NTN madam

Final CSE 412

Date :

(2)

Suppose you are a project manager of an E-commerce website. The analyst found that among the actives of the external input and output function Types three (03) are simple and three (03) are average activities. There are in total two (02) external inquiries all of which are simple, five (05) internal logical file & transactions all of which are average and three (3) external interface files, one (01) is simple and others are complex. As for the fourteen - complexity (14) weighting factor, three (03) are not essential, seven (07) are moderately essential and the rest are very essential.

	S	A	C
Ex input	3	4	6
Ex Inquiry	3	4	6
Ex output	4	5	7
Ex Interfac file	5	7	10
Internal logical file	7	10	15

1) Estimate lines of code ?

try :-

very essential → 5

" → 4

moderately " → 3

slightly " → 2

not " → 1

External input — $\{(3 \times 3) + (3 \times 4)\} = 21$

External output — $\{(3 \times 4) + (3 \times 5)\} = 27$

External inquiry $(2 \times 3) = 06$

Internal logical file $(5 \times 10) = 50$

External interface file $\{(1 \times 5) + (2 \times 10)\} = 25$

UPF = 129

Total complexity adjustment : $\{(3 \times 1) + (7 \times 3) + (4 \times 5)\}$

$= (3 + 21 + 20)$

$= 44$

$PC = \{0.65 + (0.01 \times 44)\}$

$= 1.09$

$FP = (UPF \times PC)$

$= 129 \times 1.09$

$= 140.61$

$\simeq 141$

Hossain MIM 2021-1-60-071

Line of code : 141×57

$= 8037$

$= \frac{8037}{1000} = 8.037 \simeq 8$ KLOC (Ans)

Diligent Smart Khata

Basic

Date :

In this code organic mode because we know that 2-50 KLOC line of the code is organic model.

Effort (E) : $a^b (KLOC)^{b6}$

$= 2.4 (8)^{1.05}$

$= 21.30$ person month

Development time (D): $C_b (E)^{d_b}$

$= 2.5 (21.30)^{0.38}$

$= 7.99$

$\simeq 8$ month

Average staff size (SS): E/D

$= 21.30/8$

$= 2.67$

$\simeq 3$ person

productive (P): $(KLOC/E)$

$= 8/21.30$

$= 0.38$

$= (0.38 \times 1000)$

$= 380$ per person month. Ans

Example 01:- suppose that a project was estimated to be 400 KLOC. calculate the E and D for each of the 3 modes?

Ans :-

organic model:-

$E = 2.4 (400)^{1.05}$

$= 1295.31$ person month

$D = 2.5 (1295.31)^{0.38}$

$= 38.07$ month

Embedded:-

$E = 3.6 (400)^{1.20}$

$= 4772.81$ person month

$D = 2.5 (4772.81)^{0.32}$

$= 37.59$ month

Ans

Semi detached:-

$E = 3.0 (400)^{1.12}$

$= 2462.80$ person moth

$D = 2.5 (2462.80)^{0.35}$

$= 38.45$ moth

Diligent Smart Khata

Date :

Example :02

A project size of 200 KLOC is to be developed. software development team has average exprience on similar type of projects. The project schedule is not very tight. calculated E, D, SS and P ?

Ans :—

200 KLOC project size thats means it is semi dete detached model.

$E = 3.0 (200)^{1.12}$

$= 1133.12$ person month

$D = 2.5 (1133.12)^{0.35}$

$= 29.30$ month

$SS = E/D$

$= 1133.12/29.30$

$= 38.68$ person

$P = 200/1133.12$

$= (0.177 \times 1000)$

$= 177$ LOC per person month

Ans

Example :03

A new project with ~~em~~ estimated 400 KLOC embeded system has to be developed project manger has a choice of hiring from two pools of developers : 1) very highly capable with very little experience in the programming language being used.

