



考试成绩：60 试卷总分：72 通过分数：60 提交时间：2024-05-11 17:06:07
考生：代云舟 (0000099499) 部门：数智财税前端开发组 岗位：研发-程序开发-前端开发

一、单选题（本题共27小题，共27分）

1. 在 Jest 中，下面哪个函数用于测试一个异步函数的返回结果？（ 1分 ）

- ☐ A. testAsync()
- ☐ B. asyncTest()
- ☒ C. awaitTest()
- ☐ D. test()

答案 D

打分：☒ 0

2. 哪个模块测试的优先级最低（ 1分 ）

- ☒ A. service(API)
- ☐ B. React Component
- ☐ C. React hooks
- ☐ D. React Context

答案 A

打分：☒ 1

3. 在 @testing-library/react 中，下面哪个函数用于获取一个元素节点？（ 1分 ）

- ☐ A. getByTestId()
- ☐ B. getByRole()
- ☒ C. getLabelText()
- ☐ D. getByAltText()

答案 B

打分：☒ 0

4. 在React Testing Library中，可以使用什么方法来定位异步加载的元素？（ 1分 ）

- ☒ A. getBy
- ☐ B. queryBy
- ☐ C. findBy
- ☐ D. LocateAt

答案 C

打分 :  0

5. Jest配置文件中的testMatch选项是用来做什么的？（1分）

- ☒ A. 配置测试文件的匹配模式
- ☐ B. 配置测试覆盖率报告的输出目录
- ☐ C. 配置模块路径映射
- ☐ D. 配置测试环境

答案 A

打分 :  1

6. 下面哪个 Jest 的函数用于模拟一个函数的抛出异常？（1分）

- ☐ A. throwException()
- ☐ B. mockThrow()
- ☒ C. mockRejectedValue()
- ☐ D. mockError()

答案 C

打分 :  1

7. 在React Testing Library中,使用哪个匹配器可以断言一个元素是否存在于页面中？（1分）

- ☐ A. toBeVisible
- ☒ B. toBeInTheDocument
- ☐ C. ToExistElement
- ☐ D. toBePresent

答案 B

打分 :  1

8. Jest 中,下面哪个函数用于模拟一个模块的导出？（1分）

- ☐ A. mockImport()
- ☐ B. simulateExport()
- ☐ C. exportMock()
- ☒ D. jest.mock()

答案 D

打分 :  1

9. 在 @testing-library/react 中,下面哪个函数用于触发一个点击事件？（1分）

- ☒ A. fireEvent.click()
- ☐ B. fireEvent.change()
- ☐ C. fireEvent.submit()
- ☐ D. fireEvent.select()

答案 A

打分 : ☒ 1

10. 在进行React组件单元测试时，一般选择哪个测试辅助库（ 1分 ）

- ☐ A. Enzyme
- ☐ B. react-dom/utils
- ☒ C. testing-library
- ☐ D. react-addons-test-utils

答案 C

打分 : ☒ 1

11. 如何对React的Hooks进行测试（ 1分 ）

- ☐ A. React Hooks无法进行单元测试
- ☐ B. React Hooks无需进行单元测试
- ☐ C. 通过jest.mock 进行模拟测试
- ☒ D. 通过@testing-library/react-hooks提供的act和renderHook进行测试

答案 D

打分 : ☒ 1

12. 如果当您必须编写100行模拟数据，来测试单个函数时，首先应该想到什么？（ 1分 ）

- ☐ A. 不怕苦不怕累，都是为了提高代码质量
- ☐ B. 在整个项目长河中，是节省了时间的
- ☒ C. 考虑如何进行重构,让其变得更简单，更容易测试

答案 C

打分 : ☒ 1

13. 代码中通过import repository from "@chanjet/cc-front-repository";使用了库文件，我们执行单测时应该如何处理？（ 1分 ）

- ☐ A. 安装这个依赖包
- ☐ B. 修改源码暂时内部提供一个方法
- ☒ C. 使用vi.mock模拟模块的实现，例如：`vi.mock("@chanjet/cc-front-repository", async () => {
 return {
 default: {
 XXX
 }
 }
})`
- ☐ D. 1

