

4.3

by Student User

Submission date: 20-May-2020 10:05PM (UTC+0100)

Submission ID: 128499956

File name: 4.3_3306_1790660983.odt (1.13M)

Word count: 2709

Character count: 13651

UNIT 4.3
COMPUTING PROGRAMMING
1 TUTOR NAME: MS. IBTISAM MOGUL

SUBMITTED BY FATIMA EJAZ BARRI

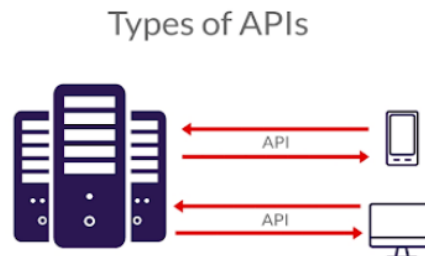
1. A) APPLICATION PROGRAMMING INTERFACE DEFINITION

Application Programming Interface (APIs) are tools and programs that help in building software applications. APIs make the app work smoothly and it connects all the different features of the application so consumers can enjoy the app without complications.

Types of APIs

APIs are used for proper functioning of:

- Operating systems
- Applications
- Websites



For example, functions of APIs used by famous apps are:

1. Google APIs: APIs used by Google allows the smooth working of the Google website while also integrating it with its other apps or functions. There are a lot of APIs for different uses such as Abusive Experience Report API, Gmail API, etc.
2. YouTube APIs: YouTube APIs are designed to make YouTube easy to use and they are of different types with specific functions to fit YouTube's characteristics like accessing your data, liking, commenting, etc. One example is YouTube Data API.
3. Amazon brand: As Amazon has different sellers and brands which use the website to sell different things, APIs used by Amazon allow sellers to make their experience easier and there are a lot of APIs for every type of seller. For example, Amazon SE3 API, etc.

ADVANTAGES OF APPLICATION PROMGRAMMING INTERFACE

- Automation: computers can do the work of APIs rather than humans. The agencies can upgrade workflows though APIs to make them quicker and more efficient.
- Personalization: any individual or organization can customize the content and services they use the most by means of APIs.

Type of API: SOAP

SOAP is an API used by web services and it's full form is Simple Object Access Protocol. It is an API made to help in messaging and it can work with other apps easily like HTTP. It is usually used for complicated web services functions because it provides security.

Advantages of SOAP:

- It provides security to its consumer.
- It is a high quality API.
- There can be additions to it according to the consumer's needs.

Disadvantages of SOAP:

- It does not have a good performance.
- It is complicated to use.
- It doesn't have many uses.



1. ¹ the Software development lifecycle (SDLC)

The software development life cycle SDLC is a process used to develop software systems. The SDLC's main goal is to develop software that meets and exceeds customer expectations.

The SDLC consists of the following steps to meet their main goal, develop a good software system.

Phase 1: planning

The planning phase consists of determining the main goal of the product and the software system. This phase also consists of planning of how to reach said goals. The two primary activities involved in the planning phase are as follows:

1. Identification of the system for development
2. Feasibility assessment

Phase 2: analysis

In the process of analysis, the business requirements needed for the completion of the project are analysed and the planning of the project is modified. The two primary activities involved in the analysis phase are as follows:

1. Gathering business requirement
2. Performing a detailed analysis

Phase 3: design

In the design phase, we plan and confirm the desired features and look of the project. This phase includes business rules, pseudo code, screen layouts and other necessary documentation. The two primary activities involved in the design phase are as follows:

1. Design of IT infrastructure
2. Designing the system model

To avoid any fail in the system or the final project, the technical base of the software should be of high quality.

³

Phase 4: development

In the development phase, all the information from the phases is kept in mind and the developing of the software system is done according to the plan. The two primary activities involved in the development are as follows:

1. Development of IT infrastructure
2. Development of database and code

In the design phase, only a rough estimation of the final project is made while in the development phase the rough plans are put to work and tweaked to perfection.

3

Phase 5: testing

In the testing phase, all the parts to develop the final system are integrated. Testers then follow the software-testing life cycle to system for any malfunctions so that they can avoid them in the final software. The two primary activities involved in the testing phase are as follows:

1. Writing test cases
2. Execution of test cases

Testing is a very important part of the whole cycle because it helps to build a high quality software. Once test cases are done, the tester compares the expected result to the actual result to see the errors in the system and devises plans to fix them. Writing test cases and executing them is a difficult task for organizations but it promises the execution of a great project if done properly.