Ans :—

$E = a_i (KLOC)^{b_i} * EAF$ (In the project 400 KLOC it is the Embedded model)

$= 2.8 (400)^{1.20}$

$= 3712.18$ person month $* 0.82 \times 1.14 = 3470.14$ person moth

$D = 2.5 (3712.18)^{0.32}$

$= 34.70$ month

$D = 2.5 (3470.14)^{0.32}$

$= 33.96$ month

$SS = E/D$

$= 3470/34$

$SS = E/D$

$= 3470.14/33.96$

$= 102.18$ person

$P = 400/3470.14$

$= 0.115 \times 1000$

$= 115$ per person moth

↓

LOC

Ans

Diligent Smart Khata

Or

Date :

Developers of low quality but lot of experience use programming language. what is the impact of hiring all developers from one or the other pool.

Ans)

$E = a_i (KLOC)^{b_i} * EAF$

$= \{2.8 (400)^{1.20}\} \times \{(1.29) \times (0.95)\}$

$= 4549.28$ person month (Ans)

$D = 2.5 (4549.28)^{0.32}$

$= 37$ months (Ans)

$SS = 4549.28 / 37$

$= 122.96$ person (Ans)

$P = 400 / 4549.28$

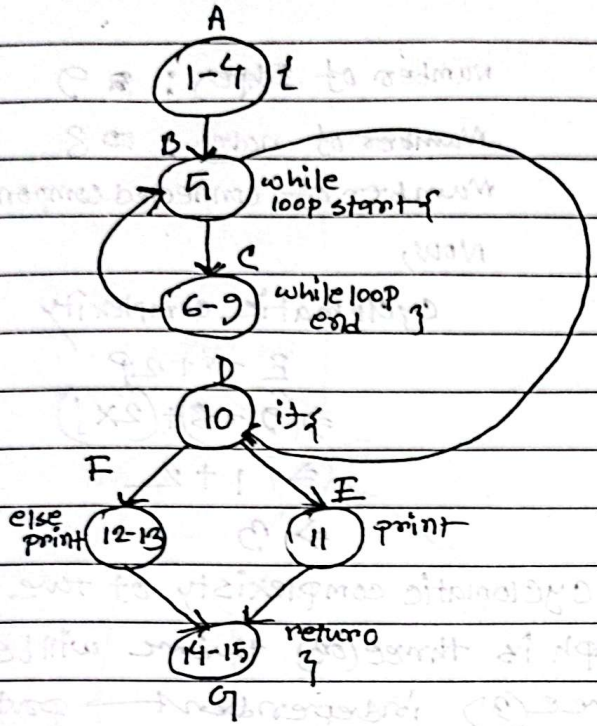
$= 0.0879$

$= 87.9$ LOC per person month (Ans)

(7) white

Assignment 01

Date :



Number of edges:- 8

Number of nodes:- 7

Number of connected component

Now,

cyclomatic complexity:

$E - N + 2P$

$\Rightarrow \{(8-7)+(2\times1)\}$

$\Rightarrow (1+2)$

$\Rightarrow 3$

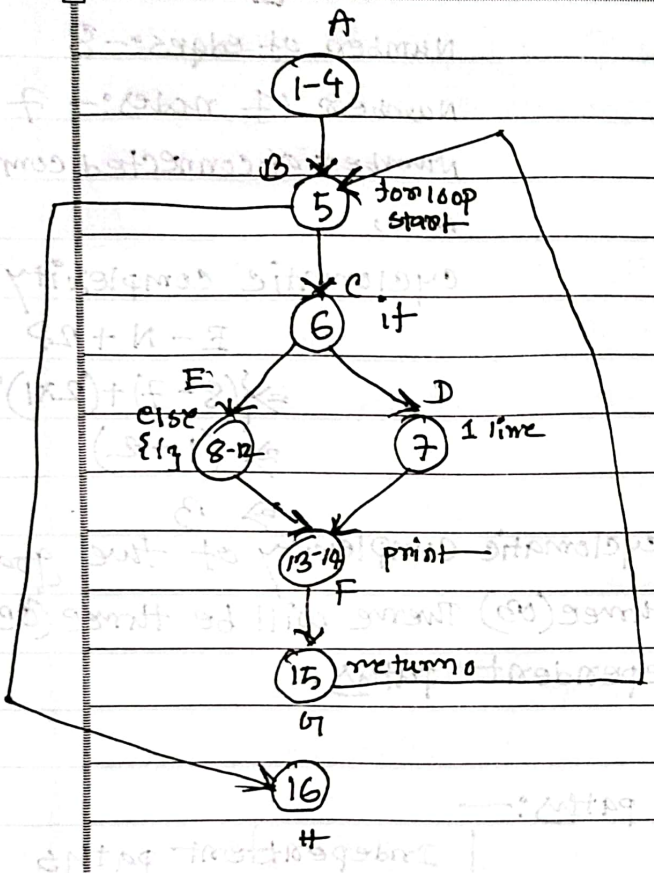
The cyclomatic complexity of the graph is three (03) There will be three (03) independent paths.