答案 C

打分：☒ 1

14. 哪个模块不可以mock（1分）

- ☐ A. react-router
- ☐ B. localStorage
- ☐ C. axios
- ☒ D. 以上都可以mock

答案 D

打分：☒ 1

15. 在React Testing Library中，如何模拟点击事件？（1分）

- ☐ A. click()
- ☐ B. simulate('click')
- ☒ C. fireEvent.click()
- ☐ D. triggerEvent()

答案 C

打分：☒ 1

16. 为了避免多个mock函数互相影响，mockFn.mockClear()和mockFn.mockReset()之间的区别是什么？（1分）

- ☒ A. mockClear清空使用数据；mockReset重置实现。
- ☐ B. mockClear重置实现；mockReset清空使用数据。
- ☐ C. 它们的功能相同。
- ☐ D. mockReset用于异步函数，mockClear用于同步函数。

答案 A

打分：☒ 1

17. Jest匹配器用于做什么？（1分）

- ☐ A. 用于描述测试套件的结构
- ☒ B. 用于检查值是否满足一定的条件
- ☐ C. 用于模拟异步接口
- ☐ D. 用于生成随机数据

答案 B

打分 : ☒ 1

18. 在 Vitest 中，vi.mock 函数的作用是什么？（ 1分 ）

- ☐ A. 创建一个新的测试用例
- ☐ B. 创建一个新的测试套件
- ☐ C. 创建一个 模拟 函数
- ☒ D. 创建一个模拟 模块

答案 D

打分 : ☒ 1

19. 不是screen模块的方法是（ 1分 ）

- ☐ A. getByText
- ☐ B. queryByText
- ☒ C. selectByText
- ☐ D. findByText

答案 C

打分 : ☒ 1

20. 你想要在Jest中启用代码覆盖率报告，应该在配置中使用哪个选项？（ 1分 ）

- ☐ A. codeCoverageEnabled
- ☐ B. enableCoverage
- ☒ C. collectCoverage
- ☐ D. coverageReport

答案 C

打分 : ☒ 1

21. 在 Jest 中，如何断言一个函数被调用？（ 1分 ）

- ☒ A. expect(func).toBeCalled()
- ☐ B. expect(func()).toBeCalled()
- ☐ C. expect(func).toBeCalledWith()
- ☐ D. expect(func()).toBeCalledWith()

答案 A

打分： 1

22. Cypress 的主要特点是什么？（1分）

- ☐ A. 模拟用户与浏览器的交互行为
- ☒ B. 适用于端到端测试
- ☐ C. 可以在实际浏览器中运行测试
- ☐ D. 提供了丰富的断言库

答案 A

打分： 0

23. 如何对React的Hooks进行测试（1分）

- ☐ A. React Hooks无法进行单元测试
- ☐ B. React Hooks无需进行单元测试
- ☐ C. 通过jest.mock 进行模拟测试
- ☒ D. 通过@testing-library/react-hooks提供的act和renderHook进行测试

答案 D

打分： 1

24. 在 Jest 中，如果你希望使用真实的全局日期、性能和定时器 API，你应该调用哪个函数？（1分）

- ☐ A. jest.useFakeTimers()
- ☐ B. jest.advanceTimersByTime(msToRun)
- ☐ C. jest.setSystemTime(now)
- ☒ D. jest.useRealTimers()

答案 D

打分： 1

25. 为了异步等待元素出现在测试中，你应该使用哪个函数？（1分）

- ☐ A. waitForElement()
- ☐ B. wait()
- ☐ C. asyncWait()
- ☒ D. waitFor()

答案 D

打分： 1

26. 在Jest单元测试中，以下测试代码的输出顺序是？

```
describe('执行顺序', () => {  
  console.log(1);  
  describe('describe inner', () => {  
    test('test inner', () => console.log(2));  
    console.log(3);  
  });  
  test('test outer', () => console.log(4));  
}) ( 1分 )
```