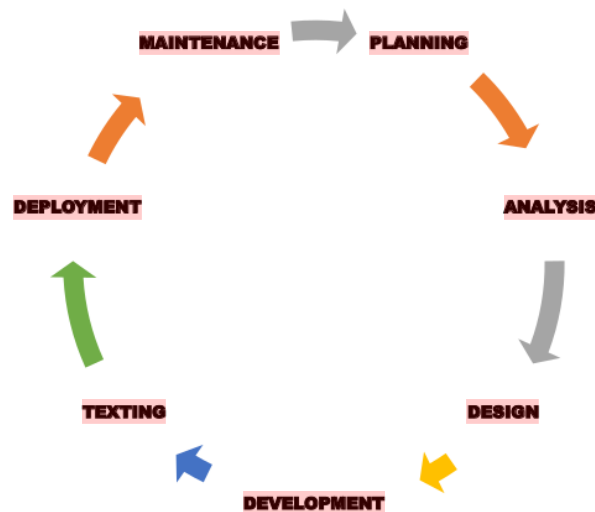
Phase 6: deployment

In this phase, the software is put out in the world for the clients use. All the components of the system are put out for public use. This phase is also called the delivery phase.

Phase 7: maintenance

In the maintenance phase, important changes are made to fit the system in the client's expectations, the organization works to improve the system and updates it. This is necessary to enable the software to stay relevant and helpful in the future. The three primary activities involved in the maintenance phase are as follows:

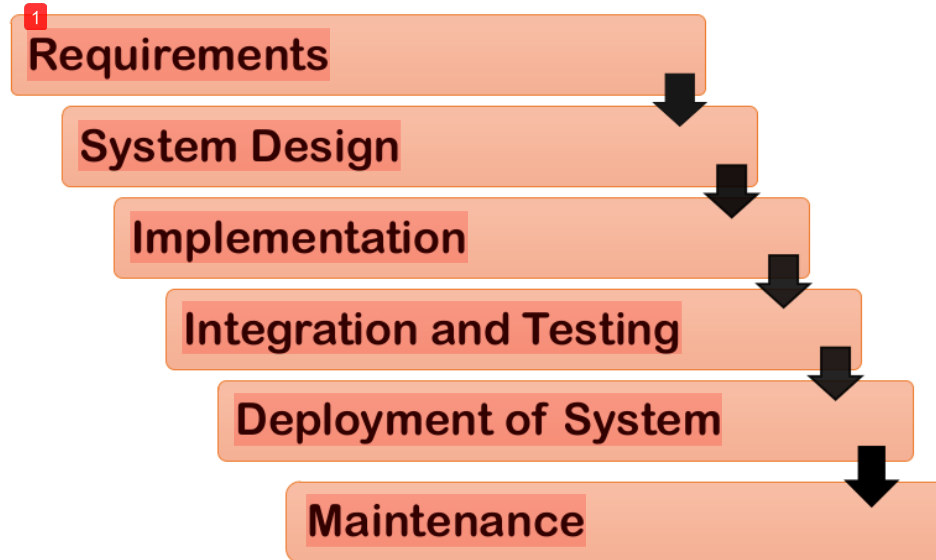
1. Support the system users
2. System maintenance
3. System changes and adjustment



Waterfall Model With Diagram And All The Phase

The waterfall model is a model with linear and sequential approach used in system development life. The team waterfall is used for this classical model because it develops systematically from one phase to another in a downward fashion.

Diagram of the waterfall model



Phases of the waterfall model

Requirements: in the phase of requirement, the specification of the inputs and outputs and the final product is marked.

System design: in this phase, a design is made for the system. In addition, the code of the system is created in this stage.

Implementation: from the system design, prepared individual units are integrated in the next system and the whole system is made.

Integration and testing: in the phases of integration & testing, the company puts in the use after system has been successfully tested. Testing is done so that the client does not have any problem during the installation of the software. If there are any flaws or errors. Every unit is tested for its functionality. If it's there are many errors then they will surface at this point in the process.

Deployment of the system: after the setting up of the system and testing, the system is released in the market and is available for customers use.

Maintenance: this step is done after the release of the system, this is done to solve any problems faced by the clients after live use of the system and to update it according to modern upgrades so that the system can be used in the future.

