Final

Q/ consider the attribute "numbers of bug found" for software testing. ① Define an ab-solute measure scale for this attribute

Ans-1 ① Define the scale

- zero point :- Zero if no bugs found (mean (meaningful zero point on scale)
- Unit :- number of bugs

② Scale :- (0, 1, 2. . . .)

zero point :- 0 if No bugs

Unit :- each number represents bugs found in testing

(11) why it is not an abosulate scale measure of program correctness ?

Ans 2:- number of bug is an emphirical measure doesn't directly measure correctness

① ~~All~~ Not All bugs are equal

② False Negative / Positive

③ Incomplete Information

chapter-4

measuring the size of software

measuring the lines of code

NKLOC → Non commented thousand lines of code

LLOC → logical lines of code

what is Language gearing factor?

→ compares the expressiveness of Language and takes into account the differences in productivity in Language

The more productive a Language → The fewer lines of code you should write

→ more gearing factor ∝

→ Less Lines of code

2005

1991 Gearing Factor

Language	1991 Gearing Factor	2005
C	128	148
C++	55	60
JAVA	53	60
HTML 3.0	15	43

Reused code classification schema

1. Reusued Verbatim
2. Slightly modified (< 25%)
3. Extensively modified (≥ 25%)
4. New

Functional Points

counting Function points

FPA :- Functional component Analyis

→ is a sottwane metnic on techinique that measune the functiond size of a sottwane application based on functionality pnovided to end usen

UFPs :- Unadjusted Function Points

→ ane counted by summing all complexity nating

VAF :- Value Adjustment Facton

→ is calculated based on the complexity of overall system

AFP :- Adjusted Function Points

calculated by multipying VAF by UFPs

AFP = UFPs x VAF

Function points components

1. Input
2. Output
3. Data Files / Logical Files
4. Intenface
5. Inquinies / Requests

Function point types

1. Simple
2. Avenage
3. complex

Final

Q You have a small pnognam with 4 simple inputs, 1 data file (trival) and 2 outputs both of avenage complexity. How many UFPs UNadjusted Function points would be?

component	simple	Avenage	complex
Inputs	[✓ 3]	4	6
outputs	4	[✓ 5]	7
Data files	[✓ 7]	10	15
Intenface	5	7	10
Inquinies	3	4	6

Ans :- Inputs = 4 × 3 = 12

Outputs = 2 × 5 = 10

Data files = 1 × 7 = 7

∴ Total UFPs = 12 + 10 + 7

= 29

Ans

calculeted based on GSC

$$VAF = \Sigma\left(\text{General system characteristic}_i \times \text{rate}\right)$$

1 × rate

~~= No of General~~

rate 0 to 5 :- 0 → no infuence
5 → more influence

General system Characteristic (14)

1. Data communication
2. Distributed Data
3. Performance objective
4. Heavily Used configuration
5. Transaction Rate
6. Online Data Entry
7. End User Efficiency
8. Online Update
9. Complex processing
10. Reusability
11. Conversion Ease
12. operation Ease
13. multiple site use
14. Facilitate change

$$AFPs = UFPs * (0.65 + 0.01 * VAF)$$

Q You have a small project that has UFPs = 25. In GSC all factors are scored as 1 and complex processing and data communication scores 5. How many AFPs do you have?

2

$$\boxed{\text{VAF}} = (14-2) \times 1 + (2 \times 5)$$

$$= 12 \times 1 + 10$$

$$= 22$$

$$\text{UFPs} = 25$$

$$\boxed{\text{AFPs} = \text{UFPs} * (0.65 + 0.01 * \text{VAF})}$$

$$= 25 * (0.65 + 0.01 \times 22)$$

$$= 21.75$$

Ans

$$\text{Schedule} = FP^{0.4}$$

$$\text{Staff} = \frac{FP}{150}$$

$$\text{Effort} = \text{staff} \times \text{schedule}$$

#Software Productivity Research

#SPR Function Point complexity adjustment Factor

Program complex Rating

Rate ↓

1. All simple Algorithm + simple calculation
2. Majority of simple Algorithm + simple calculations
3. Average complexity of Algorithm and cAlculation
4. Some difficult/complex algorithm + calculations
5. Many difftult algorithm + complex calculations

PC=5

Data complexity Rating

DC=1

1. Simple data with few elements and Relationship
2. Numerous variable and constant data items but simple relationships

③ Average complexity with multiple files, field and data relationships

④ Complex file structure and complex data Relationships

⑤ Very complex file structures and very complex data relationship

— o —

Q A program with UFP = 35. The program has many difficult algorithms but simple data with few elements / relationship. what is $\boxed{\text{AFP count}}$ using SPR FP complexity adjustment method?