- ☐ A. 1 2 3 4
- ☒ B. 1 3 2 4
- ☐ C. 2 4 1 3
- ☐ D. 4312

答案 B

打分：☒ 1

27. 下面哪个 Jest 的函数用于模拟一个函数的返回值？（ 1分 ）

- ☒ A. mockReturnValue()
- ☐ B. simulateReturn()
- ☐ C. returnMockValue()
- ☐ D. mockValue()

答案 A

打分：☒ 1

二、多选题（本题共29小题，共29分）

1. Vitest中通过哪个方法可以选择性的执行部分测试套件或者测试用例？（ 共1分，漏选不得分 ）

- ☒ A. .skip
- ☒ B. .only
- ☐ C. .include
- ☐ D. .exclude
- ☐ E. .runIf

答案 A、B、E

打分：☒ 0

2. beforeEach函数可以用来做什么（ 共1分，漏选不得分 ）

- ☒ A. 设置测试环境
- ☒ B. 初始化测试数据
- ☒ C. 清理测试数据

- ☐ D. 执行异步操作

答案 A、B、C

打分：☒ 1

3. Jest测试框架提供了哪些全局函数 （共1分，漏选不得分）

- ☒ A. beforeAll
- ☒ B. beforeEach
- ☒ C. afterEach
- ☒ D. afterAll

答案 A、B、C、D

打分：☒ 1

4. 在 Vitest 中，如何定义一个测试用例？ （共1分，漏选不得分）

- ☒ A. 使用 it 函数
- ☐ B. 使用 describe 函数
- ☒ C. 使用 test 函数
- ☐ D. 使用 case 函数

答案 A、C

打分：☒ 1

5. 单元测试的意义？ （共1分，漏选不得分）

- ☒ A. 验证代码逻辑
- ☒ B. 减少代码缺陷
- ☒ C. 促进代码设计
- ☒ D. 增强代码信心
- ☒ E. 便于缺陷定位

答案 A、B、C、D、E

打分：☒ 1

6. jest可以测试哪些文件 （共1分，漏选不得分）

- ☒ A. .js
- ☒ B. .ts
- ☒ C. .jsx
- ☒ D. .tsx

答案 A、B、C、D

打分：☒ 1

7. 为什么要进行单元测试？（共1分，漏选不得分）

- ☒ A. 确保代码质量：单元测试可以验证代码的正确性和功能是否按照预期工作。通过编写针对每个单元（函数、方法等）的测试用例，并检查其预期输出是否与实际输出一致，可以提高代码的质量并减少潜在的缺陷。
- ☒ B. 提供快速反馈：单元测试可自动化执行，并且通常比其他类型的测试更快。这意味着您可以在每次修改代码后立即运行单元测试，以获取快速反馈。这样可以快速发现问题并及时修复，而不会导致问题逐渐累积和扩散到整个系统。
- ☒ C. 支持重构和维护：单元测试在代码重构和维护过程中起到了关键作用。通过有一套全面的单元测试覆盖率，您可以放心地对代码进行重构，保证所做的更改没有引入新的缺陷。此外，当需要修复或优化某个功能时，单元测试可以充当一个安全网，确保修改不会破坏现有的功能。
- ☒ D. 促进团队协作：单元测试具有规范和明确的接口，可以促进开发团队之间的协作。开发人员可以根据单元测试的要求编写代码，并与其他团队成员共享测试用例和预期结果。这样可以提高沟通效率，减少因误解而导致的问题。
- ☒ E. 支持持续集成和部署：在持续集成和持续部署（CI/CD）流程中，单元测试是至关重要的组成部分。通过自动运行单元测试，可以确保每次代码提交都符合预期的质量标准。这有助于减少错误在生产环境中被部署的风险，提供更可靠的软件交付流程。

