

Екзаменаційні питання по дисципліні «Програмування на мові java»

Теоретичні питання

1. Історія створення мови програмування java.
2. Проблема мобільності програмного забезпечення. Основні підходи до розробки мобільного програмного забезпечення.
3. Поняття віртуальних обчислювальних середовищ. Віртуальні машини.
4. Основные особенности Ява - машины. Понятие Рі-кода. «Запуск» Ява программы.
5. Технология получения исполняемой Ява программы. Понятие JDK.
6. Основная структура программы языка программирования Ява. Основная единица программы. Классы и методы. Метод main;
7. Зарезервовані ключові слова мови Ява. Коментарі. Складені оператори.
8. Правила запису імен «операторів» і іменування ідентифікаторів класів, об'єктів, змінних на мові Ява.
9. Основные простые типы данных, поддерживаемых в языке программирования Ява. Правила записи литералов (констант).
10. Змінні, правила їх іменування і способи оголошення в мові Ява.
11. Прості типи змінних для роботи з числовими даними. Способи оголошення і області допустимих значень.
12. Простые типы переменных для работы с символьными данными. Способы объявления и области допустимых значений.
13. Прості типи змінних для роботи з логічними даними. Способи оголошення і області допустимих значень.
14. Согласование типов переменных. Правила допустимых преобразований типов. «Временное» преобразование типов.
15. Одні мірні масиви в мові Ява. Способи оголошення типу і кількості елементів.
16. Много мерные массивы в языке Ява. Способы объявления типа и количества элементов. Ключевое слово new.
17. Завдання початкових значень для змінних і масивів на мові Ява.
18. Поняття області видимості змінних мови Ява.
19. Статичні і динамічні змінні і об'єкти. Поняття «часу життя» змінної.
20. Основні оператори мови програмування Ява. Групи операторів. Оператор привласнення.
21. Основные арифметические операции в языке Ява.
22. Основні бітові операції в мові Ява.
23. Основні логічні операції в мові Ява.
24. Основні операції порівняння (дії) в мові Ява.
25. Теральний оператор в мові Ява. Його блок схема.
26. Пріоритети операцій з простими типами даних, перетворення типів змінних (дії над значеннями) в мові Ява.
27. Оператори управління програмою. Складений оператор. Мітка складеного оператора. Оператор break;
28. Оператори управління програмою. Оператори if-else, break, return.
29. Оператори управління програмою. Оператори циклів while, do-while, for. Оператор continue
30. Понятие о классах, методах и свойствах. Объявление класса и метода. Метод main.
31. Объявление метода. Вызов метода. Передача параметров методу. Скрытие переменных представителей (ссылка this).
32. Поняття інкапсуляції. Область видимості змінних і методів. Опис public і private

33. Конструкторы. Вызов конструктора. Совмещение методов в классе. this в конструкторе.
34. Поняття спадкоємства (extends супер_клас). Супер клас, клас нащадок. Заміщення методів. Посилання super і this.
35. Динамічне і статичне призначення методів і властивостей.. Спосіб виклику динамічних і статичних методів.
36. Модификаторы методов и переменных - final, finalize, static, abstract – назначение и способы применения.
37. Поняття про пакет (package). Простір імен класів. Оператор import. Обмеження доступу (private, protected, public).
38. Інтерфейси. Оператор implements. Використання спільно використовуваних змінних і методів.
39. Обробка виключень. Типи виключень. Перехоплені і не перехваленні виключення (try . catch . finally, throw - throws).
40. Поняття про утиліти мови програмування Ява. Прості оболонки для атомарних типів Number (Double, Float, Integer, Long), Character, Boolean. Призначення і огляд основних методів.
41. Поняття про утиліти мови програмування Ява. Утиліти Runtime, System, Date, Math, Random. Призначення і огляд основних методів.
42. Робота з рядками. Оболонка String. Основні методи.
43. Робота з легковагими процесами. Пріоритет під процесу, синхронізація під процесів, обмін повідомленнями.
44. Організація введення - висновку засобами мови програмування Ява. Поняття про вхідні і вихідні потоки.
45. Мережеві засоби Ява. Сокети і дейтаграммы.
46. Аплети. Призначення, правила запуску і використання.
47. Аплеты. Обзор классов и методов работы с графическими объектами.
48. Аплети. Принципи обробки подій.
49. Огляд програм пакету JDK. Призначення програм java, javac, appletviewer, jar.
50. Принцип само документирования библиотечных классов языка программирования Ява. Генератор документации javadoc – назначение и принцип использования.

