

# Tutorial: Uso do PSP0

Personal Software Process<sup>SM</sup>  
para Engenheiros: Parte 1

Software Engineering Institute  
Carnegie Mellon University  
Pittsburgh, PA 15213

Copyright 2018 Carnegie Mellon University. All rights reserved.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

NO WARRANTY. THIS MATERIAL IS FURNISHED ON AN “AS-IS” BASIS WITH NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, ANY WARRANTY WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT, OR THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

[Distribution Statement A] This material has been approved for public release and unlimited distribution. The United States Government has Unlimited Rights in this material as defined by DFARS 252.227-7013.

The text and illustrations in this material are licensed by Carnegie Mellon University under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

The Creative Commons license does not extend to logos, trade marks, or service marks of Carnegie Mellon University.



# Antes de você começar o curso

PSP não é o melhor processo para desenvolver software.

PSP é um processo para aprender sobre processos de software.

PSP é o equivalente em processo de software para um exemplo de código em codificação.

Ele poderá funcionar muito bem para muitos estudantes que queiram utilizá-lo no cotidiano do trabalho, mas este não é o propósito!

Após você completar o curso, você conseguirá:

- examinar seus dados de PSP,
- revisar suas experiências e os PIPs,
- ajustar o PSP para suas necessidades.

# Objetivos do tutorial

## Após este tutorial, você:

- entenderá o processo PSP0,
- saberá como usar os scripts e formulários do PSP0,
- estará preparado para usar o PSP0 no Programa 1.



# Processo PSP0

PSP0 é um processo pessoal simples e definido:

- faça um plano,
- use seus métodos atuais de projeto e desenvolvimento para criar um pequeno programa,
- registre os dados de tempo e de defeitos quanto ao programa em desenvolvimento,
- prepare um relatório resumido.

# Objetivo do PSP0

Os objetivos do PSP0 são:

- mostrar o uso de um processo definido para o desenvolvimento de pequenos programas,
- incorporar medidas básicos no processo de desenvolvimento de software,
- demandar mudanças mínimas nas suas práticas de desenvolvimento de software.

# Fases do processo PSP0

PSP0 possui seis fases.

**Plan**

Planejamento – produz um plano para o desenvolvimento do programa, considerando os requisitos do software.

**Design**

Projeto/Desenho – produzir uma especificação de projeto para o programa, considerando os requisitos do software.

**Code**

Codificação – transformar a especificação do projeto em comandos de linguagem de programação.

# Fases do processo PSP0

(continuação)

Compilação – traduzir os códigos de programação em código executável.

**Compile**

Teste – analisar se o código executável satisfaz os requisitos do software.

**Test**

Encerramento – resumir e analisar os dados do projeto.

**Postmortem**

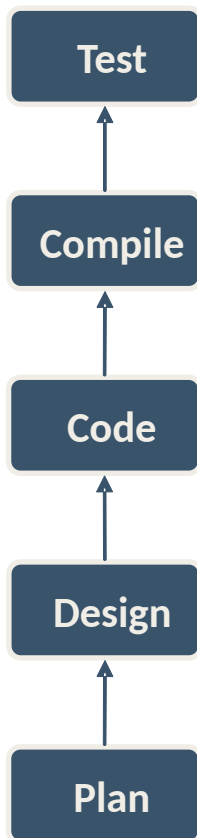


# Ordem das fases do processo

PSP parece ser um processo em cascata, mas ele não é.