Test case design from independent paths:—

Test ID	Input	Result	Independent paths
01	0	0 is a palindrome	A → B → D → E → G
02	1	1 is a palindrome	A → B → C → B → D → E → G
03	10	10 is not a palindrome	A → B → C → B → D → F → G

Hossain Mim

Date :



Number of Edges : 9

Number of nodes : 8

Number of connected component

Now,

cyclomatic complexity

$E - N + 2P$

$\Rightarrow (9 - 8) + (2 \times 1)$

$\Rightarrow 1 + 2$

$\Rightarrow 3$

The cyclomatic complexisty of the graph is three (03) there will be three (3) independent paths.

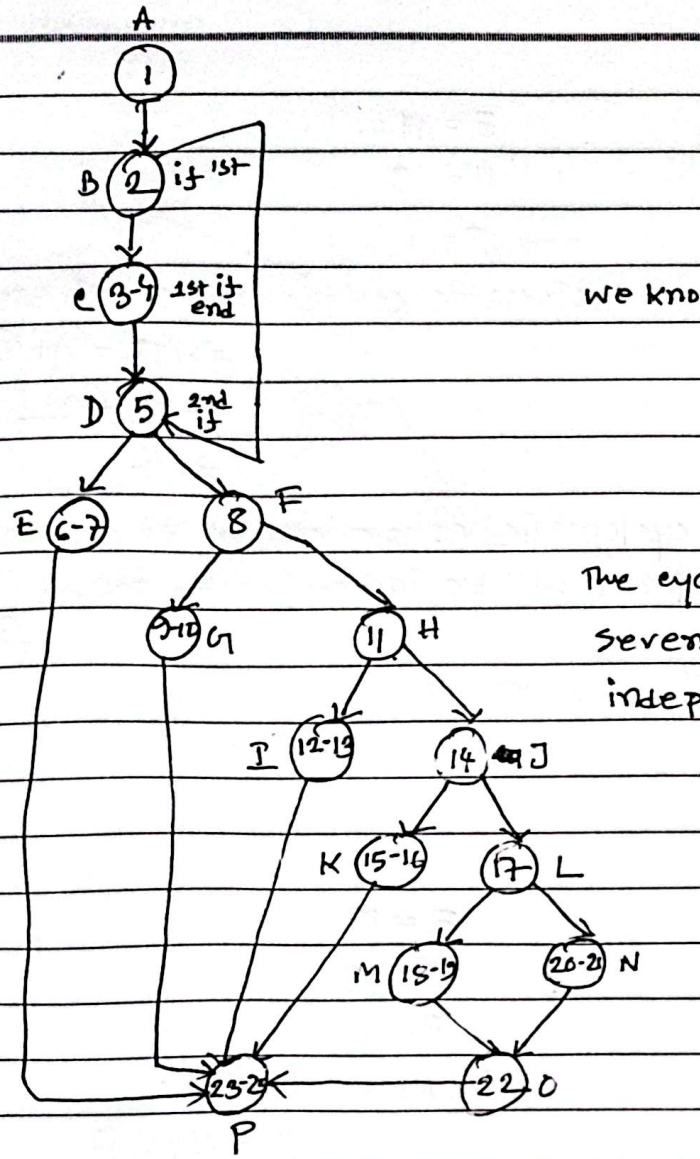
The test case design from independent path :-

Test ID	Input	Result	Indiependent path
01	0	First 0 terms of Fibonacci series :	H → B → H
02	1	First 1 terms of Fibonacci series : 0	A→B→C→D→F→G→B→H
03	2	First 2 terms of Fibonacci series : 0	A→B→C→E→F→G→B→H

Ans

Sec 1 : [illegible]

Date :



E = 21

N = 16

P = 1

we know that,

cyclomatic complexity : E - N + 2P

$\Rightarrow (21-16) + (2 \times 1)$

$\Rightarrow 5 + 2$

$\Rightarrow 7$

The cyclomatic complexity of the graph is seven (07) there will be seven (07) independent paths.

