

1. The following requirements: Good or bad requirements? (and why?)
 - a. The user can choose a document type from the drop-down list.
 - b. As a user, I want to sign up via email and password so that I can access my account.
 - c. The list of the users should be stored as a linked list.
2. An example of refactoring is adding new features to satisfy a customer requirement discovered after a project is shipped.
3. Which kind of process/methodology would you use for a large system with lots of vague requirements?
4. What is the difference between requirements definition and specification? Explain + examples?
5. What is the difference between fault and failure? Give an example?
6. Why is it useful to know design patterns?
7. What is the main difference between unit testing and integration testing?
8. Explain with example "100% coverage is not always a reasonable target".
9. What is Regression Testing?
10. How many unit tests do you need to cover every branch? How many unit tests do you need to cover every statement in the following code? Provide the unit tests (inputs), explain your answer!

```
int max(int a, int b){  
    int m=a;  
    if(a>=b)  
        m=a;  
    return m;  
}
```

11. Are black-box testing and white-box testing the same? Explain your answer + example.

12. List the problem with the following code? How to fix (Refactoring) it?

function that returns avg price for two lists of items and

```
def avgFunc(price_items1,price_items2):
```

```
    targetList = list(price_items1.values())
```

```
    sumOfRates = 0
```

```
    for rate in targetList:
```

```
        sumOfRates += rate
```

```
    avgRate1 = sumOfRates / len(targetList)
```

computer avg price for list of salary

```
    targetList = list(price_items2.values())
```

```
    sumOfRates = 0
```

```
    for rate in targetList:
```

```
        sumOfRates2 += rate
```

```
    avgRate2 = sumOfRates / len(targetList)
```

```
    return avgRate1,avgRate2
```