

# Aprenda Java e Processo de software

*Baseado no Personal Software Process (PSP) para Engenheiros - Parte 1*

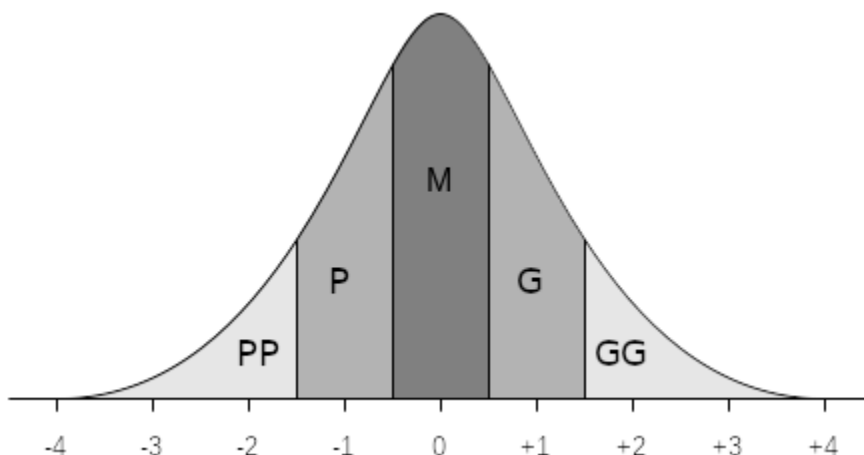
## Programa 4

### 1. Requisitos do software

No método de estimativa PROBE do PSP, tabelas de tamanho relativo são utilizadas para ajudá-lo a julgar o tamanho de novas partes dos produtos em planejamento. Essas tabelas podem ser criadas a partir do tamanho das partes de softwares anteriormente desenvolvidas, utilizando-se o desvio padrão para calcular as faixas da valores considerando uma distribuição normal.

Por exemplo, a faixa de valores médio (M) é a área que fica de -0,5 desvio padrão até +0,5 desvio padrão em relação à media, conforme apresentado na Figura 1. Assumindo que os dados seguem uma distribuição normal, a chance para que as partes estejam a mais ou menos 0,5 desvio padrão da média é de 38,3%. Seguindo lógica similar, as porcentagens para cada faixa são:

- 6,68 % para muito pequeno (PP),
- 24,17% para pequeno (P)
- 38,2% para médio (M),
- 24,17% para grande (G),
- 6,68% para muito grande (GG).



**Figura 1.** Faixas de desvios padrão.

O método de estimativa PROBE divide os dados históricos em categorias que representam o tipo de trabalho que você faz. Primeiro, divida seu conjunto de dados históricos em categorias funcionais de modo que cada uma possua pelo menos de 6 a 8 elementos (por exemplo, nas categorias cálculo, texto e dados). Para cada categoria, você pode calcular as faixas de tamanho relativos PP, P, M, G e GG utilizando o seguinte procedimento.

1. Divida os tamanhos das partes pela quantidade de elementos daquela parte, determinado o tamanho por item. Caso não tenha uma quantidade de elementos razoável, você pode considerar a divisão dos elementos em elementos menores. Por exemplo, ao invés de considerar LOC por classe implementada, você pode utilizar LOC por método (você provavelmente terá mais métodos do que classes).
2. Transforme seus lados para uma distribuição log-normal. Isto é necessário porque você não pode ter tamanhos negativos e, além disso, as faixas mais baixas tendem a ter mais instâncias que as demais. A transformação para log-normal permite que seus dados sejam calculados e apresentados a partir de uma média de valor zero. Para cada valor  $x_i$  de tamanho, calcule o logaritmo natural  $\ln$ , obtendo então  $\ln(x_i)$ .

$$avg = \frac{\sum_{i=1}^n \ln(x_i)}{n}$$

3. Calcule a média dos  $n$  valores logaritmos:

$$var = \sigma^2 = \frac{\sum_{i=1}^n (\ln(x_i) - avg)^2}{(n - 1)}$$

4. Calcule a variância desses valores:

5. Calcule o desvio padrão:  $\sigma = \sqrt{var}$

6. Calcule as faixas a partir da média e do desvio padrão logaritmos:

- $\ln(PP) = avg - 2\sigma$
- $\ln(P) = avg - \sigma$
- $\ln(M) = avg$
- $\ln(G) = avg + \sigma$
- $\ln(GG) = avg + 2\sigma$

