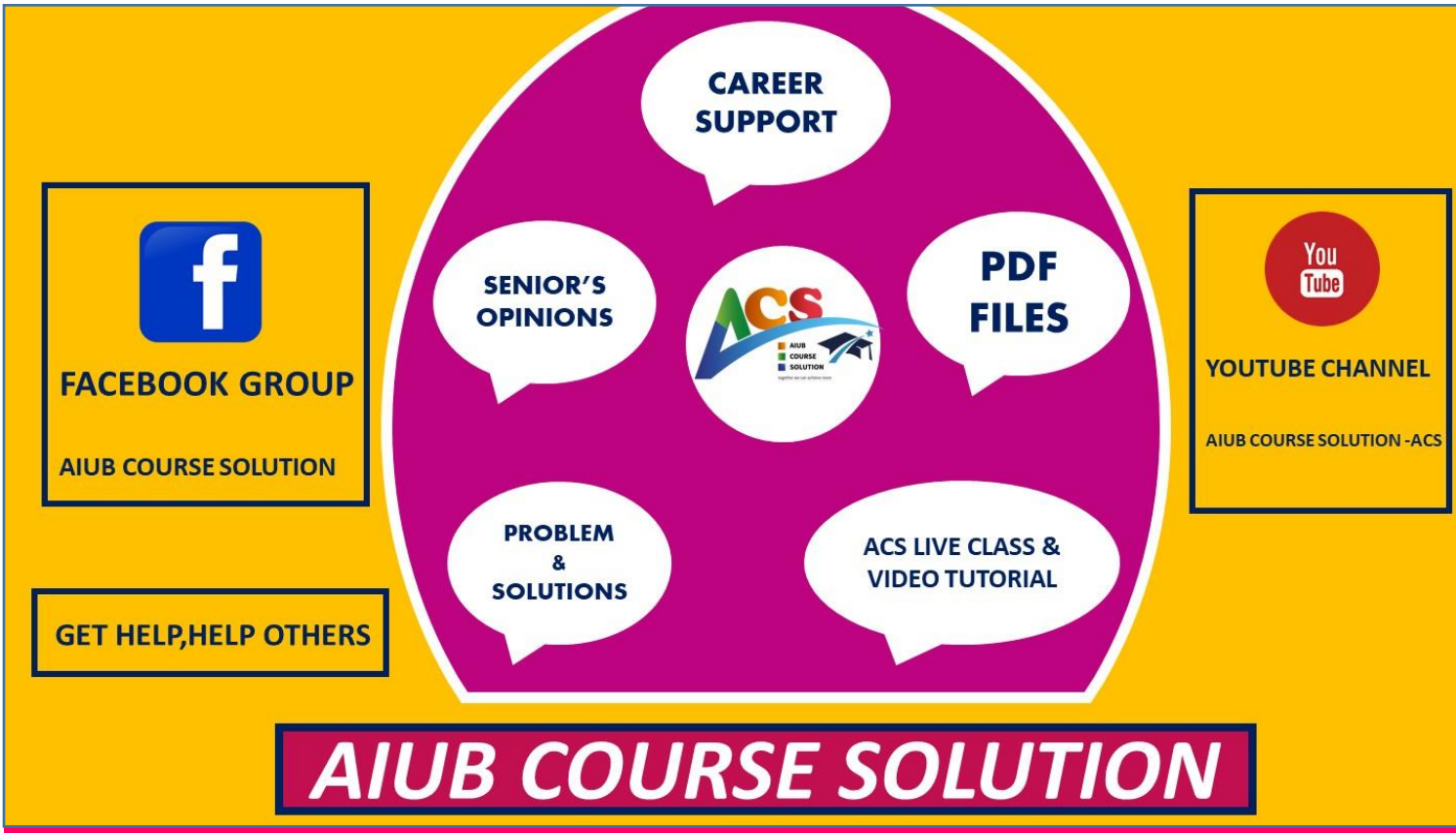
AIUB COURSE SOLUTION



COURSE NAME:

CHAPTER:

SOLVED BY

NAME: ZULKAR NAYEEM

AIUB COURSE SOLUTION-ACS

LINK = https://www.youtube.com/channel/UCC3KjA8kstFtM-2CxVr-jcg

AIUB COURSE SOLUTION

LINK= https://www.facebook.com/groups/aiubcoursesolution/

ACS LIVE CLASS & VIDEO TUTORIAL

1.Messages:

Synchronous:The sender suspends execution until the message is complete.

Example:lift er button,portal e login,atm e login

Asynchronous:The sender continues execution after sending the message.

Example:messenger texts,email sending.

2.What is message in diagram or code?

Message is an element that defines one specific kind of communication between lifelines

of an interaction between two or more objects.

3.Statechart diagram :state,transition and event.

Example:State:Student Object sitting for an exam and giving result.

4.Activity diagram:An activity diagram shows actions and the flow of control

and data between them.

5.Activity and Action diff:Activity is a pack of many actions together.

If Activity is expanded, we get action.Action is atomic task which can't be expanded.

6.Programming:C++,parameter nai

7.Multiple Inheritence:1 ta child ,many parents.

8.Component:Component is a modular unit with well-defined interfaces which is replaceable within it's environment and it is encapsulated.

Inside component there are interfaces,usage dependency,course and connectors.Components are: Faculty component,Student component,Office component,Library component,Registration component.Project component; lab,reception

9.Provide and required diff:A required interface on a port specifies one or more operations

required by behaviors of the methods.

A provided interface on a port specifies one or more operations that a part must provide.

These two often work together to build a system.

13.Provided and require interface example:Required pre-requisite and provide request course.

10.Port:Port is an interaction point between component and environment.

Example:lab test kit,office component and administration environment.

11.Connectors:Connects 2 componenets. external contract is connected to internal structure.

2 kinds of connectors:delegation(external contract and internal realization)example:bank account and cash withdrawal(debit),deposit(credit)

assembly(required and provided e use hoi)example:office and manager.

12.Interface:Interface is a set of class,method and rules to complete task between two components.

14.Generalization is a sub class.such as person and student,staff and teachers.

Realization relation indicates that one class implements a behavior specified by other another class.

15.Process: A process is any software related activity such as change analysis,specification,design

coding and testing.

16.Internal and external attribute diff: Internal attribute deals with the code's complexity,reusablility

External attribute works for the maintainability of program source code or productivity of software personnel.

17.McCabe's cyclometric complexity: edge-node+2

23.Cyclomatic complexity: In Directed graph we need relation between node and flow of control which is represented by edges .we need cyclometric complexity to calculate linearly independent path .

Highly complex program use this to change method.

18.Object oriented software matric:

1.Weighted methods per class (WMC)

2.depth of inheritance tree(DIT)

3.number of children(NOC)

4.coupling between classes (CBC)

5.lack of cohesion in methods(LCOM)

19.Coupling:2 or more class depend on each other that's called coupling.

20.Lcom calculate:P-Q p=non cohesive,q=cohesive

21.Class and Object: Class is an code template where we write methods,attributes and rules for creating and objects.Object is an instance of a class.Example:Sir and student.

24.Forking:If in activity diagram2 or more work are done together simultaneously or parallelly that's when we use forking.

25. (Activation bar): In Sequence diagram Activation bar is the box placed on the lifeline.it is used to indicate that an object is active during and interaction between to objects.