INTRODUCTION TO SOFTWARE ENGINEERING

Date 10-2 2024

NUMERICALS

2. FUNCTION POINT FP

Function Point Estimation

- It is a metric which gives the degree of functionality delivered by the system.
- Function point is an indirect measurement of functionality of the system.
- FP is derived using an empirical relationship based on countable (direct) measures of software's information domain & assessment of software complexity

5 Components of Function Point.

Data Functions
- Internal Logical Files
- External Interface Files

Transactional Functions.
- External Inputs
- External Inquiries
- External Outputs

① No. of External Inputs (EIs):
EI originates from a user or transmitted from another application

② No. of External Inputs (EOs):
EO derives within the application & provides information to the user e.g. an error response

Date ______ 20

③ No. of External Inquiries (EQs):
An online input that results in the generation of some immediate software response.

④ No. of Internal Logical Files (ILFs):
A logical grouping of data that resides within the applications boundary & maintained via external inputs. e.g. databases.

⑤ No. of External Logical Files (ELFs):
A logical grouping of data that resides outside of the application but provides data to the application. e.g. some external database source.

Formula for F.P.

$$F.P. = \text{count total} \times [0.065 + 0.01 \times \sum(F_i)].$$

count total → sum of all FP entries.

0.065 & 0.01 → empirically derived constants.

F_i → ($i = 1$ to 14) value adjustment factors (VAF).

- There are 14 VAF, we have to rate each on the scale of 0 – 5 and then sum all 14 values.

Effort Calculation: Model Names

① $E = -91.4 + 0.355\ FP$ → Albrech & Gaffney

② $E = -37 + 0.96\ FP$ → Kemerer

③ $E = 0.054 \times FP \times 1.353$ → SM PEEM.

Date ____ 20__

Slide Example:

Information Domain Value	Count		Simple	Avg	Complex	=	
External Inputs (EI)s	10	x	3	4 ✓	6	=	40
External Outputs (EOs)	8	x	4	5 ✓	7	=	40
External Inquiries (EQs)	12	x	3	4 ✓	6	=	48
Internal Logical Files (ILFs)	6	x	7	10 ✓	15	=	60
External Interface Files (EIFs)	2	x	5	7 ✓	10	=	14

Count Total ⟶ + = 202
(Unadjusted Function Points (UFP))

Complexity weighing factors

1. Backup recovery 4
2. Data communication 1
3. Distributed processing 0
4. Performance Critical 3
5. Existing Operating Environment 2
6. Online data Entry 5
7. Input transaction over multiple screens 5
8. Master file updated online 3
9. Information domain values complex 3
10. Internal processing complex 2
11. Code design for reuse 0
12. Conversion/installation in design 1
13. Multiple installation 3
14. Application designed for change 5

$\Sigma Fi = 37$

$F.P = (0.65 + 0.01 \times 37) \times 202$

$F.P = 1.07 + 202$

$\boxed{F.P = 206.04}$

Efforts:

① $E = -91.4 + 0.355(206.04)$

$\boxed{E = -17.9558}$

② $E = -37 + 0.96(206.04)$

$\boxed{E = 160.7984}$

③ $E = 0.054 \times FP \times 1.353$

$\boxed{E = 15.05369448}$

Date

Example: Safe Home.