7. Finalmente, converta os valores em logaritmo natural de volta para sua forma original, calculando  $e$  para a potência do valor logaritmo e, dessa forma, obtendo os valores dos limites de cada faixa de tamanho:

- $PP = e^{\ln(PP)}$
- $P = e^{\ln(P)}$
- $M = e^{\ln(M)}$
- $G = e^{\ln(G)}$
- $GG = e^{\ln(GG)}$

Uma vez explicado o método para calcular as faixas de valores considerando um conjunto de dados numéricos, construa um programa para calcular faixas de tamanho relativos para as faixas muito pequeno, pequeno, médio, grande e muito grande, considerando para isso o desvio padrão. Teste adequadamente o programa, usando pelo menos os dados fornecidos nas Tabelas 1 e 2, cujos resultados esperados são fornecidos na Tabela 3.

Nome da parte	Tamanho da parte em LOC	Quantidade de métodos
each_char	18	3
string_read	18	3
single_character	25	3
each_line	31	3
single_char	37	3
string_builder	82	5
string_manager	82	4

Nome da parte	Tamanho da parte em LOC	Quantidade de métodos
list_clump	87	4
list_clip	89	4
string_decrementer	230	10
Char	85	3
Character	87	3
Converter	558	10

**Tabela 2.** Dados referentes ao tamanho em LOC e quantidade de métodos das partes.

Capítulo	Quantidade de páginas
Preface	7
Chapter 1	12
Chapter 2	10
Chapter 3	12
Chapter 4	10
Chapter 5	12
Chapter 6	12
Chapter 7	12
Chapter 8	12
Chapter 9	8
Appendix A	8
Appendix B	8
Appendix C	20
Appendix D	14
Appendix E	18
Appendix F	12

**Tabela 2.** Dados referentes ao tamanho em páginas por capítulos de um livro.

Relação entre tamanho e elemento	Muito pequeno (PP)	Pequeno (P)	Médio (M)	Grande (G)	Muito grande (GG)
LOC/Método	4,3953	8,5081	16,4696	31,8811	61,7137
Páginas/Capítulo	6,3375	8,4393	11,2381	14,9650	19,9280

**Tabela 3.** Resultados esperados considerando os dados das Tabelas 1 e 2.

## 2. Processo de software: Visão geral

### 2.1. Propósito

Guiar o desenvolvimento de programas modulares.

### 2.2. Critérios de entrada

- Descrição do problema.
- Formulário de Resumo de Planejamento de Projeto.
- Registro de Tempo.
- Registro de Erros.
- Padrão de estilo de codificação.

## **2.3. Atividades**

- **Planejar**
  - Produzir ou obter requisitos.
  - Estimar o tempo necessário para desenvolvimento.
  - Preencher os dados de planejamento no formulário de Resumo de Planejamento de Projeto, incluindo o tamanho previsto para as partes do software.
  - Completar o Registro de Tempo.
- **Desenvolver**
  - Projetar o programa.
  - Implementar o projeto.
  - Compilar o programa, consertar e registrar todos os erros encontrados.
  - Testar o programa, usando testes automatizados e critérios de teste, consertar e registrar todos os erros encontrados.
  - Completar o Registro de Tempo.
- **Encerrar**
  - Completar o formulário de Resumo de Planejamento de Projeto com os dados de tempo efetivo, erros e tamanho, incluindo o tamanho efetivo das partes do software.

## **2.4. Critérios de saída**

- Programa rigorosamente testado, considerando a satisfação de critérios de teste referente a comandos e desvios.
- Formulário de Resumo de Planejamento de Projeto preenchido com dados estimados e efetivos.
- Registro de Tempo completo.
- Registro de Erros completo.

# **3. Processo de software: Planejar**

## **3.1. Propósito**

Guiar o processo de planejamento do PSP.

## **3.2. Critérios de entrada**

- Descrição do problema.
- Formulário de Resumo de Planejamento de Projeto.
- Registro de Tempo.

## **3.3. Atividades**

- **Analisar requisitos**

- Produzir ou obter os requisitos para o programa.
- Garantir que os requisitos estão claros e sem ambiguidade.
- Resolver qualquer questão quanto aos requisitos.
- **Definir partes do software**
  - Desenhar o software, organizando-o em partes para fins de planejamento e análise.
  - Preencher os dados de tamanho das partes projetadas.
- **Estimar recursos**
  - Fazer sua melhor estimativa do tempo necessário para desenvolver o programa.
  - Preencher os dados de tempo estimado/planejado no formulário de Resumo de Planejamento de Projeto.
  - Usando como guia os dados totais de tempo consumido por fase do projeto (To Date %), distribuir o tempo total de desenvolvimento para cada fase do projeto do programa atual.