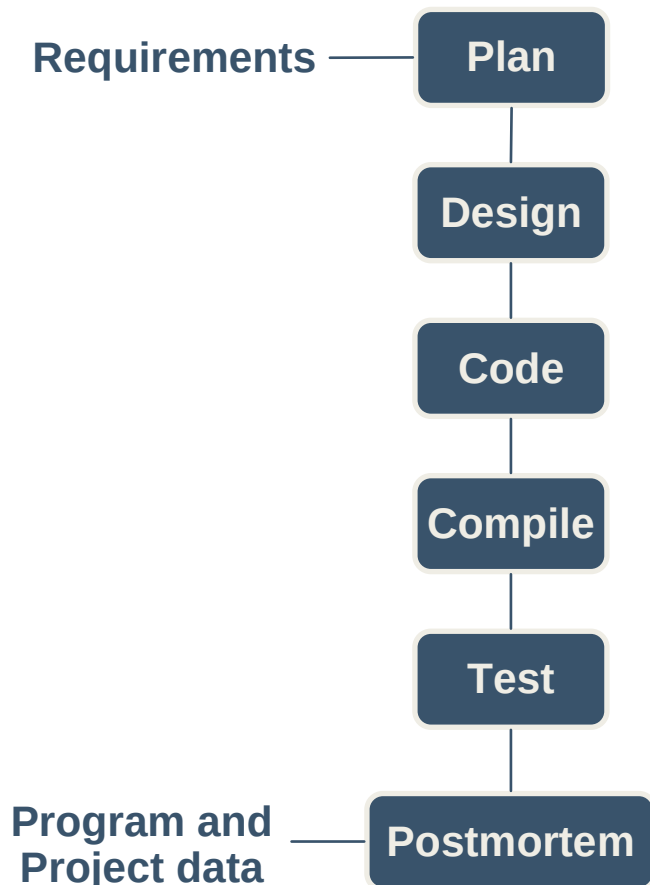
A ordem das fases é determinada pela dependência entre as fases:

- Você não pode testar o código antes dele estar compilado.
- Você não pode compilar o código antes dele estar escrito.
- Você não pode usar o projeto se ele for produzido após escrever o código.
- Não existe motivo para fazer um plano depois que você terminou.



Conclusão: você começa aqui, com um plano.

# Ordem das fases do processo



Quando os programas são pequenos ou simples, você pode executar as fases em ordem.

Produza um plano.

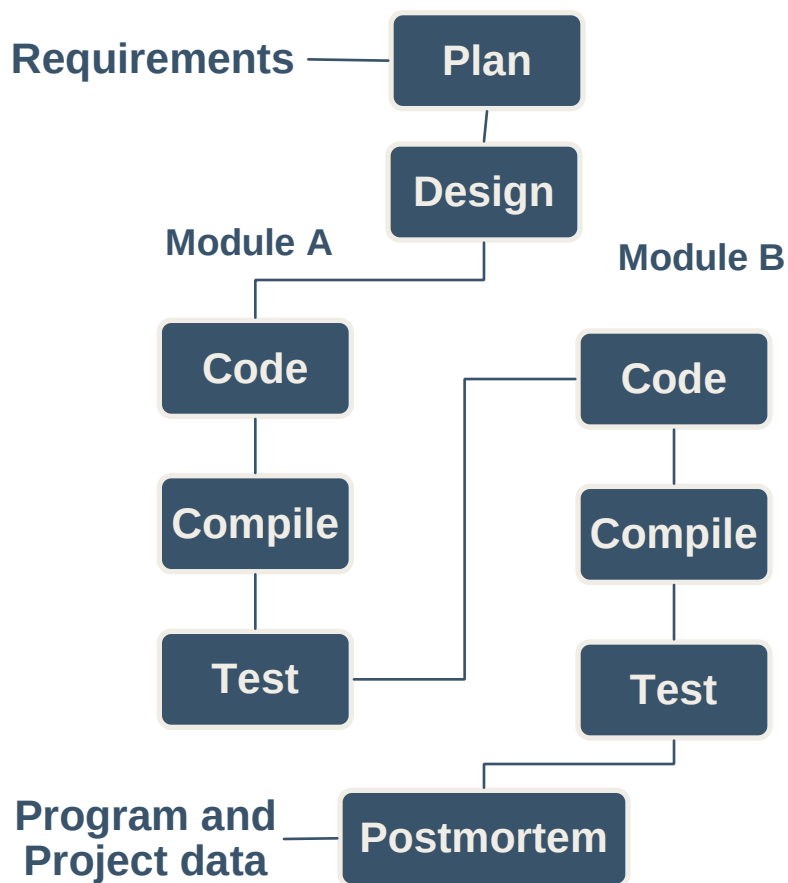
Projete todos os módulos.