Advantages of the waterfall model

- It is a basic and simple model.
- It is easy to apply due to its inflexible structure.

- The steps are independent and are finished before proceeding to the next step.
- Tests are done to ensure correction in every phase.
- It helps to find errors earlier

Disadvantages of the waterfall model

- If something with the design does not work out, many difficulties will be faced in the implementation step.
- There are many uncertainties and risks associated with the steps.
- It is not the best model to follow for complicated projects.
- It does not help in long and continuously updated projects.
- It is not a suitable project for projects which require updates after short periods.



3. 5) Languages constructs

- Data types: by default visual studio variable are variant types of data. The data type variant can store integer, date time, or string when a variable is declared, a data type that specifies. For visual studio including variant are integer, long, single, double string, currency byte and Boolean.

A list of the data types for visual studio is given below.

1. Numeric

Byte	Store integer values within the 255 range
Integer	Store integer values in the (-32,768)-(+32,767) range
Long	Store integer values in the (-2,147,483,648)-(+2,147,483,648) range
Single	Store floating point values in the (-3.4x10 ⁻³⁸)-(+3.4x10 ³⁸) range
Double	Store large floating value that surpasses the single data from value
Currency	Stores values related to currency. It has 4 digits after decimal and 15 digits before it.

2. String

A variable longitude string can store about 4 billion characters

3. Date

A variable declared as data form can store both data and the date values from 01/01/0100 to 12/31/999 and use this to store the time and date values.

4. Boolean

Boolean types of data either a true or a false value. These numeric values are not stored and cannot be used as such. Values are stored internally as one (true) and zero (false), and every non-zero value is considered true.

5. Variant

Store any data type, and is the default type of visual studio data. In visual data if we default to declare a variable without any data, type the data type is assigned.

II. Variables

Variable **are** used to store information in a computer program which is to be indexed and manipulated. In addition, they provide a way to label data with a descriptive so that the reader and over selves can understand our programs more its sole purpose is to memorize and label data. For computing programming, naming variable is regarded as one of the variables. When naming variable, consider the names intensely. Try to make sure the give attribute to another reader is correctly concise and understandable. Even when you review a program or even years ago the other reader is yourself. When assigning a variable, use the symbol =.

For example,

Irb: 001 > middle name = 'Jane'

⇒ "Jane"

Here we assigned middle_name to the variable the value 'Jane' which is a string.

Irb: 002 > middle_name

⇒ "Jane"

we have now stored the 'Jane' string in memory for use in the program.

Irb: 001 > i = 2

⇒ 2

Irb: 002 > k = b

⇒ 2

Irb: 003 > i = 5

⇒ 5

III. Constants

A value that stays the same after a system is introduced for public use in the world is called a constant. They are of two types:

- ⇒ Literal constants: Values that are entered in the source code are called literal constants. For example, 'Hello, player4567'.
- ⇒ **Named constants**: Values in which a name represents a literal constant is called a named constant. For example, the 'last level' of the game is referred as 100.

IV. Conditional statement

11 This is the ability to measure a variable against a value and to act in one way if the variable fulfils the number, if statements.

For example,

If num % = 0

Print num

V. Loops

Software programs use program constructs that are used to ensure smooth working of the program and they are of three types:

- I. Sequence: This construct controls what statement will be displayed next. T is the statement that comes after the first statement or any statement you are reading. You can also skip to any other statement by using a goto statement (eg, GOTO 400 which makes the system to go to statement 400).
- II. Selection: This is used for selection between two different statements. It is of three types:
 - If statements
 - Case statements
 - Pattern matching
- III. Repetition: It is also called looping construct. This construct makes the statement be repeated many times until the required result is gained. They are of two types:
 - Fixed count loops: to repeat a statement for known number of times.
 - Variable count loops: to repeat a statement for unknown number of times.

Loops make the repeating tasks easier. There are three main types of loop:

- For: When this starts to work, it first ensures that the conditions to perform the command are fulfilled, if yes, the loop will start and it only stops when the result is gained.
- While: The test condition is also supposed to be fulfilled before the start of loop in this. In this case, if the conditions are true the loop starts but if it is false, the loop does not work. In case of true value of condition, the loop doesn't stop until the condition becomes false.
- Do While: In this loop, the conditions to carry out test are known after the loop is done. It is similar to While loop as it only works if the conditions are true.