### 3.4. Critérios de saída

- Requisitos documentados.
- Formulário de Resumo de Planejamento de Projeto preenchido com as estimativas de tempo de desenvolvimento e o tamanho previsto para as partes do software.
- Registro de Tempo completo.
- Registro de Erros completo.

## 4. Processo de software: Desenvolver

### 4.1. Propósito

Guiar o desenvolvimento de programas pequenos.

### 4.2. Critérios de entrada

- Requisitos.
- Registro de Tempo.
- Registro de Erros.
- Padrão de estilo de codificação.

### 4.3. Atividades

- **Desenhar / Projetar**
  - Revisar os requisitos e produzir um projeto para atendê-los.
  - Registrar no registro de erros qualquer erro encontrado nos requisitos.
  - Registrar o tempo no Registro de Tempo.
- **Programar**
  - Implementar o projeto, obedecendo o padrão de estilo de codificação.
  - Registrar no Registro de Erros qualquer erro encontrado nos requisitos e no projeto.
  - Registrar o tempo no Registro de Tempo.
- **Compilar**
  - Compilar o programa até que não existam mais erros de compilação.
  - Corrigir todos os erros encontrados.

- Registrar erros no Registro de Erros.
- Registrar o tempo no Registro de Tempo.
- **Testar**
  - Criar casos de teste, considerando critérios de teste de software.
  - Testar o programa até que os casos de teste executem sem erros e satisfaça os critérios de teste.
  - Corrigir todos os erros encontrados.
  - Registrar erros no Registro de Erros.
  - Registrar o tempo no Registro de Tempo.

#### **4.4. Critérios de saída**

- Programa rigorosamente testado, considerando a satisfação de critérios de teste referente a comandos e desvios, e implementado de acordo com o padrão de estilo de codificação.
- Registro de Tempo completo.
- Registro de Erros completo.

### **5. Processo de software: Encerrar**

#### **5.1. Propósito**

Guiar o processo de encerramento do PSP.

#### **5.2. Critérios de entrada**

- Descrição do problema e requisitos.
- Formulário de Resumo de Planejamento de Projeto com dados de tempo de desenvolvimento do programa.
- Registro de Tempo.
- Registro de Erros.
- Programa testado e executável que está em conformidade com o padrão de estilo de codificação.

#### **5.3. Atividades**

- **Garantir a consistência de dados de erros**
  - Verificar que todos os erros encontrados em cada fase foram realmente registrados.
  - Verificar se os dados de cada erro no Registro de Erros estão corretos e completos.
  - Verificar se a quantidade de erros inseridos e removidos em cada fase é razoável e correta.
  - Usando o melhor possível sua memória, corrigir qualquer dado omissos ou incorreto sobre os erros.
- **Garantir a consistência de dados de tempo**
  - Revisar o Registro de Tempo, procurando por erros e omissões.
  - Usando o melhor possível sua memória, corrigir qualquer dado omissos ou incorreto sobre os registros de tempo.
- **Medir tamanho do programa**

- Medir o tamanho do programa completo e de suas partes.
- Determinar o tamanho do código base, apagado, adicionado, reutilizado (que não seja o código base) e total.
- Preencher os dados de tamanho no Formulário de Resumo de Planejamento de Projeto.

#### **5.4. Critérios de saída**

- Programa rigorosamente testado, considerando a satisfação de critérios de teste referente a comandos e desvios, e implementado de acordo com o padrão de estilo de codificação.
- Formulário de Resumo de Planejamento de Projeto completo. incluindo o tamanho efetivo das partes do software.
- Registro de Tempo completo.
- Registro de Erros completo.

### **6. Entrega**

Quando você completar a fase de encerramento, envie os dados do pacote da tarefa, código fonte e resultados de teste para o professor.

O pacote de tarefa deve conter os seguintes a seguir, na ordem apresentada:

- Formulário de Resumo de Planejamento de Projeto,
- Registro de Tempo,
- Registro de Erros,
- Listagem do código fonte do Programa 4,
- Resultados dos testes.