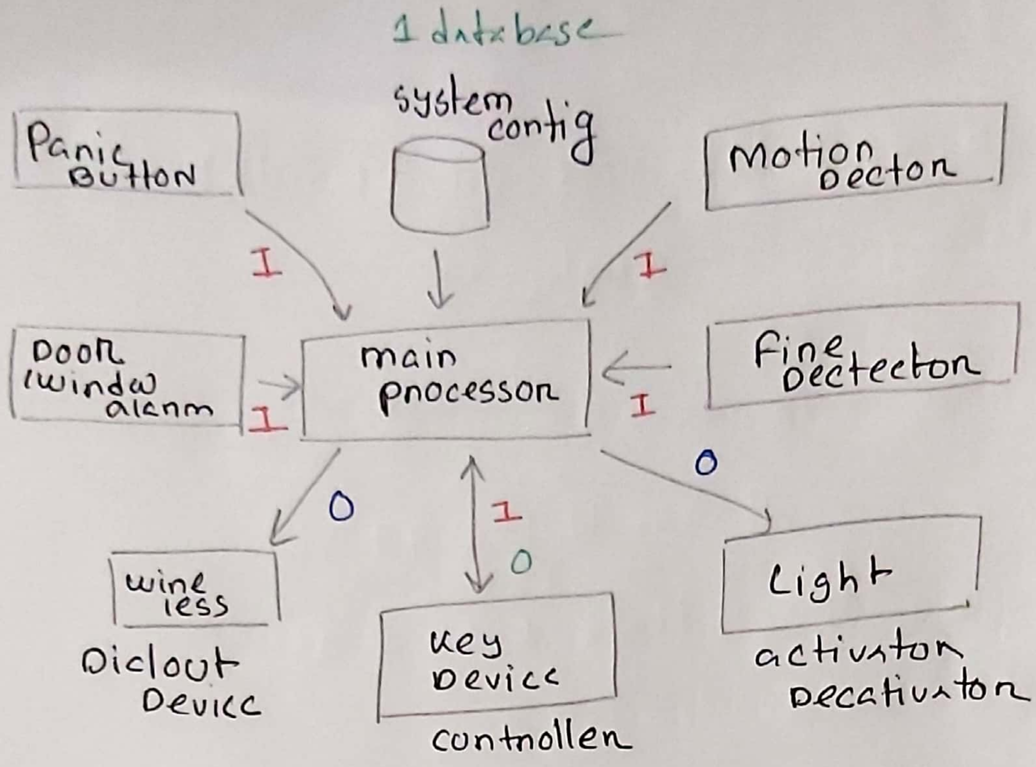
PC = 5

dC = 1

$$\boxed{AFPs = UFPs\,(0.4 + 0.1\,(DC + PC))}$$

$$= 35\,(0.4 + 0.1\,(1 + 5))$$

$$= 35$$



Home security anchectune

I = Input
O = output
Sim

O = medium output

① Ans:-

Inpot = 5 x 3 (simple)

Output = 2 x 4 (simple) + 1 x 5 (avenag

Database = 1 x 7 (simple)

∴ $\boxed{\text{UFPs} = 35}$

Ans

②

GSC	Rate
Data commonication	3
Diptributed Functions	2
Performance	3
Heavily used configuration	1
Transaction Rate	0
online Data Entry	1
End user efficiency	1
online update	1
complex processing	1
Resuasibility	3
Installation Ease	5
operational Ease	5
multiple sites	5
Facilation of change	2

$\boxed{VAF = 33}$

③ $AFP_s = 35 * (0.65 + 33 \times .01)$

$\boxed{AFP = 34.3}$ Ans

④ Estimate the effort to build the system :-

Given,

10 FP / staff month

SD = ± 1 staff month

estimated number of staff month

$$= \frac{AFP_s}{10}$$

$$= \frac{34.3}{10}$$

$$= 3.43$$

$$SD = \frac{34.3}{10+1} \text{ or } \frac{34.3}{10-1}$$

$$= 3.81 \text{ or } 3.12$$

effort is between 3.12 to 3.81 staff month