

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ
им. проф. М. А. БОНЧ-БРУЕВИЧА»

(СПбГУТ)

В. П. Грибачев

***КРОССПЛАТФОРМЕННОЕ
ПРОГРАММИРОВАНИЕ.
ВОПРОСНИК ДЛЯ ПОДГОТОВКИ К
ЭКЗАМЕНУ***

УЧЕБНОЕ-МЕТОДИЧЕСКОЕ ПОСОБИЕ

СПб ГУТ)))

САНКТ-ПЕТЕРБУРГ

2018

Лекция 1.

Введение в кроссплатформенное программирование.

1. *Какое ПО называют кроссплатформенным*
2. *Какие технологии кроссплатформенности вы знаете и чем они друг от друга отличаются.*
3. *Чем компиляция отличается от интерпретации, какие языки программирования интерпретируются, какие компилируются.*
4. *Чем эмуляция среды выполнения отличается от скриптового ПО.*
5. *Что такое бай-код, чем он отличается от машинного кода.*
6. *Что такое виртуальная машина и зачем она нужна.*
7. *Что такое эмулятор среды выполнения и чем он отличается от виртуальной машины.*
8. *Преимущества языка Java.*
9. *Недостатки языка Java.*
10. *Особенности языка Java в сравнении с C++.*
11. *На чем основана концепция безопасности языка Java.*
12. *Какие типы Java – приложений вы знаете.*

Лекция 2.

Этапы разработки java-приложений.

13. *Процесс компиляции Java – программ. Что содержит файл xxx.classes.*
14. *Что входит в состав пакета JDK.*

Лекция 3.

Переменные и типы данных.

15. *Набор типов языка. Различие между ссылочными и примитивными типами.*
16. *Особенности целочисленных типов Char и Integer.*
17. *Особенности логических типов.*
18. *Особенности вещественных типов.*
19. *Особенности вычислений с большими числами*
20. *Различие между объявлением и инициализацией переменной.*
21. *Особенности строкового типа. Инициализация строк. Функции работы со строками.*

Лекция 4.

Операторы и функции.

- 22. Объявление и определение функций.*
- 23. Передача параметров в функции. Возвращаемое значение функции.*
- 24. Глобальные и локальные области видимости переменных.*
- 25. Общая структура программы. Операторы ветвлений и циклов.*

Лекция 5.

Объектно ориентированное программирование.

- 26. Классы. Объявление и инициализация.*
- 27. Три «кита» ООП – инкапсуляция, наследование и полиморфизм.*
- 28. Поля и методы классов. Перегрузка методов.*
- 29. Статические поля и методы класса.*
- 30. Режимы доступа к полям и методам класса.*
- 31. Особенности различных режимов доступа с точки зрения области видимости.*
- 32. Отличие перегрузки функций от переопределения.*
- 33. Отличие классов от интерфейсов.*

Лекция 6.

Массивы и строки.

- 34. Типы массивов. Объявление массива, инициализация и обращение к элементам массива.*
- 35. Функции копирования массивов. Функции сравнения массивов.*
- 36. Правило приведения типов. Различие между явным приведением типа и указанием модификатора типа.*
- 37. Функции работы со строками. Сравнение строк.*

Лекция 7.

Организация ввода-вывода данных.

- 38. Организация ввода-вывода данных. Консольный ввод-вывод: средства ввода-вывода.*
- 39. Файловый ввод-вывод. Функции стандартной библиотеки ввода-вывода.*
- 40. Методы класса File и их назначение.*

41. Основные классы и методы библиотеки `nio`. Преимущества библиотеки `nio`.
42. Классы потокового ввода/вывода из пакета `java.io`.
43. Основные методы классов потокового ввода/вывода из пакета `java.io`.

Лекция 8.

Обработка исключений.

44. Иерархия классов исключений `java`. Основные классы исключений.
45. Обработка исключений.
46. Выброс пользовательских исключений.
47. Как написать собственный класс исключений.

Лекция 9.

Потоки.

48. Что такое процесс и чем он отличается от потока.
49. Специальный класс для создания и управления потоками `java.lang.Thread`.
50. Константы, определяющие состояние потока.
51. Методы класса `java.lang.Thread`, предназначенные для управления потоками.
52. Использование интерфейса `java.lang.Runnable`. Реализация многопоточного приложения с использованием интерфейса `java.lang.Runnable`.
53. Сравнение метода создания многопоточного приложения с использованием класса `java.lang.Thread` и интерфейса `java.lang.Runnable`.
54. Синхронизация потоков с помощью оператора `synchronized`.
55. Синхронизация потоков с помощью оператора семафоров.

Лекция 10.

Подключаемые библиотеки `java`.

56. Общие принципы использования графических библиотек в `java`.
57. Иерархия классов библиотеки `AWT` и `Swing`.
58. Различия между библиотеками `AWT` и `Swing`.
59. Создание простейших приложений с графическим интерфейсом.
60. Устройство системы обработки событий `AWT`.
61. Классы библиотеки `Swing`.
62. Реализация паттерна `MVC` в библиотеке `Swing`.
63. Пример простого шаблона приложения, использующего `Swing`.