VI. Arithmetic operators

The computer and its various applications make calculations easier. We can make a computer program to do any time of mathematical equations even if they are easy or complicated. There are some concepts that make understanding arithmetic in computer programs easier. Following are some arithmetic operations.

Operator	Description	Example
+	It adds two values together	1+4=5
-	It removes the second value from the first	5-4=1
*	It multiplies two values	8*2=16
/	It divides the numerator from denominator	14/2=7

VII. Relational operators

An operator that compares two values and forms a relationship between them is called a relational operator. This generally includes equalities and inequalities between two values. They help to create a smooth flowing program and when evaluated it creates a Boolean value. There are six common relational operators:

Operator	Meaning
<	It indicates that one value is lesser than the other

>	It indicates that one value is greater than the other
<=	It indicates that one value may be less than or equal to the other one
>=	It indicates that one value may be greater than or equal to other value
==	It indicates that the two values are equal
!= or <>	It indicates that the two values are not equal

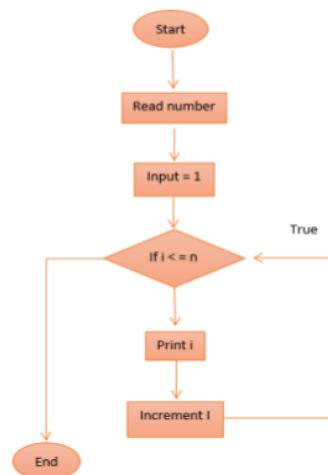


2. B) design

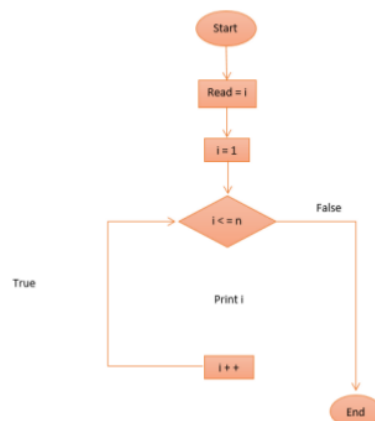
Flowchart:

A flowchart is a kind of diagram, describing a process or workflow.

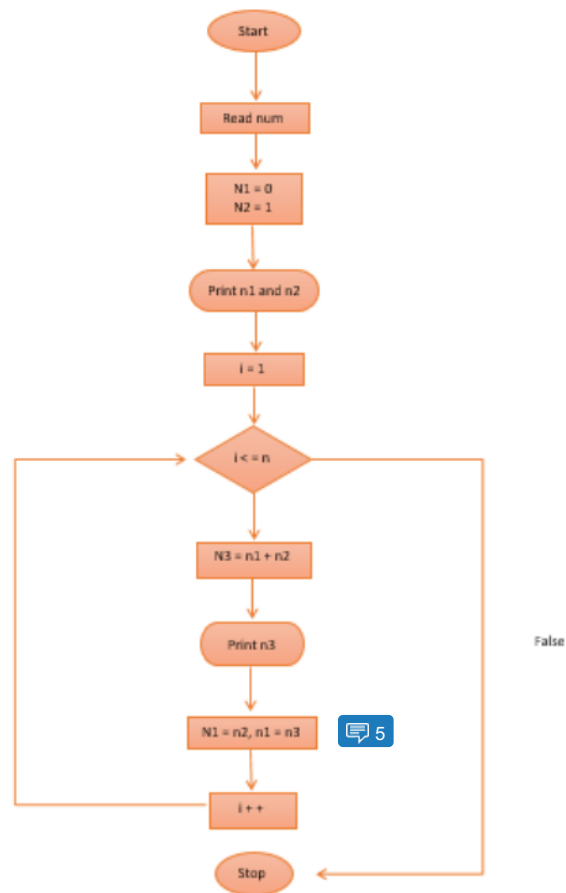
I. Area operation



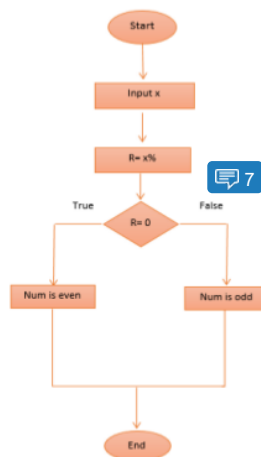
II. Perimeter operation



III. Fibonacci series



IV. Even or odd



Pseudo code

I. Fibonacci series

Start

Read n

Unitializen $n1 = 0, n2 = 1$

$i = 1$

Print $n1$ & $n2$

For loop until $i \leq n$

Calculate $n3 = n1 + n2$

Print $n3$

$N1 = n2, n2 = n3$

$i = i + +$

loop end

II. Even or odd

If $(r = 0)$

Then print num is even else

Print num is odd

End if

End

III. Area

$N = n1 = 0; n2 = 1,$

```
Int (i = 0; i < n; i++)
```

```
N3 = n1 + n2,
```

```
Display add (n3); n1 = n2, n2 = n3.
```