答案 A、B、C、D、E

打分：☒ 1

8. 下列哪些工具可以用于前端单元测试？（共1分，漏选不得分）

- ☒ A. Jest
- ☐ B. Cypress
- ☒ C. Mocha
- ☐ D. Jasmine

答案 A、C、D

打分：☐ 0

9. jest覆盖率统计了哪些信息（共1分，漏选不得分）

- ☒ A. statements
- ☒ B. branches
- ☒ C. functions
- ☒ D. lines

答案 A、B、C、D

打分：☒ 1

10. 基于 Testing Library的前端单元测试中，以下哪些API支持获取DOM（）？（共1分，漏选不得分）

- ☒ A. getByRole
- ☒ B. getByTestId
- ☒ C. getByLabelText
- ☐ D. getElementById

答案 A、B、C

打分：☒ 1

11. vitest中下面有关模拟定时器说法正确的是？（共1分，漏选不得分）

- ☒ A. 模拟定时器,可以使用vi.useFakeTimers方法
- ☒ B. vi.runAllTimers方法将调用每个已经启动的定时器，直到定时器队列为空，这意味着在 runAllTimers 期间调用的每个定时器都会被触发
- ☒ C. vi.runAllTimersAsync 方法将异步调用每个已启动的定时器，直到定时器队列为空。这意味着在 runAllTimersAsync 期间调用的每个定时器都会被触发，即使是异步定时器
- ☒ D. 在调用vi. useFakeTimers()之后，可以通过调用vi.setSystemTime()方法模拟用户更改系统时钟

答案 A、B、C、D

打分：☒ 1

12. 以下哪些是我们约定的单测原则？（共1分，漏选不得分）

- ☒ A. 不测试别人的代码
- ☒ B. 代码要尽量减少依赖，尽量通过参数传递上下文
- ☒ C. 必须是自动化的
- ☐ D. 一个测试用例尽量覆盖多个场景

答案 A、B、C

打分：☒ 1

13. Jest中经常用mock函数来测试一个组件的回调，"mockFn.mock.calls"属性包含什么信息？（共1分，漏选不得分）

- ☒ A. 所有调用的参数
- ☐ B. 所有调用的结果
- ☒ C. 调用次数
- ☐ D. 调用的上下文

答案 A、C

打分：☒ 1

14. 下面哪些是React Testing Library的特点？（共1分，漏选不得分）

- ☒ A. 测试组件页面DOM渲染
- ☒ B. 鼓励更符合实际用户行为的测试
- ☒ C. 测试组件的实例（instance）
- ☐ D. 测试组件的样式布局

答案 A、B

打分：☒ 0

15. 以下哪些方式可以实现前端的持续集成（共1分，漏选不得分）

- ☒ A. 自动化测试
- ☒ B. 自动化语法检查
- ☒ C. 对NPM模块进行自动化依赖检测
- ☒ D. 自动化资源构建

答案 A、B、C、D

打分 : ☒ 1

16. Cypress 是一款优先的端到端测试框架，以下哪些能力是Cypress不支持的？（共1分，漏选不得分）

- ☒ A. Cypress 是自集成的，提供了一套完整的端到端测试，无须借助其他外部工具，安装后即可快速地创建、编写、运行测试用例，且对每一步操作都支持回看
- ☒ B. 不同于其他只能测试 UI 层的前端测试工具，Cypress 允许编写所有类型的测试，覆盖了测试金字塔模型的所有测试类型【界面测试，集成测试，单元测试】
- ☒ C. Cypress 底层协议与 WebDriver一样，也是通过在外部浏览器运行并在网络上执行远程命令来运行
- ☒ D. Cypress 底层协议不采用 WebDriver

答案 A、B、D

打分 : ☐ 0

17. 在 Jest 中，可以使用哪些方法来模拟函数的行为？（共1分，漏选不得分）

- ☒ A. mockReturnValue()
- ☒ B. mockImplementation()
- ☒ C. mockResolvedValue()
- ☒ D. mockRejectedValue()