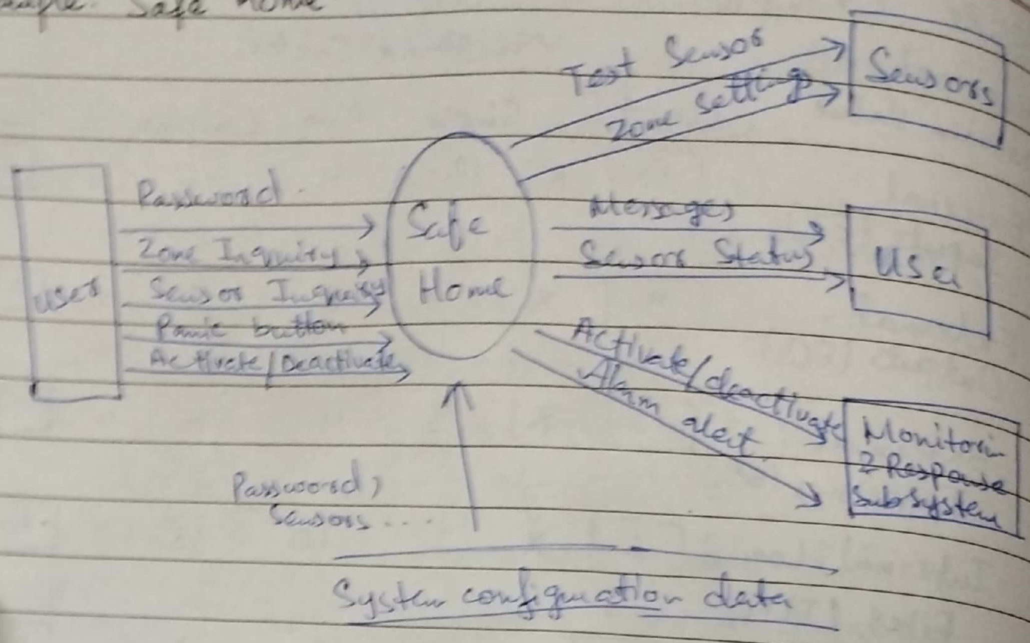


Figure: A flow model for safe home user interaction function.

Information Domain Value	Count	x	Simple	Avg	Complex	=	
			Weighing factors				
EIs	3	x	3 ✓	4	6	=	9
EOs	2	x	4 ✓	5	7	=	8
EQs	2	x	3 ✓	4	6	=	6
ILFs	1	x	7 ✓	10	15	=	7
ELFs	4	x	5 ✓	7	10	=	20

Count Total / Unadjusted Function Points - UFP = 50

∴ FP = Count Total x [0.65 + 0.01 Σ(Fi)]

Let all VAF = 2.

2 x 14 = 28 => ΣFi

Date ______ 20__

$F.P. = 50 \times [0.65 + 0.01(28)]$

$F.P. = 50 \times 0.93$

$\boxed{F.P. = 46.5}$

Efforts.

① $E = -91.4 + 0.355(46.5) = -74.8925$

② $E = -37 + 0.96(46.5) = 7.64$

③ $E = 0.054 \times (46.5) \times 1.353 = 3.397383$.

Ex 2: Given the following value, calculate the functional point when complexity adjustment factors are significantly complex product & weighing factors are high.

User input = 55 User Output = 35

User inquiries = 40 User Files = 8

External Interfaces = 5.

solve

Info Domain Value	Count	x	Simple	Avg	Complex	=	
EIs	55	x	3	4	6	=	330
EOs	35	x	4	5	7	=	245
EQs	40	x	3	4	6	=	240
ILFs	8	x	7	10	15	=	120
ELFs	5	x	5	7	10	=	50
						+	$\boxed{985}$

Each VAF is 4. => $14 \times 4 = 56$

$FP = 985 \times [0.65 + 0.01(56)]$.

$FP = 985 \times 1.21$

$\boxed{FP = 1191.85}$

Efforts:

① $E = -91.4 + (1191.85)^{\times 0.355} = \cancel{1100.45}$ 331.70675

② $E = -37 + 0.96(1191.85) = 1107.176$

③ $E = 0.054 \times 1141.85 \times 1.353 = 87.0789447$

3. Performance Evaluation & Review Technique
Critical Path Method
→ Determining Critical Path
→ Free Slack
→ Total Slack
→ Earliest Start Time (EST)
→ Earliest Finish Time (EFT)
→ Latest Start Time (LST)
→ Latest Finish Time (LFT)
} For each activity

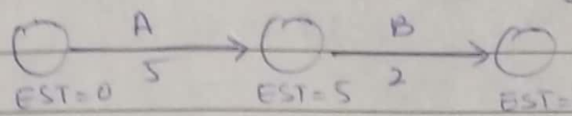
- Critical Path Method (CPM) or Critical Path Analysis is a project network analysis technique used to predict total project duration.
- CPM for a project determines the earliest time by which a project can be completed.
- It is the longest path through the the network diagram & has the least amount of slack.
- Slack / float is the amount of time an activity can be delayed without impacting other activities
- If the slack begins to decrease, this means the activity is taking longer than anticipated
  Slack = LST - EST or LFT - EFT
- Critical path is the longest path through the network diagram.
- If an activity on the critical path takes longer than planned, the whole project will be delayed.
- Activities not on critical path can only be delayed till the slack value / time is > 0. After that it will also effect the project

① A forward pass determines the early start & early finish dates for an activity.
② A backward pass determines the late start & late finish dates for an activity.

