|  |  |
| --- | --- |
| **Описание: E:\!Кафедра ПИКС\Логотип БГУИР\Символика.jpg** | **Описание: E:\!Кафедра ПИКС\Логотип ПИКС\17 мая 2013\Логотип ПИКС_3.jpg** |

**ВОПРОСЫ К ЭКЗАМЕНУ**

**по дисциплине**

**«РАЗРАБОТКА ПРИЛОЖЕНИЙ НА ЯЗЫКАХ ВЫСОКОГО УРОВНЯ»**

**Весенний семестр 2023-2024 учебного года**

**Специальность 1-39 03 01 «Программируемые мобильные системы)»**

**(группы 113871)**

1. Чем отличается интерпретируемый язык программирования от компилируемого?
2. Объясните разницу между процедурным, объектно-ориентированным и функциональным программированием.
3. Какие основные принципы лежат в основе языка Python?
4. Почему использование высокоуровневых языков программирования предпочтительнее низкоуровневых при разработке программ?
5. Что такое модуль в программировании и для чего он используется?
6. Какая разница между встроенными модулями и сторонними библиотеками?
7. Каким образом вы можете установить и использовать стороннюю библиотеку в вашем проекте на языке Python?
8. Каковы преимущества использования стандартной библиотеки языка программирования?
9. Объясните разницу между массивом и связным списком. В каких ситуациях каждая из них предпочтительнее?
10. Что такое стек и очередь? Какие операции можно выполнить с ними?
11. Как работает ассоциативный массив? Приведите примеры его использования.
12. Какие базовые структуры данных чаще всего используются для реализации алгоритмов сортировки?
13. Объясните основные принципы работы алгоритмов сортировки
14. В чем отличие между линейным и бинарным поиском? Когда каждый из них применяется?
15. Как оценивается эффективность алгоритма? Что такое асимптотическая сложность?
16. Какой алгоритм сортировки будет наиболее эффективным для сортировки больших объемов данных и почему?
17. Что такое графический интерфейс пользователя (GUI) и зачем он нужен?
18. Какие основные элементы управления чаще всего используются в GUI-программировании?
19. Каковы основные шаги создания GUI-приложения?
20. Какие преимущества предоставляют GUI-интерфейсы по сравнению с текстовыми интерфейсами?
21. Какие инструменты и библиотеки можно использовать для создания GUI-приложений на языке Python?
22. Чем отличаются фреймворки для разработки GUI на разных платформах (например, Tkinter, PyQt, WxPython)?
23. Какие критерии следует учитывать при выборе инструмента для разработки GUI-приложения?
24. Каковы преимущества использования графических редакторов для создания пользовательского интерфейса?
25. Что такое тестирование программного обеспечения и зачем оно нужно?
26. Какие виды тестирования существуют? Когда каждый из них применяется?
27. Что такое модульное тестирование и как оно отличается от интеграционного тестирования?
28. Каковы основные этапы проведения тестирования программного обеспечения?
29. Чем отличается ручное тестирование от автоматизированного?
30. Какие инструменты можно использовать для автоматизации тестирования программного обеспечения?
31. Какие преимущества и недостатки автоматизации тестирования?
32. Какие критерии помогут определить, какие тесты следует автоматизировать?
33. Что такое система контроля версий и зачем она нужна?
34. Какие основные команды используются в Git для управления версиями файлов?
35. Что такое ветвление и слияние в Git и как они используются в рабочем процессе разработки?
36. Каковы основные принципы работы с репозиториями в Git?
37. Что такое конфликт слияния в Git и как он решается?
38. Какие инструменты предоставляет Git для разрешения конфликтов?
39. Какие стратегии можно использовать для предотвращения конфликтов при слиянии веток?
40. Каковы лучшие практики работы с слияниями в Git в командной разработке?
41. Что такое параллельное выполнение программ и в чем его преимущества?
42. Какие методы параллельного выполнения программ существуют?
43. Какие проблемы могут возникнуть при параллельном выполнении программ и как их избежать?
44. Какие инструменты и технологии используются для управления параллельным выполнением программ?
45. Что такое база данных и какие типы баз данных существуют?
46. Какие операции можно выполнять с базой данных?
47. Каковы основные принципы нормализации баз данных и зачем она нужна?
48. Какие языки программирования и инструменты можно использовать для работы с базами данных?
49. Что такое сетевое программирование и зачем оно нужно?
50. Какие протоколы чаще всего используются в сетевом программировании?
51. Каковы основные шаги создания клиент-серверного приложения?
52. Какие проблемы могут возникнуть при сетевом взаимодействии и как их решать?
53. Что такое нейронная сеть и зачем она используется?
54. Какие типы нейронных сетей существуют и в чем их отличие?
55. Как происходит обучение нейронной сети? Какие методы обучения существуют?
56. Какие приложения имеют нейронные сети в современном мире и как они улучшают работу различных систем?
57. Какие математические концепции лежат в основе работы нейронных сетей?
58. Что такое обучение нейронной сети с учителем и без учителя? Какие задачи решаются каждым из этих методов?

Вопросы разработали:

ЯЧИН Николай Сергеевич – магистр техники и технологии, старший преподаватель кафедры ПИКС

СМОРГУН Евгений Святославович – ассистент кафедры ПИКС