Implemente todos os módulos.

Compile o programa.

Resuma os dados do projeto durante o encerramento.

# Ordem iterativa das fases do processo - 1



Programas maiores ou mais complexos podem exigir uma abordagem iterativa.

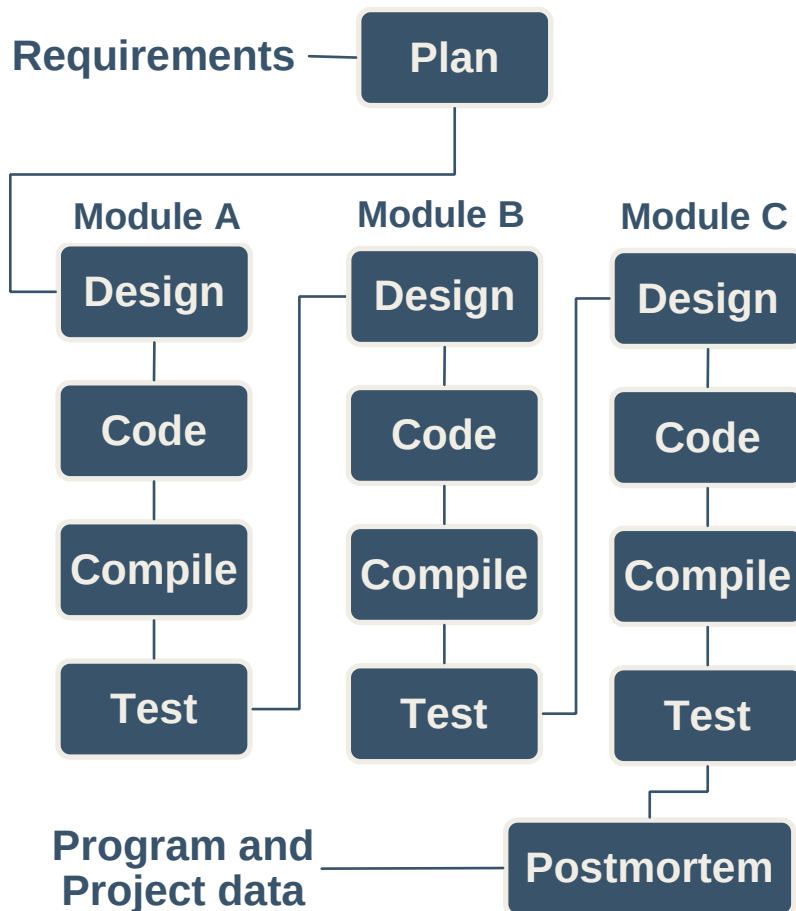
Neste exemplo, o projeto é feito em uma etapa.

Dois módulos são identificados durante o projeto: módulos A e B.

Então cada módulo é separadamente implementado, compilado e testado.

Este exemplo utiliza as fases do PSP0 e dois ciclos de implementação-compilação-teste.

# Ordem iterativa das fases do processo - 2



Também podem existir mais de dois ciclos e ciclos que incluem a fase de projeto, como neste exemplo.

Observe que cada ciclo tem como objetivo produzir uma parte funcional do programa, referente aos Módulos A, B ou C.

O tamanho de cada parte é um fator chave para determinar quantos ciclos são necessários:

- Uma linha de código é muito pouco
- Um programa pode ser muito grande

Uma ou mais classes, módulos, procedimentos ou funções provavelmente são do tamanho certo.

Você precisa determinar o que funciona para você.