Hossain Mim

The test case design from independents paths :—

Test ID	Input Next day	Amount	Result	Independent paths
01	Yes	1000	1799.5 6994.50	A→B→C→D→E→P
02	No	999	994.50 1151.0915	A→B→D→F→G→P
03	No	1000	1785	A→B→D→E→P
04	No	199	226.67	A→B→D→F→H→I→P
05	No	99	116.127	A→B→D→F→H→J→K→P
06	No	49	59.81	A→B→D→F→H→J→L→N→O→P
07	No	24	199	A→B→D→F→H→J→L→M→O→P

Ans

n_1 ⇒ Number of operators N_1 = Total Number of occurrences op
n_2 ⇒ " " operands N_2 " " " operands

Date :

if (x[i] < x[j])

Calculate of operators

{

save = x[i];

x[i] = x[j];

x[j] = save;

}

Calculated of N:—

Occurrences operator(N_1):— if ,

if ([ ] < [ ])

{ = [ ] ; [ ] =

[ ] ; [ ] = ; }

Total occurrences operators : 24 (N_1)

Occurrences operands N_2 :—

x i x j save

x i x i x

j x j save

total occurrences operands : $N_2 = 14$

∴ program length is N : 24 + 14

⇒ 38 Ans

Distinct operator n_1 :—

if ([ ] <) { = ; }

Total distinct of operators n_1 : 10

total distinct of operands n_2 : x i j save

= 4

program vocabulary $n = n_1 + n_2$

= 10 + 4

= 14 → Ans

Diligent Smart Khata

Date :

i) program volume (V) : $N * \log_2(n)$

$\Rightarrow 88 * \log_2(14)$

$\Rightarrow 144.68$ Ans

ii) protential minimum volume (V*) : $(2+n_2^*) * \log_2(2+n_2^*)$ here, $n_2^* = 6$

$\Rightarrow (2+6) * \log_2(2+6)$

$\Rightarrow 8 * \log_2(8)$

$\Rightarrow 24$ Ans

iii) program level (L) : n_1 / n_2

$\Rightarrow 10/4$

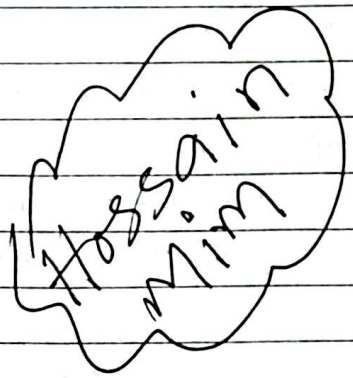
$\Rightarrow 2.5$ Ans

iv) program difficulty (D) : $(n_1/2) * (N_2/n_2)$

$\Rightarrow (10/2) * (14/4)$

$\Rightarrow 5 * 3.5$

$\Rightarrow 17.5$ Ans



v) program Effort (E) = V * D

$\Rightarrow 144.68 \times 17.5$

$\Rightarrow 2531.9$ Ans

vi) program Time (T) $\Rightarrow$ E/P

$\Rightarrow 2531.9/18$ [P = 18]

$\Rightarrow 140.66$ Ans

vii) Estimated program length is $\hat{N}$: $n_1 \log_2(n_1) + n_2 \log_2(n_2)$

$\Rightarrow 10 \log_2(10) + 4 \log_2(4)$

$\Rightarrow 41.2$

Ans

Diligent Smart Khata

A-5

Bve → 5
Robus → 7
worst → 5

Date :

Example 01:— An integer number range:— [1, 100] check prime Number

We know that,

Bve : 4n+1
⇒ (4+1)
⇒ 5

min : 1
min+ : 2
max : 100
max- : 99
nominal : 50-55

Test case design:—

ID	Integer	Result
01	1	Not prime
02	2	prime
03	100	Not prime
04	99	Not prime
05	53	prime

Ans

we know that;

