UNIT-4 SOFTWARE METRICS

Q What is Software Metrics ?

Ans:- The continuous abblication of measurement based techniques to the software development process and its products to supply meaningful and timel management information, together with the use of those techniques to improve that process and its products.

Categories Of Metrics

1 Product metrics :- describe the characteristics of the product such as size, complexity, design features, performance, efficiency, reliability, portability etc.

2 process metrics :- describes the effectiveness and quality of the processes that produce the software product.

Examples are :-

- effort required in the process.
- time to produce the product
- effectiveness of defect removal during development.
- number of defects found during testing
- maturity of the process.

- number of software developers
- staffing pattern over the life cycle of the software.
- Cost and schedule
- productivity.

Some metrics belong to multiple categories like quality metric may belong to all three categories. It focuses on the quality aspects of the product process and the project. Some important metrics are discussed in subsequent sections of the chapter.

Q What are the Areas of Application?

Ans: The most established area of software metrics is cost and size estimation techniques. There are many proprietary packages in the market that provide estimates of software system size, cost to develop a system, and the duration of the development or enhancement of the project.
These packages are based on estimation models like COCOMO 81, COCOMO-II developed by Berry Boehm [BOEH 81].

Metrices for Source Code :- LOC

Halstead Metrics

Halstead ~~Metrics~~ Complexity measures are Software metrics introduced by Maurice Halstead in 1977.
Halstead made the observation that metrics of the Software should reflect the implementation or expression of algorithms in different languages, but be independent of their execution on a specific platform.
These metrics are therefore computed statically from the Code.

- Goal :- His goal is to identify measurable properties of software and relations between them.

This is similar to the identification of measurable properties of matter (like the volume, mass and pressure of a gas) and the relationships between them (analogous ~~of~~ the gas equation).
Thus his metrics are actually not just complexity metrics.

η_1 =

η_2 = The number of distinct operands

N_1 = The total number of operators

N_2 = The total number of operands

From these numbers, several measures can be Calculated:

Program Vocabulary : $\eta = \eta_1 + \eta_2$

Program length : $N = N_1 + N_2$

- Calculated estimated program length :

$$\hat{N} = \eta_1 \log_2 \eta_1 + \eta_2 \log_2 \eta_2$$

- Volume : $V = N \times \log_2 \eta$

- Difficulty : $D = \frac{\eta_1}{2} \times \frac{N_2}{\eta_2}$

- Effort : $E = D \times V$

~~The difficulty measures translate into actual Coding time using the following relation.~~

The difficulty measures is related to the difficulty of the program to write or understand. eg when doing code review

The effort measure translates into actual coding time using the following relation.

- Time required to program : $T = \frac{E}{18}$ seconds

Halstead's delivered bugs (B) is an estimate for the number of errors in the implementation.

- ~~Num~~ Number of delivered Bugs : $B = \frac{E^{2/3}}{3000}$ or,

more recently $B = \frac{V}{3000}$ is accepted

Example :-

Consider the following C program:-

```
main ()
{
  int a, b, c avg;
  Scanf ("%d %d %d", &a, &b, &c);
  avg = (a+b+c)/3;
  printf ("avg = %d", avg);

}
```

Solution :-

The distinct operators (η_1) are = main, (), {}, , int, scanf, &, =, +, /, printf, ", ;

The distinct operands (η_2) are = a, b, c, avg, "%d %d %d", 3, "avg = %d"

- $\eta_1 = 12$, $\eta_2 = 7$

$\eta = \eta_1 + \eta_2$

$\eta = 12 + 7 = 19$

$N_1 = 27$ $N_2 = 15$

$N = 42$

Calculated Estimated Program length :-

$\hat{N} = 12 \times \log_2 12 + 7 \times \log_2 7$

$\hat{N} = 62.67$

Volume : $V = N \times \log_2 \eta$

$V = 42 \times \log_2 19$

$V = 178.4$

Difficulty : $D = \frac{\eta_1}{2} \times \frac{N_2}{\eta_2}$

$D = \frac{12}{2} \times \frac{15}{7} = 12.85$

Effort : $E = 12.85 \times 178.4 = 2292.44$

Time required to Program :- $T = \frac{E}{18}$

$T = \frac{2292.44}{18}$

$T = 127.357$ seconds

Number of delivered bugs :-

$B = \frac{V}{3000}$

$B = \frac{2292.44^{2/3}}{3000} = 0.05$