Date ______ 20___

EST Calculation.

The earliest time, an activity can begin after its predecessor

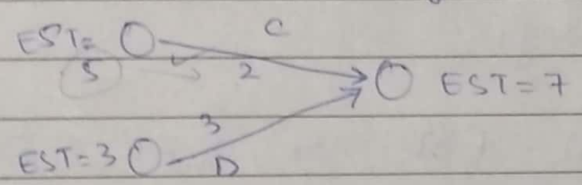


If (only one arrow coming into a node).

EST = EST of prev node + time estimate

else if (more than one arrow coming into a node)

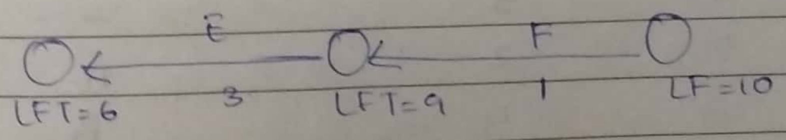
EST = prev node with greater EST + Time estimate



- when all ESTs are calculated, the EST of last node is project's duration.
- Project duration is also the LFT of last node
- Calculation is reversed for LFT's of nodes

LFT Calculation:

- Latest time an activity can end without delay.
- Start with last node that has LFT = Project duration.

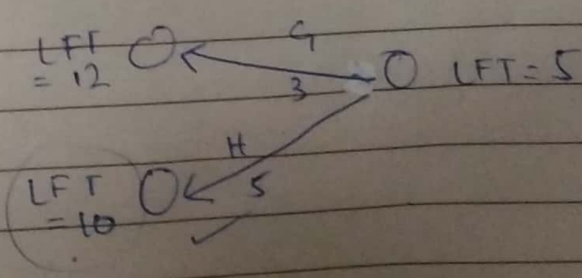


if (one arrow)

LFT = LFT of prev node - Time estimate

else if (more than one arrow).

LFT = Prev node with smaller LFT - Time estimate



Date

Activity Times.

EST = 5 ○ $\xrightarrow[4]{X}$ ○ LFT = 12

- Available Time = LFT − EST
  for X => 12 − 5 = 7
- Total Slack = Available time − time estimate.
  Slack for X = 7 − 4 = 3.
- Activity X has a slack of 3, can start within 3 days.