Практичні завдання

Розробити клас, набір методів (конструктор і мінімум два методи) для програмної моделі заданого об'єкту. Опис об'єкту і його основних властивостей приводиться нижче. Привести фрагмент програми (public static main), що використовує об'єкти розробленого класу.

1. Об'єкт «комплексні числа». Операції визначаються по загальний прийнятим формулам. Передбачити можливість операції привласнення, складання, множення і перекладу в текстовий рядок поточних значень. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
2. Объект «комплексные числа». Операции определяются по обще принятым формулам. Предусмотреть возможность операции присваивания, вычитания, умножения и перевода в текстовую строку текущих значений. Конструктор должен позволить создавать объекты без и с начальной инициализацией.
3. Об'єкт «комплексні числа». Операції визначаються по загальний прийнятим формулам. Передбачити можливість операції привласнення, складання, ділення і перекладу в текстовий рядок поточних значень. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.

4. Об'єкт «комплексні числа». Операції визначаються по загальний прийнятим формулам. Передбачити можливість операції привласнення, складання, множення і перекладу в показову ($A \cdot \ell^{i \cdot \varphi}$) форму з можливістю роздруку на консоль. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
5. Об'єкт «вектор на площині» заданий в системі декартових координат. Почало вектора розташовано на початку координат. Операції визначаються згідно загальний прийнятих формул лінійної (векторної) алгебри. Передбачити можливість операції привласнення, складання, скалярного множення і роздруку координат поточних значень. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
6. Об'єкт «вектор на площині» заданий в системі декартових координат. Почало вектора розташовано на початку координат. Операції визначаються згідно загальний прийнятих формул лінійної (векторної) алгебри. Передбачити можливість операції привласнення, віднімання, скалярного множення і роздруку координат поточних значень. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
7. Об'єкт «вектор на площині» заданий в системі декартових координат. Почало вектора розташовано на початку координат. Операції визначаються згідно загальний прийнятих формул лінійної (векторної) алгебри. Передбачити можливість операції привласнення, порівняння модулів, скалярного множення і роздруку координат поточних значень. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
8. Объект «вектор на плоскости» заданный в системе декартовых координат. Начало вектора расположено в начале координат. Операции определяются согласно обще принятых формул линейной (векторной) алгебры. Предусмотреть возможность операции присваивания, нахождения угла между векторами, скалярного умножения и распечатки координат текущих значений. Конструктор должен позволить создавать объекты без и с начальной инициализацией.
9. Об'єкт «рівнобедрений трикутник заданий довжинами сторін». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод, що визначає існує або такий трикутник. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
10. Об'єкт «рівносторонній трикутник заданий довжинами сторін». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод, що визначає існує або такий трикутник. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
11. Об'єкт «прямокутний трикутник заданий довжинами сторін». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод, що визначає існує або такий трикутник. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
12. Об'єкт «рівнобедрений трикутник заданий завдовжки рівнобедреною стороною і кутом між ними». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод, що відповідає на питання – гостро або тупо вугільним є заданий трикутник. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
13. Об'єкт «трикутник заданий завдовжки двох стороною і кутом між ними». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод, що відповідає на питання – гостро або тупо вугільним є заданий трикутник. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.
14. Об'єкт «прямокутник заданий довжинами двох сторін». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод,

[illegible]

множині. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією.

25. Об'єкт «прямокутник заданий довжинами двох сторін». Передбачити можливість операції привласнення, визначення площі і периметра, а так само логічний метод, що відповідає на питання – чи міститься, вказаний параметрами методу прямокутник, усередині прямокутника. Конструктор повинен дозволити створювати об'єкти і з початковою ініціалізацією