**Вопросы к экзамену:**

1. Определение системы мониторинга и её основные задачи.
2. Ключевые компоненты любой системы мониторинга.
3. Понятие метрики и её роль в системах мониторинга.
4. Основные типы метрик Prometheus: Counter, Gauge, Histogram, Summary.
5. Принцип работы алертинга в системах мониторинга.
6. Понятие временных рядов и их применение в мониторинге.
7. Основные возможности и особенности Prometheus.
8. Формат данных, используемый Prometheus для хранения метрик.
9. Роль exporters в экосистеме Prometheus.
10. Принцип работы PromQL (языка запросов Prometheus).
11. Какие данные можно извлекать с помощью PromQL?
12. Механизм работы правил алертов в Grafana.
13. Основные принципы настройки алертинга.
14. Разница между статусами Firing, Pending и Normal.
15. Интеграция Telegram с системой алертинга.
16. Как проверить корректность работы алертинга?
17. Как добавить теги к кастомным метрикам в формате для Prometheus?
18. Как добавить теги к кастомным метрикам в формате для InfluxDB?
19. Как обеспечить отказоустойчивость системы мониторинга?
20. Особенности мониторинга распределенных систем.
21. Преимущества использования Prometheus для мониторинга.
22. Ограничения и недостатки Prometheus.
23. Основное назначение Grafana в системах мониторинга.
24. Поддерживаемые источники данных в Grafana.
25. Этапы настройки нового источника данных в Grafana.
26. Принцип создания дашбордов в Grafana.
27. Основные типы панелей визуализации в Grafana.
28. Как настроить легенду на графике в Grafana?
29. Основные характеристики базы данных InfluxDB.
30. Разница между InfluxDB и Prometheus.
31. Основные команды для работы с данными в InfluxDB.
32. Принцип работы с тегами в InfluxDB.
33. Значение временной метки в записях InfluxDB.
34. Интеграция InfluxDB с Grafana.
35. Настройка Telegraf для работы с InfluxDB.
36. Преимущества использования InfluxDB для хранения метрик.
37. Ограничения InfluxDB в сравнении с другими решениями.
38. Примеры сервисов, поддерживаемых Telegraf.
39. Преимущества использования Telegraf для сбора метрик.
40. Настройка базовой конфигурации агента Telegraf.
41. Какую информацию содержат логи приложений?
42. Почему логи важны для диагностики ошибок в работе программного обеспечения?
43. Какие уровни логирования существуют.
44. В чем заключается разница между уровнями логирования Trace и Debug?
45. Для чего используется уровень логирования Info?
46. Когда применяется уровень логирования Warning?
47. В чем особенность уровня логирования Error?
48. Что означает уровень логирования Fatal?
49. Что такое "белый шум" в контексте системы алертинга?
50. Какие компоненты системы можно отслеживать с помощью мониторинга?
51. Как мониторинг помогает предотвратить перегрузку сервера?
52. Какие проблемы можно предсказать с помощью сбора статистики?
53. Что такое PromQL и для чего он используется?
54. Почему важно мониторить состояние виртуальных машин?
55. Как настроить алерты для мониторинга виртуальных машин?
56. В чем разница между логированием и мониторингом?
57. Как получить информацию о логах, которые пишет контейнер?
58. Подход к организации хранения секретных данных с помощью .env
59. Какие инструменты предоставляет Bitrix для работы с базами данных, и как осуществляется оптимизация запросов?
60. Какие технологии и подходы применяются в Bitrix для обеспечения высокой отказоустойчивости и масштабируемости сайтов?

**Примеры практических заданий:**

1. Визуализировать распределение кодов ответа сайта при обращении к эндпоинтам на круговой диаграмме.
2. Создать нагрузочный тест на любые два GET и POST запроса с сайту/API, создать для этого последовательный сценарий. Первый запрос должен выполняться в 4 раз реже, чем второй. Тесты должны быть успешными.
3. Настроить запись данных в InfluxDB через Telegraf.