



What's what PSPM

EISSN: 2756-7729 = 22

MARCH 2023/ VOL 1

$(a+b)^2 = a^2 + 2ab + b^2$
 $\frac{a}{b} = \frac{c}{d} \Rightarrow \frac{ad}{bd} = \frac{bc}{bd}$
 $\frac{a}{b} = \frac{c}{d} \Rightarrow \frac{ad-bc}{bd}$
 $\frac{a}{c} = \frac{b}{d} \Rightarrow \frac{ad}{cd} = \frac{bc}{cd}$
 $\frac{a}{c} = \frac{b}{d} \Rightarrow \frac{ad-bc}{cd}$
 $\frac{a}{b} + \frac{c}{d} = \frac{ad+bc}{bd}$
 $\frac{a}{b} - \frac{c}{d} = \frac{ad-bc}{bd}$
 $\frac{a}{b} \cdot \frac{c}{d} = \frac{ac}{bd}$
 $\frac{a}{b} \div \frac{c}{d} = \frac{ad}{bc}$
 $x^2 - 4x \leq 0$
 $y = ax$
 $(0,1)$
 (b)
 (b)
 $n(B) = 68$
 $n(C) = 84$
 $n(B \cup C) = n(B) + n(C) - n(B \cap C)$
 $M = \frac{0.046765 \text{ mol} = 0.016 \text{ M}}{3 \text{ OL}}$
 $2\text{SO}_2 + \text{O}_2 \rightleftharpoons 2\text{SO}_3$
 $\text{C}_3\text{CO}_3 \rightleftharpoons \text{C}_3\text{O} + \text{CO}_2$
 $\text{He} = 4.002602$
 $\text{Na} = 22.989769$
 $\text{Ar} = 39.948$
 $z_1 = a \frac{\begin{vmatrix} D_1 & B_1 \\ D_2 & B_2 \end{vmatrix} - b \begin{vmatrix} D_1 & A_1 \\ D_2 & A_2 \end{vmatrix}}{a^2 + b^2 + c^2}$
 $E = MC^2$
 $\sqrt{5 + \sqrt{24}} = \sqrt{5 + \sqrt{6 \cdot 4}}$
 $f(x) = a(x-x_1)(x-x_2)$
 $a(bc) = (ab)c$
 $a+b = b+a$
 $a(b+c) = ab+ac$
 $126 = 6xy$
 $2x + 2y = 20$
 $a_n = \frac{1}{2^{n-1}} = \frac{1}{2^{10-1}}$
 $= \frac{1}{2^9} = \frac{1}{512}$
 $v = \frac{1}{4} \pi r^2 h$
 $A = \pi r^2 h$
 $\cos(B) = \frac{y}{x}$
 $\cos(60^\circ) = \frac{y}{8}$
 $\frac{1}{2} = \frac{y}{8}$
 $y = 4$
 $y = ax + b$
 $\cos(60^\circ) = \frac{y}{8}$
 $\frac{1}{2} = \frac{y}{8}$
 $y = 4$
 $AB + BC = x + y$
 $|a| = |-a|$
 $|a| \geq 0$
 $|ab| = |a||b|$
 $|a+b| \leq |a| + |b|$

How are exponents used in real-world situations?

Learning Maths for Kids:
How to garner their attention!

Software Metrics in Software Engineering



SOFTWARE METRICS IN SOFTWARE ENGINEERING

Nur Ida Aniza Rusli

Pengajian Sains Pengkomputeran, Kolej Pengajian Pengkomputeran, Informatik dan Media, Universiti Teknologi MARA, Cawangan Negeri Sembilan, Kampus Kuala Pilah, 72000 Negeri Sembilan Darul Khusus, Malaysia.
idaaniza@uitm.edu.my

Software metrics are defined as numerical measurements that are used to quantify the properties and characteristics of software systems. These metrics are essential in software engineering as they help in assessing software performance, monitoring software productivity, predicting development costs, sizing the development team, and many more. Having a knowledge of software metrics and their utilization can facilitate effective management in the software development process.

Generally, there are three categories of software metrics in software engineering, as shown in Figure 1. Product metrics refer to the measurement of various characteristics of software products, such as software size, complexity, design features, performance, and quality. For example, the measurement of software size can be done through programming code by using lines of code (LOC) or function points (FP) methods.

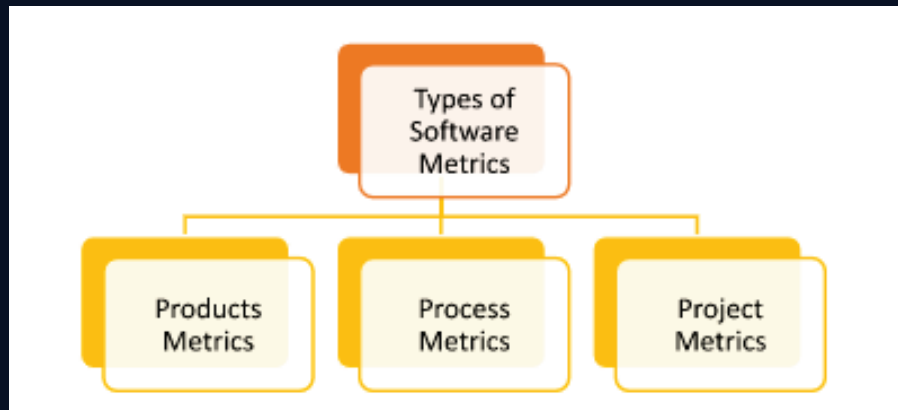


Figure 1: Categories of Software Metrics in Software Engineering

On the other hand, process metrics concentrate on improving the software development and maintenance processes. These metrics encompass various aspects, including the estimation of the duration taken to fix software bugs and the time required to implement new features. Software project metrics refer to the specific measurement [1] that are centered on the execution of software development projects. These metrics are important for project managers and software developers to accurately estimate the software cost and development schedule, track progress, and identify opportunities for improvement.

Software metrics offer various advantages [1]:

- 1.They facilitate the evaluation and measurement of software performance and quality, enabling the identification of areas in need of improvement and ensuring compliance with the required standards.
2. Software metrics aid in the planning of tasks by measuring the progress and productivity of software development. They assist in setting the goals and estimating project completion dates.
3. Development costs can also be predicted using software metrics based on the size, complexity, and quality of the software product.
4. Software metrics can help stakeholders talk to each other better by giving them a common language and understanding of the progress and quality of software development.

Despite their usefulness, software metrics have several limitations [2]:

1. The measurement can be inaccurate if the data collected is not representative of the entire system or if the wrong metrics are used.
2. Collecting information and analyzing metrics can be a time-consuming process, especially if the metrics are complex or require manual data entry.
3. The implementation of metrics can be expensive as it requires resources and tools to collect, analyze, and report the data.
4. Software metrics can be misinterpreted, leading to incorrect conclusions.

In conclusion, software metrics are important in software engineering. They provide objective and quantifiable data that can be utilized to measure and evaluate the software projects. Software developers are able to identify possible issues early in the development process and make informed decisions to improve the performance and quality of software. Software metrics practices should be aware that while software metrics can provide valuable insights, they have limitations and must be used with caution. To prevent making inaccurate estimations and decisions, it is crucial to understand the metrics being used. Overall, software metrics are an important method in software engineering that can help improve the software development process and quality of software products.

References:

- [1] Dutta, B. (2009). Enterprise Software Metrics: How to Add Business Value (Doctoral dissertation, Kent State University).
- [2] Singh, G., Singh, D., & Singh, V. (2011). A study of software metrics. IJCEM International Journal of Computational Engineering & Management, 11(2011), 22-27.