IV. Perimeter

```
Read = i
```



```
i = 1
```

```
if i <= n
```

```
then print i
```

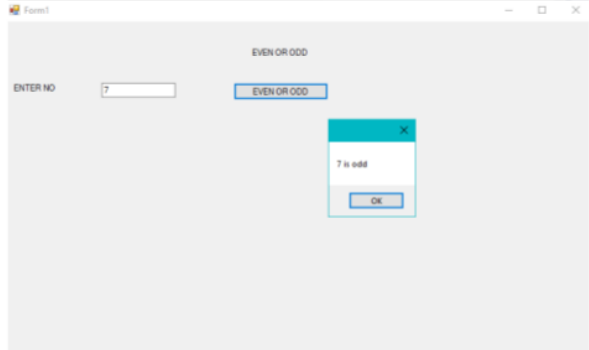
```
even else i++
```

```
end if end
```

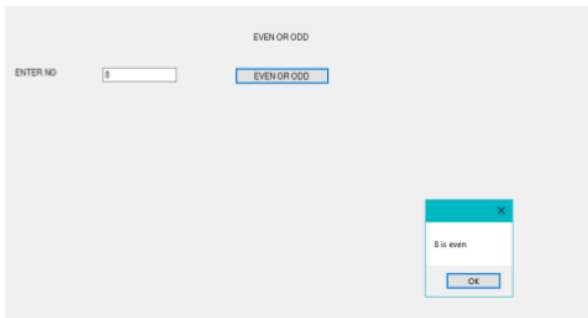
4. A) Testing

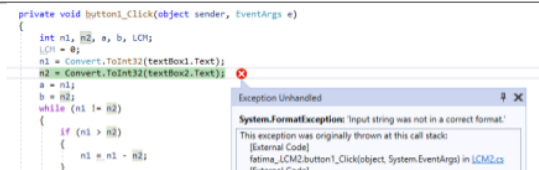
API NAME	VERSION	AUTHOR	TESTED BY	TESTING DATE
	1.0	FATIMA EJAZ BARRI	FATIMA EJAZ BARRI	

Test cases

Test ID	T2001
Module No	T2-EvenODD_Input
Test Name	EvenOdd_Valid_Odd_Input
Test scenario	Open form Enter number in the textbox txtno Click button
Text input	7
Expected result	7 is odd
Pass/ fail	Pass
Screenshot	

Test ID	T2002
---------	-------

Module No	T2-EvenOdd
Test Name	EvenOdd_Valid_Even_Input
Test scenario	Open form enter even number in the textbox txtno click button
Test input	8
Expected result	Message should display "8 is even"
Actual Result	Message "8 is even"
Pass/ fail	Pass
Screenshot	

Test ID	T4003
Module No	T4-GCD-LCM
Test Name	GCD_InValid_Input
Test scenario	Open form Enter A in one of the textbox txtno Click button
Test input	4 A
Expected result	Error message should display "enter correct format"
Actual result	System exited with errors
Pass/ fail	Fail
Screenshot	

10

4. b) Analyse ⁸Test Results

No.	Test No	Module No	Module name	Test input data	Expected result	Actual result	Pass/ fail

1	T1001	T1-Multiplication	T1-Multiplication table_Valid_Input	6	Multiplication table	Multiplication table	Pass
2	T2001	T2-EvenOdd	EvenOdd_Valid_Even_Input	7	Message should display 7 is odd	Message 7 is odd	Pass
3	T2002	T2-EvenOdd	EvenOdd_Valid_Even_Input	8	Message should display 8 is even	Message 8 is even	Pass
4	T4003	T4-GCD-LCM	GCD-LCM InValid_Input	A	Error Message	System Crash	Fail



Reference



<https://www.redhat.com/en/topics/api/what-are-application-programming-interfaces>