答案 A、B、C、D

打分 : ☒ 1

18. Cypress 视觉回归测试的主要优势是什么？（共1分，漏选不得分）

- ☒ A. 可以精确到像素级别的比较
- ☒ B. 支持多种浏览器和操作系统
- ☐ C. 测试组件内部变量、方法的实现细节
- ☒ D. 测试组件的样式布局

答案 A、B、D

打分 : ☒ 1

19. React Testing Library的render方法会返回以下哪些那些对象和方法？（共1分，漏选不得分）

- ☒ A. container
- ☒ B. baseElement
- ☒ C. debug
- ☐ D. root

答案 A、B、C

打分： 1

20. vitest中有关vi.mock 说法正确的是？（共1分，漏选不得分）

- ☐ A. vi.mock 仅对使用 import 关键字导入的模块有效。它对 require 无效
- ☒ B. vi.mock('xxx')将自动模拟xxx模块的所有导出成员，如果没有指定实现默认的模拟函数返回值为undefined
- ☐ C. 不要忘记将 vi.mock 调用提升到文件顶部。它将始终在所有导入之前执行
- ☒ D. vi.mock 被提升（换句话说，移动）到文件的顶部。这意味着无论何时写入它（无论是在 beforeEach 还是 test），它都会在此之前被调用
- ☒ E. 如果不调用 vi.mock，模块不会被自动模拟

答案 A、B、C、D、E

打分： 0

21. 在 Vitest 中，可以如何进行断言？（共1分，漏选不得分）

- ☒ A. 使用 expect 函数
- ☒ B. 使用 assert 函数
- ☐ C. 使用 should 函数
- ☐ D. 使用 verify 函数

答案 A、B

打分： 1

22. 以下哪些是 Vitest 的常用断言方法？（共1分，漏选不得分）

- ☒ A. toBe()
- ☒ B. toEqual()
- ☒ C. toHaveLength()
- ☒ D. toMatch()
- ☐ E. toBeInTheDocument()

答案 A、B、C、D

打分： 1

23. 代码覆盖率常用以下几个度量指标来表示？（共1分，漏选不得分）

- ☒ A. 语句覆盖率（Statement Coverage）：表示被执行的代码语句所占总代码语句数量的比例。它衡量了测试用例是否涵盖了程序中的每个语句。
- ☒ B. 分支覆盖率（Branch Coverage）：表示被执行的分支（例如 if、switch 等条件语句）所占总分支数量的比例。它衡量了测试用例是否涵盖了程序中的每个分支路径，包括真值和假值两种情况。
- ☒ C. 函数覆盖率（Function Coverage）：表示被调用的函数或方法所占总函数数量的比例。它衡量了测试用例是否涵盖了程序中的每个函数调用。
- ☒ D. 行覆盖率（Line Coverage）：表示被执行的代码行所占总代码行数的比例。它衡量了测试用例是否涵盖了程序中的每一行代码。

答案 A、B、C、D

得分：☒ 1

24. 前端开发中，如果哪些方式可以对代码进行静态扫描 （共1分，漏选不得分）

- ☒ A. Sonar
- ☒ B. Jest
- ☒ C. Eslint
- ☐ D. Flow

答案 A、C、D

得分：☒ 0

25. 下列哪些特点是属于Jest的？ （共1分，漏选不得分）

- ☒ A. 异步测试支持
- ☒ B. 模拟系统函数
- ☒ C. 快照测试支持
- ☒ D. 真实浏览器环境

答案 A、B、C

得分：☒ 0

26. vitest中有关模拟函数说法正确的是？ （共1分，漏选不得分）

- ☒ A. 我们可以使用 vi.fn 方法创建一个 mock 函数来跟踪其执行情况
- ☒ B. 如果要跟踪已创建对象上的方法，可以使用 vi.spyOn 方法
- ☒ C. mockImplementationOnce接受一个函数，该函数将在下一次调用时用作 mock 的实现。可以链式调用多个函数，从而产生不同的结果
- ☒ D. mock.lastCall其中包含最后一次调用的参数。如果 mock 未被调用，将返回 undefined。