Slack = LST − EST or LFT − EFT

Activity Name	Duration
EST	EFT
LST	LFT

LST = Duration − LFT

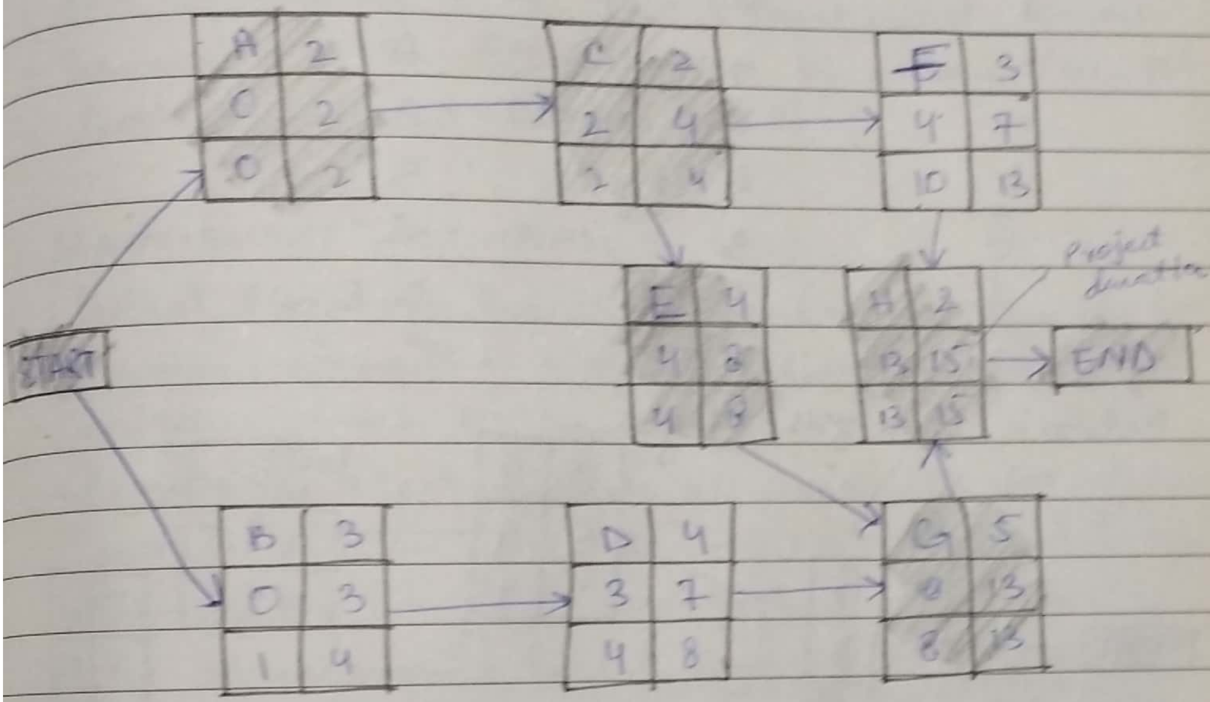
Duration = EFT − EST

Slide example.

Task	Duration	Predecessors
A	2	–
B	3	–
C	2	A
D	4	B
E	4	C
F	8	C
G	5	D, E
H	2	F, G

Date______ 20____

Network Analysis Diagram.



Activity	EST	EFT	LST	LFT	Slack	On Critical Path
A	0	2	0	2	0	Yes
B	0	3	1	4	1	No.
C	2	4	2	4	0	Yes
D	3	7	4	8	1	No.
E	4	8	4	8	0	Yes.
F	4	7	10	13	6	No
G	8	13	8	13	0	Yes.
H	13	15	13	15	0	Yes.

Critical Path

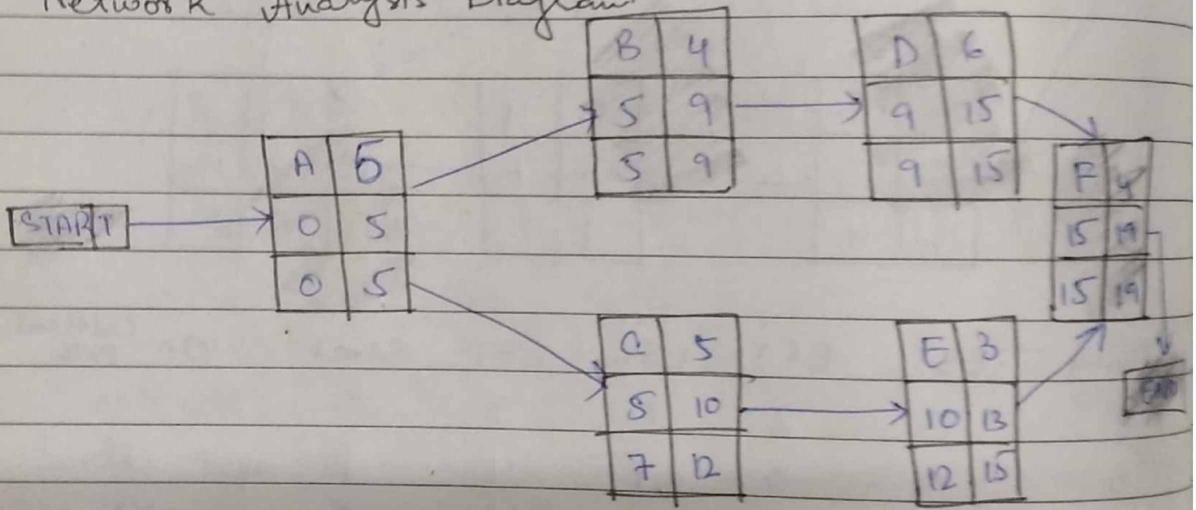
Start → A → C → E → G → H → End.

Date

Q.

Activity	Duration	Predecessors
A	5	—
B	4	A
C	5	A
D	6	B
E	3	C
F	4	D, E
~~G~~		

Network Analysis Diagram



Activity	EST	EFT	LST	LFT	Slack	On Critical Path
A	0	5	0	5	0	Yes
B	5	9	5	9	0	Yes
C	5	10	7	12	2	No
D	9	15	9	15	0	Yes
E	10	13	12	15	3	No
F	15	19	15	19	0	Yes

Critical Path

~~A~~ Start → A → B → D → F → End.