координат. Написать функцию, определяющую первый прямоугольник с заданной площадью.

Входные параметры функции:

- граф,
- искомая площадь.

Выходные параметры: -нет.

Выдаваемое значение: -искомая вершина

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую площадь прямоугольника.

Вариант 3. Имеется граф, вершинами которого являются прямоугольники со сторонами параллельными осям координат. Написать функцию, определяющую первый прямоугольник с заданным периметром.

Входные параметры функции:

- граф
- периметр искомого прямоугольника

Выходные параметры: -нет.

Выдаваемое значение: -искомая вершина

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания вершин прямоугольника,
- прямоугольник (в виде структуры) с заданными противоположными углами (точками).

Рекомендуется написать дополнительно функцию, вычисляющую периметр прямоугольника

Вариант 4. Имеется граф, вершинами которого являются окружности.

Написать функцию, определяющую первую окружность с заданной площадью.

Входные параметры функции:

- граф
- площадь искомой окружности

Выходные параметры: -нет.

Выдаваемое значение: -искомая вершина

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания центра окружности,
- окружность (в виде структуры) с заданным центром (точкой) и радиусом.

Вариант 5. Имеется граф, вершинами которого являются вектора на плоскости. Написать функцию, определяющую первый вектор с заданной длиной.

Входные параметры функции:

- граф
- длина искомого вектора

Выходные параметры: -нет.

Выдаваемое значение: -искомая вершина

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую длину вектора

Вариант 6. Имеется граф, вершинами которого являются вектора на плоскости. Написать функцию, определяющую первый вектор с заданным углом наклона к оси ОХ.

Входные параметры функции:

- граф
- угол наклона искомого вектора

Выходные параметры: -нет.

Выдаваемое значение: -искомая вершина

Для решения этой задачи ввести типы данных:

- точка (в виде структуры) –для описания начала и конца вектора,
- вектор (в виде структуры) с заданным началом (точкой) и концом.

Рекомендуется написать дополнительно функцию, вычисляющую угол наклона вектора