<https://techolution.com/types-of-apis/>

<https://bbvaopen4u.com/en/actualidad/8-advantages-apis-developers>

<https://www.webopedia.com/TERM/A/API.html>

<https://www.freetutes.com/learn-vb6/lesson2.html>

<https://launchschool.com/books/ruby/read/variables>

https://www.tutorialspoint.com/computer_programming/computer_programming_operators.htm

<https://press.rebus.community/programmingfundamentals/chapter/relational-operators/>

<https://www.bbc.co.uk/bitesize/guides/zc6s4wx/revision/5>

<https://opentechschoool.github.io/python-beginners/en/conditionals.html>

<https://www.sciencedirect.com/topics/computer-science/looping-construct>

<https://cgi.csc.liv.ac.uk/~frans/OldLectures/2CS45/progCons/progCons.html>

ORIGINALITY REPORT

24%

SIMILARITY INDEX

17%

INTERNET SOURCES

1%

PUBLICATIONS

23%STUDENT PAPERS

PRIMARY SOURCES

1**Submitted to Western International College
(WINC London)**

Student Paper

10%

2**inglesinstrumentalna.blogspot.com**

Internet Source

3%

3**blog.testlodge.com**

Internet Source

2%

4**ms-office2k32k7.blogspot.com**

Internet Source

2%

5**www.scribd.com**

Internet Source

1%

6**Submitted to Universiti Teknologi MARA**

Student Paper

1%

7**Submitted to South Birmingham College**

Student Paper

1%

8**Submitted to Bolton Institute of Higher
Education**

Student Paper

1%

9	kidsadda.com Internet Source	1%
10	Submitted to RDI Distance Learning Student Paper	1%
11	Submitted to Grwp Llandrillo Menai Student Paper	1%
12	Submitted to University of Warwick Student Paper	<1%
13	Submitted to Curtin University of Technology Student Paper	<1%
14	Submitted to Newcastle College, Tyne & Wear Student Paper	<1%
15	Submitted to BMTC Student Paper	<1%

Exclude quotes On
Exclude bibliography On

Exclude matches < 4 words

FINAL GRADE

/100

GENERAL COMMENTS

Instructor

Overall satisfactory effort. Application is incomplete. Sources are not in Harvard format. Intext citation is not performed.

Task	Crit	Achieved	Remark
No	eria		
1.a	1.1	Yes	API described, and evaluated. SDLC
1.b	1.2	Yes	described and waterfall model explained. Student has produced a plan of the program
2.a	1.3	Yes	which explains the language constructs which are to be used

			Flowchart: Perimeter, Area are wrong. correction needed: Fibo Even Odd : x% ????? Pseudocode: Area and Perimeter are wrong. Fibo is correct. Even Odd is incomplete. GCD, LCM is missing
2.b	2.1	No	
3.a	2.2	No	Incomplete Application
3.b	1.3	No	Not achieved Describe Testing, Unit Testing, Integration, System, Acceptance Testing. Incomplete test cases.
4.a	3.1	No	Not complete
4.b	3.2	No	No attempt
4.c	3.3	No	

PAGE 1

PAGE 2

PAGE 3



Comment 1

API described. and evaluated.

Task 1a achieved

PAGE 4

PAGE 5

PAGE 6



Comment 2

SDLC described and different models explained.

Task 1b achieved

PAGE 7

PAGE 8

PAGE 9



Comment 3

Student has produced a plan of the program which explains the language constructs which are to be used

Task 2a achieved



Comment 4

Flowchart:

Perimeter, Area are wrong.

correction needed: Fibo

Even Odd : x% ?????

PAGE 10



Comment 5

$n_2 = n_3$



Comment 6

Flowchart:

correction needed: Fibo

Even Odd : x% ?????

PAGE 11



Comment 7

Flw: Even Odd : x% ?????



Comment 8

Initialize

PAGE 12



Comment 9

Pseudocode:

Area and Perimeter are wrong.

Fibo is correct.

Even Odd is incomplete.

GCD, LCM is missing in flowchart & Pseudocode

Task 2b not achieved

PAGE 13



Comment 10

Many test cases missing

PAGE 14



Comment 11

Describe Testing, Unit Testing, Integration, System, Acceptance Testing.

Test cases and Analysis is incomplete.

Task 4a and 4b not achieved.

Task 4c not attempt so not achieved



Comment 12

Sources are not in Harvard format