Правила успешного
собеседования:===

Правило минус-ноль-плюс (обойти конкурентов)

    * НЕДОСТАТОЧНО дать правильный ответ

Виды тестирования - каждый тест является частью РАЗНЫХ классификаций

    * Login/Password

        - functionality / regression / build acceptance

        - black box/gray/white

        - manual/automated

        - module/integration/system

        - negative/positive

Для каждого ответа на вопрос, особенно про типы тестирования, дайте пример из опыта

Для каждого ответа на вопрос расскажите какие были проблемы и как их решали

Показать разные точки зрения - очень даже неплохо

Спросите как у них в компании-в команде к этому вопросу подходят

Ответ должен звучать естественно, не заученно

По времени - коротко, по существу. Спросите не хотят ли они более подробного ответа.

===
======

Уровни Тестирования

        - Unit/Component Testing

        - Integration Testing (2+ units)

        - System - вся система целиком

        - User acceptance

Обычные вопросы по Тестированию ПО на собеседовании

- What are Bug Report components?
- What is a Test Case?
- What is a test plan?
- If you log a bug and the developer says she cant reproduce it, what should you do?
- If you find a bug and the developer says it is as-designed, what would you do?
- How do you test a toaster?
- How do you test a banking ATM machine?
- What is use case?
- How do you determine when you have done enough testing?
- Describe the bug life cycle?
- What is the software development life cycle?
- How can a tester be sure that bug was fixed?
- What do you prefer: white or black box testing?
- Describe a bug
- How do you see yourself in 3-5 years as a QA engineer?
- Why do you choose the QA career?
- When do you automate testing?
- What is the test documentation? Tell about this all that you know?
- How many bugs do you report in one bug report? Why?
- When does Testing begin?
- What is Re-testing?
- What are some of the differences between Agile and Waterfall?
- If you found a bug what do you have to do next?
- How do you perform GUI testing?

==

When the product is good enough to be released?

- When major bugs are fixed

How do you determine when you have done enough testing?

- Number of open bugs does not decrease over time (quality of product does not improve over time)

When do you automate testing?

1. when testing cannot be done manually:

- multi-user

- load

- stress

- performance

- volume

2. When the test is expected to be launched many times

- build acceptance (the most frequent) - done as many times as we have builds

- regression (second most frequent) - done against builds, which passed the build acceptance test

When does TESTING begin?

- Development stage - after the code is ready to be tested

When does QA begin?

- begins from day one with analysis and requirements

If you found a bug what do you have to do next?

- Report it immediately (unless instructed otherwise) upon making sure the bug is reproducible

If you find a bug and the developer says it is as-designed, what would you do?

- First make sure the developer is right - do not take her words as 100% truth.

- Then close the bug if she is right.

How do you test UI

- WEB application - Along the usability guidelines published on www.usability.gov

- Windows application - Along the GUI guidelines published by Microsoft

- iOS application - Along the guidelines published by Apple

- Android application - Along the guidelines published by Google

If you log a bug and the developer says she cannot reproduce it, what should you do?

-Make sure it was not fixed earlier (is still reproducible on your system)

-Compare the environments: OS, version of the product, browser, etc.

================================HOMEWORK================================

Testing Types/Levels to be discussed: - разберитесь с этими видами тестирования

- Black-white-gray box testing
- Functionality testing? (not Functional)
- Regression testing
- What is the performance testing? Load? Stress testing? Volume (non-functional)?
- Ad hoc, Exploratory
- Define User Acceptance Test? Beta-testing?
- Smoke testing? (superficial release/build acceptance; sanity check)
- What is Unit testing?
- Integration testing? System testing? End-to-End testing?
- What is API Testing?
- Test harness (automated test framework).
- Browser compatibility testing (cross-browser)?
- Operating system (cross-platform) Compatibility?
- Define Equivalence class? Equivalence partitioning? Boundary testing? Boundary Value Analysis?
- What is error handling testing (exception handling)?
- Localization and internationalization testing

===

Entry-Level - после курса, без опыта - начальные денежки (#20 в час)

Junior   - 1-2 года опыта   ($30 в час)

Mid-level   - 3 года   ($40-50 в час)

Senior   4-7 лет

Principal   - МНОГО