

①

AS.

Aktifya Sehgaz

(1911109)

B3 batch.

STQA E&E

Q1 1.1) c

1.2) b

1.3) b

1.4) d

1.5) d

1.6) b

1.7) a

1.8) a

1.9) c

1.10) c

Q 1) (B)

6) - Regression Testing is a type of testing that is performed throughout the life cycle of the system.

- It is done to make sure modifications do not introduce new faults in the part that was not actually modified.
- It is a subphase of unit, integration and system level testing.

Selective retest is an approach of performing regression testing using minimal and specific test cases. The main categories are:

- Coverage Technique - This ensures coverage of modified components of a software program which may be covered under the test and making the selection of test cases.
- Minimization Technique - similar to coverage technique but it is carried out using a minimal set of test cases.
- Safe Techniques - It's a technique of selecting every possible test case that may cause a modified software to generate a different output than expected.

PS.

Aditya Sehgal

(171109)

(3)

B3 batch.

Q1 B) 5) Testing

- Def: Testing is the process of finding bugs and errors.

- Testing is done by a tester mainly.

- Testing is displaying errors.

- complete knowledge of design not needed.

- can be done by internal or external party.

- composed of verification and validation

debugging.

- Def: Debugging is the process of correcting the bugs that have been found in testing.

- Debugging can be done by programmer or developer.

- Debugging is fixing those errors.

- The design should be understood well by the developer / programmer who is debugging.

- cannot be done by external party, only internal.

- composed of fixing bugs by matching cause with effect.

PS.

Aditya setgal.

(1711109)

B3 batch.

Q(1) B) 2) Mutation testing P5 when an application is tested for code that was modified after fixing a bug/ defect. It helps to find out which code and which strategy is more effective.

A MUTANT is a small change or a modification.

If the mutant is killed or is dead we conclude the test case has failed.

MUTATION SCORE = Percentage of non equivalent mutants killed by the test suite.

$$= 100 * \frac{D}{(N-E)}$$

where D: Dead mutants ; N: Total mutants

E: No of equivalent mutants.

1) In Random Testing, the test inputs are selected randomly from the input domain of the system. The main four steps are:

- 1) Identify Input domain
- 2) Independently select test cases from domain
- 3) System is executed on these inputs.
- 4) Results are compared against specification.

Random testing allows us to easily estimate software reliability from test outcomes.

It rewards failure times and helps estimate reliability.

Q5.

Activity a

(5)

(17/11/09)

EE batch.

Q1 B) 7) Ishikawa diagram shows the causes of an event and is used in manufacturing and product development, to outline steps of a process.

(6)

Ans.

Amitya Sehgal.

(17/11/09) B3 batch.

Q(3) a) off-the-shelf (OTS) component testing is the testing done on OTS component before purchasing. The 2 main types of tests are:

(i) Acceptance testing of OTS component based on specific guidelines.

(ii) Integration testing is integration of the component with other components that have been developed in house or from third party.

If the documentation is not good or if the vendor is not offering support, it ~~can cause~~ a problem in integration.

First the ~~if~~ component is very thoroughly tested for acceptance testing against the specified requirements.

This helps making integration testing easier and now we just need to test how integrated components perform.

Further, 3 types of testing techniques are used to determine the suitability of the component.

→ Black Box component testing: It is used to determine the quality of the component.

Aditya Sehgal

(1711109)

(7)

B3 batch.

Q③ a) wnt...

→ system level fault injection testing: used to determine how well a system will tolerate a failure component. It can help predict the behaviour of system if theOTS component fails.

→ operational system Testing: Used to determine the tolerance of a software system when the OTS component is functioning perfectly. It helps access perfect fit of component in the system.

Q③ b) The 5 level of maturity (CMM) are:

1) Initial Level (Level 1) - The process is usually chaotic and ad hoc.

- Capabilities are characterised on the basis of individuals here not the organisation

- Progress not measured.

- Usually over budget and off schedule

Key process area: None. There is total chaos.

2) Managed level (Level 2) - The requirements are managed. Progress is measured.

- Parameters like cost, schedule and

(8)

AS.

Aditya sehgal.

(1711109)

B3 batch

Q ③ b) cont...

function are estimated.

- Project standards are defined.
- Plans and processes are developed.
- Processes may differ between projects.

key process area: software project planning
 planning and oversight
 software quality management
 requirement management.

3) Defined level (level 3) - here we make sure
 the product is meeting the requirements and
 intended use. Standards for organisation followed.

- clarify customer requirements.
- Analyse decisions and rectify and control
 problems and mitigate risks.
- Define an implementation process.

key process area: Training programs.

software Product Engineering
 peer review

organisation level process definition.

4) Quantitatively managed (level 4) - manages projects' processes and sub processes statistically.

- Understand project process performance and quantitatively manage organisational project.

key process area: software quality management
 quantitative process management.

Q3 b) cont...

- 5) optimised level (level 5) - Here we detect and remove causes of defects early.
• Identify and deploy new tools and process improvements to meet needs and business objective.