答案 A、B、C、D

得分：☒ 1

27. 在Cypress视觉回归测试中，测试用例通常包括哪些步骤？ （共1分，漏选不得分）

- ☒ A. 设置测试数据
- ☒ B. 执行操作
- ☒ C. 验证页面元素
- ☒ D. 对比界面截图

答案 A、B、C、D

得分：☒ 1

28. 以下哪些是单元测试的优势？ （共1分，漏选不得分）

- ☒ A. 安全重构
- ☒ B. 保证代码质量
- ☒ C. 尽早发现软件Bug
- ☒ D. 促进变化并简化集成
- ☒ E. 提供文档

答案 A、B、C、D、E

打分：☒

29. 下面关于测试金字塔说法正确的是？（共1分，漏选不得分）

- ☒ A. 测试金字塔（Test Pyramid）是一种软件测试策略的模型，用于指导测试团队在不同层次上选择和执行各种测试类型。测试金字塔的形状类似于一个金字塔，由三个层次组成，从底部到顶部分别是单元测试、集成测试和端到端测试
- ☒ B. 单元测试（Unit Testing）：单元测试是针对程序的最小可测试单元（例如函数或方法）进行的测试。它们通常由开发人员编写，并旨在验证代码的单个部分是否按预期工作。单元测试应该快速、自动化且易于维护
- ☒ C. 集成测试（Integration Testing）：集成测试涉及多个组件或模块之间的交互，目的是确保这些组件在整体上能够正确地协同工作。集成测试可以验证接口、数据流和系统中的重要交互。它们可以由开发团队或专门的集成测试团队执行。
- ☒ D. 端到端测试（End-to-End Testing）：端到端测试也称为系统测试或功能测试，是对整个应用程序或系统进行的高级测试。它模拟真实的用户场景，通过与应用程序的外部接口进行交互来验证整个系统的功能和性能。端到端测试通常由专门的测试团队执行。

答案 A、B、C、D

打分：☒

三、判断题（本题共16小题，共16分）

1. @testing-library/react库的waitForElement函数用于等待指定元素出现在DOM中。（1分）

- ☐ 正确
- ☒ 错误

答案 错误

打分：☒

2. React Testing Library 提供了 act，render，screen，fireEvent 等常用方法和模块（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒

3. screen模块可以使用css选择器（1分）

- ☐ 正确
- ☒ 错误

答案 错误

打分：☒ 1

4. Create React App使用Jest和React Testing Library作为默认测试工具（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒ 1

5. 所有的react hook都不需要在组件的环境中进行测试（1分）

- ☐ 正确
- ☒ 错误

答案 错误

打分：☒ 1

6. 可以通过rerender方法改变组件的props（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒ 1

7. @testing-library/react库的render函数返回一个包含渲染组件的快照。（1分）

- ☐ 正确
- ☒ 错误

答案 错误

打分：☒ 1

8. @testing-library/react库的waitFor函数用于等待异步操作完成。（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒ 1

9. 测试React组件时，API请求模块需要进行mock（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒

10. 组件使用了mobx-redux，无法使用Jest进行测试（1分）

- ☐ 正确
- ☒ 错误

答案 错误

打分：☒

11. Jest可以mock计时器，直接跳过等待执行回调（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒

以下代码不可以进行单元测试

12.

```
function sum(a, b) {  
  console.log(a+b)  
}
```

（1分）

- ☐ 正确
- ☒ 错误

答案 错误

打分：☒

13. React Testing Library 提供了测试react组件的运行环境（1分）

- ☒ 正确
- ☐ 错误

答案 正确

打分：☒

14. Jest.mock可以用来mock一个函数（1分）

☒ 正确

☐ 错误

答案 错误

打分：☒ 0

15. 可以使用fireEvent模块触发事件，从而执行事件的监听回调（ 1分 ）

☒ 正确

☐ 错误

答案 正确

打分：☒ 1

16. React Testing Library简化了React组件的单元测试（ 1分 ）

☒ 正确

☐ 错误

答案 正确

打分：☒ 1