Robust : 6n+1
⇒ 6+1
⇒ 7

min : 1
min+ : 2
min- : 0
max : 100
max- : 99
max+ : 101
nominal : 50-55

Test case design:—

01	1	Not prime
02	2	prime
03	0	Invalid
04	100	Not prime
05	99	Not prime
06	101	Invalid
07	53	prime

Ans

Example: 02

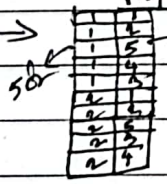
a, b range [1-10] b = [1-5]

Bve = 4n+1
= (4×2)+1
= 9

Bve → | 5 | N |
| N | 5 |

Robu → | 7 | N |
| N | 7 |

worst →



min : 1	1
min+ : 2	2
max : 10	5
max- : 9	4
nominal : 5	3

The case test design:—

01	1	3	1
02	2	3	8
03	10	3	1000
04	9	3	729
05	5	1	5
06	5	2	25
07	5	5	3125
08	5	4	625
09	5	3	125

Ans

Robust : 6n+1 ⇒ (6×2)+1 = 13

min : 1	1	
min- : 0	0	
min+ : 2	2	
max : 10	5	
max- : 9	4	
max+ : 11	6	
nominal : 5	3	

Test case design:—

01	1	3	1
02	0	3	Invalid
03	2	3	8
04	10	3	1000
05	9	3	729
06	11	3	Invalid
07	5	1	5
08	5	0	Invalid
09	5	2	25
10	5	5	3125
11	5	4	625
12	5	6	Invalid
13	5	3	125

Ans

worst = 5 = 25

01	1	1	1
02	1	2	1
03	1	5	1
04	1	4	1
05	1	3	1
06	2	1	2
07	2	2	4
08	2	5	32
09	2	4	16
10	2	3	8
11	10	1	10
12	10	2	100
13	10	5	100000
14	10	4	10000
15	10	3	1000
16	9	1	9
17	9	2	81
18	9	5	59049
19	9	4	6561
20	9	3	729
21	5	1	5
22	5	2	25
23	5	5	3125
24	5	4	625
25	5	3	125

Ans

Diligent Smart Khata

BVC → 

Robust → 

A, B, C

↓ rang [1-30]

Check large numbers

Date : .....

BVC: (4×3)+1

= 12+1

= 13

min	1
min+	2
max	50
max−	49
nominal	25-30

01	1	28	30	30
02	2	25	25	25
03	50	26	26	50
04	49	26	26	49
05	25	1	29	29
06	26	2	28	28
07	25	50	27	50
08	25	49	25	49
09	26	25	1	26
10	25	26	2	26
11	25	27	50	50
12	25	25	49	49
13	27	28	25	28

Robust: (6×3)+1

= 18+1

= 19

min	1
min−	0
min+	2
max	50
max−	49
max+	51
nominal	25-30

min	1
mint	2
max	100
max−	99
nominal	50-55

min	1
min−	0
mint	2
max	100
max−	99
max+	101
nominal	50-55

BVC: 4+1 = 5

01	1	Not prime
02	2	prime
03	100	Not
04	99	Not
05	53	prime

Robust

01	1	Not
02	0	invalid
03	2	prime
04	100	Not
05	99	Not
06	101	Invalid
07	53	prime

same → of worst case g.

01	1	25	28	28
02	0	27	29	Invalid
03	2	28	25	28
04	50	25	25	50
05	49	25	25	49
06	51	25	25	Invalid
07	25	1	28	28
08	25	0	29	Invalid
09	26	2	30	30
10	27	50	25	50
11	26	49	25	49
12	25	51	25	Invalid
13	25	25	1	25
14	25	25	0	Invalid
15	26	25	2	26
16	27	27	50	50
17	28	28	49	49
18	29	29	51	Invalid
19	30	30	25	30

Retailer

iv) Partitioning Data :

The decision table :

		R1	R2	R3	R4	R5	R6	R7	R8
Condition Stub	C1 : Days	T	T	T	F	F	F	F	F
	C2 : Agent	F	T	F	T	F	T	F	F
	C3 : Retailer	F	F	T	F	T	F	T	F
	C4 : amt ≥ 50000	T	T	T	F	F	F	F	F
	C5 : amt ≥ 20000 to < 50000	T	F	F	T	T	F	F	F
	C6 : amt < 20000	F	F	F	F	F	T	T	F
	C7 : Furniture	F	F	F	F	F	F	F	T
Action Stub	A1 : 5%							X	
	A2 : 8%					X	X		
	A3 : 10%	X		X					X
	A4 : 12%				X				
	A5 : 15%		X						