Key process area: process change management
Defect prevention.

Q4 McCall quality factors and criteria.

→ A quality factor helps represent the system's behavioural characteristics like efficiency, testability, portability etc.
There are the following 11 factors:

- 1) correctness - the extent to which a program is correct or satisfies specifications.
- 2) reliability - the extent to which a program can be expected to perform function properly.
- 3) efficiency - the amount of computing resources and code required by a program to function.
- 4) Integrity - the extent to which access to software / data by unauthorised person can be controlled.
- 5) Usability - effort required to learn to operate it.

Q ④ cont...

- 6) Maintainability - Effort required to locate and fix an error in program.
- 7) Flexibility - Effort required to modify a program that is in operation.
- 8) Testability - Effort required to test a program is performing it's function or not.
- 9) Portability - Effort required to transfer software from one hardware / software environment to another.
- 10) Reusability - How easily can it be used (reused) with other programs.
- 11) Interoperability - Effort required to use one system with another.

quality criteria - It is an attribute of a quality factor that is related to software development.

like: modularity is an attribute of architecture.

Some quality criteria are:

- Access Audit
- Consistency
- Access control
- Error tolerance
- Accuracy
- Generativity
- Completeness
- Simplicity
- Concise ness
- Training

(11)

AS

Aditya Sehgal · (1711109)

B3 batch

Q2

```
void sort (int a[], int n) {
```

```
    int i, j; - ①
```

```
    for (i = 0; i <= n-1; i++) - ②
```

```
        for (j = i+1; j < n; j++) - ③
```

```
            if (a[i] > a[j]) { - ④
```

```
                temp = a[i];
```

```
                a[i] = a[j];
```

```
                a[j] = temp
```

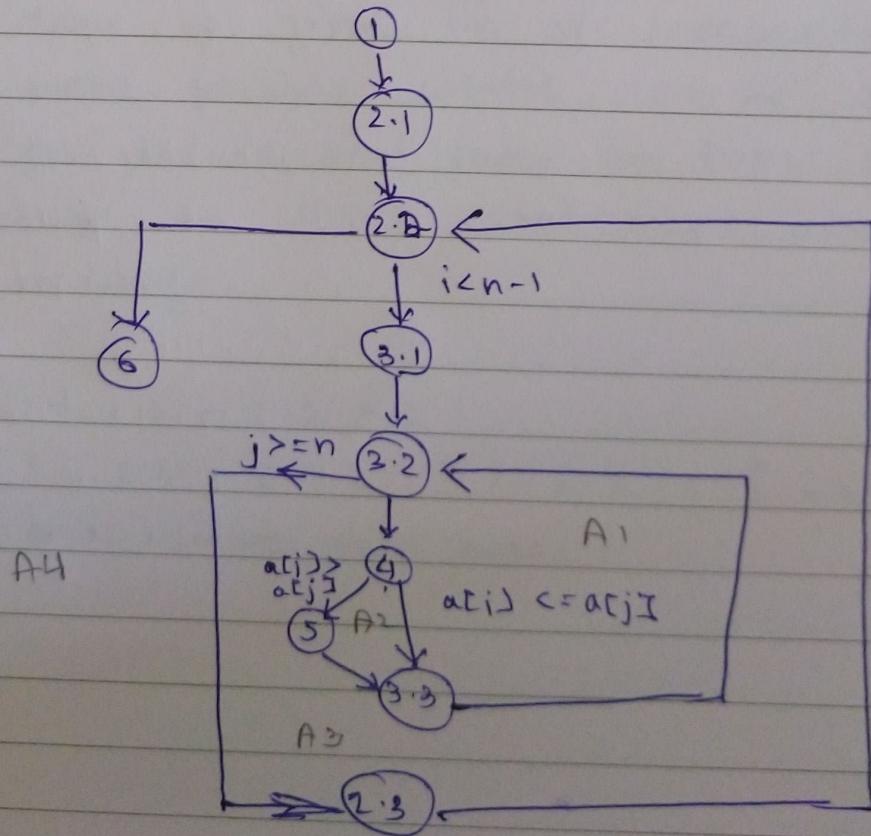
y

y

- ⑤

- ⑥

a) control flow graph :



Q5.

Aditya Sengar (17M1109)

(12)

B3 batch

Q2 cont...

b) cyclomatic complexity

$$V(G) \doteq E - N + 2$$

↑ ↑
Edges Nodes

- calculated from
control flow
graph.

$$= 12 - 10 + 2$$

$$= 4$$

cyclomatic complexity is 4.

c) Cyclomatic complexity is a metric that provides a quantitative measure of the logical complexity of the program.

Helps us find no of independent paths possible which can be used to develop test cases on those paths.

Helps do 100% statement and branch coverage.

d) P1: 1 - 2.1 - 2.2 - 6

P2: 1 - 2.1 - (2.2 - 3.1 - 3.2 - 2.2) - 2.2 - 6

and 2 more test cases.