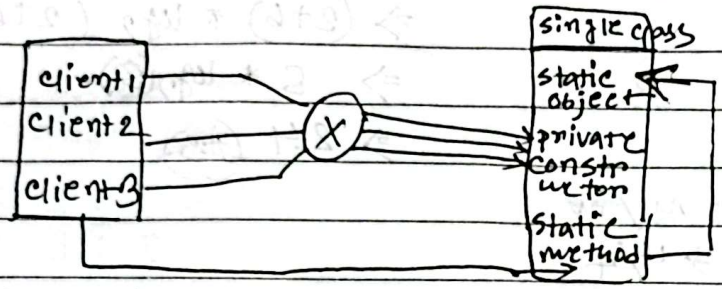
The test case devided from decision table

01	Days	50000	No	10%
02	Agent	52000	No	15%
03	Retailer	53000	No	10%
04	Agent	25000	No	12%
05	Retailer	26000	No	8%
06	Agent	15000	No	8%
07	Retailer	15000	No	5%
08	Days	50000	Yes	10%

Date :

Singleton Pattern :— It is probably the most widely used design pattern. Sometimes it is used in excess and in scenarios where it is not required!

The singleton design pattern ensures a class only has one instance and provides a global point of access to it.



i) Important for some classes to have exactly one instance.

ii) Ensure only one instance available and easily accessible

iii) Global variables gives access but doesn't keep your from instantiating many objects

iv) Give class responsibility for keeping track of itstance.


```
public class Database{
private static Database DB;
---
private Database() (---)
public static Database getDB(){
if (DB == Null)
    DB = new Database();
    return DB;
}
}
```

Implementation:—

i) Declare all of class constructors private

ii) prevent other classes from directly creating an instance of the class

iii) Hide the operation that creates the instance behind a class operation (getInstance)

Hossain Mim

Diligent Smart Khata

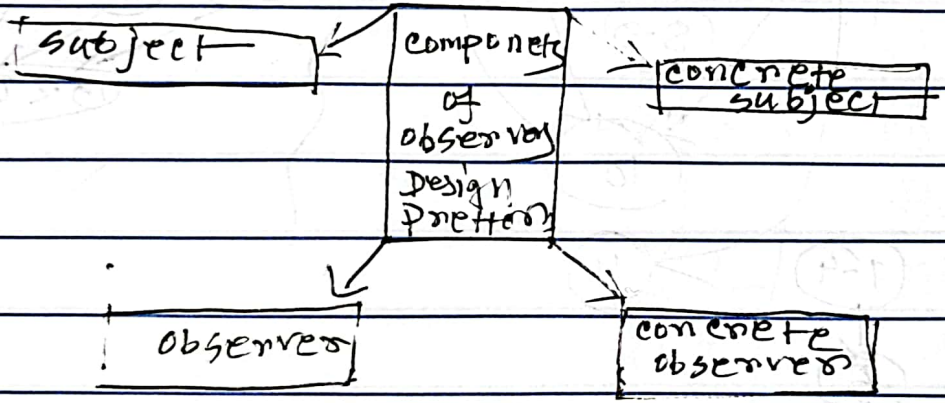
$= 8014.77/1000$ $= 380$ Any

$= 8.014$

$= 7.99$ months Any

$\simeq 8$ KLOC Any

The observer design pattern is behavioral design pattern defines a one to many dependency between objects so that when object changes state all its dependents are notified and update automatically. A subject (publishers) maintain a list of observers.



Advantage of the observers and subjects are independent, ~~without~~

- Observer get notified without manual intervention
- Easily add or remove observer

The builder (DP) is a creational pattern used in software construct a complex object step by step It allows the construction of a product in a step-by-step fashion, where the construction process can vary based on the type of product being build. The pattern separates the construction of a complex object from its representation allow the same construction process to create different representation.

ii) use Observer for event-driven or state-dependent-system

iii) use builders for complex object construction with various configurations.