# Scripts de processo

Scripts de processo fornecer um guia de nível especialista sobre como usar o processo.

Eles geralmente tem uma ou duas página.

Cada script descreve:

- Propósito
- Critérios de entrada
- Diretivas gerais
- Passos
- Critérios de saída

PSP0 Process Script		
<b>Purpose</b>	To guide the development of module-level programs	
<b>Entry Criteria</b>	<ul style="list-style-type: none"><li>- Problem description</li><li>- PSP0 Project Plan Summary form</li><li>- Time and Defect Recording logs</li><li>- Defect Type standard</li><li>- Stopwatch (optional)</li></ul>	
Step	Activities	Description
1	Planning	<ul style="list-style-type: none"><li>- Produce or obtain a requirements statement.</li><li>- Estimate the required development time.</li><li>- Enter the plan data in the Project Plan Summary form.</li><li>- Complete the Time Recording log.</li></ul>
2	Development	<ul style="list-style-type: none"><li>- Design the program</li><li>- Implement the design.</li><li>- Compile the program, and fix and log all defects found.</li><li>- Test the program, and fix and log all defects found.</li><li>- Complete the Time Recording Log.</li></ul>
3	Postmortem	Complete the Project Plan Summary form with actual time, defect, and size data.
<b>Exit Criteria</b>		<ul style="list-style-type: none"><li>- A thoroughly tested program</li><li>- Completed Project Plan Summary form with estimated and actual data</li><li>- Completed Time and Defect Recording logs</li></ul>

# Scripts PSP0 - 1

Planejamento: Estime o tempo de desenvolvimento.

Desenvolvimento: Desenvolva o produto usando os métodos disponíveis.

Encerramento: Complete o resumo sobre o planejamento do projeto com o tempo gasto e erros encontrados e inseridos em cada fase.

# Scripts PSP0 - 2

Projeto: Desenhe/projeto o programa usando métodos disponíveis de projeto.

Implementação: Implemente o programa.

Compilação: Compile o programa até que ele não tenha erros de compilação.

Teste: Teste o programa e conserte todos os erros.

Registre os erros no registro de erros e o tempo gasto em cada fase no registro de tempo.

# Uso dos scripts de processo

Os scripts de processo o guiarão durante a execução do processo.

Você deve:

- Verificar os critérios de entrada antes de iniciar uma fase;
- Registrar o tempo de início da fase;
- Executar os passos e instruções da fase;
- Registrar erros conforme eles forem encontrados e corrigidos;
- Verificar os critérios de saída antes de terminar uma fase;
- Registrar o tempo de término da fase;
- Ir para a próxima fase.

Obrigue-se a usar esse paradigma até que ele se torne um hábito.



# Medidas e formulários do PSP0

## Medidas do PSP0:

- Tempo – registra o tempo em cada fase.
- Erros – registra os erros conforme eles são encontrados e consertados.

## PSP0 possui quatro formulários:

- Resumo do plano do projeto do PSP0 – resume o tempo e quantidade de defeitos planejado e obtidos para cada fase.
- Registro de tempo do PSP0 – usado para registrar o tempo de início e término de cada fase.
- Registro de erros do PSP0 – usado para registrar erros.
- Padrão de tipos de erros do PSP0 – usado para estabelecer o tipo de cada erro encontrado.

# Alguns dos erros mais comuns

- Não registrar o tempo nas fases de planejamento e de encerramento.
- Pular fases ou executar fases fora de ordem
- Omitir alguns erros encontrados nas fases de compilação e de teste.
- Esquecer de medir o tamanho efetivo do programa desenvolvido (PSP1.x em diante).
- Confundir fases com atividades (por exemplo, contar tempo gasto com atividades de reprojeto ou reimplementação durante a fase de teste como tempo de projeto ou de implementação).

# Dicas para medição

Obtenha e registre os dados conforme você trabalha, não depois. Caso se esqueça, registre o quanto antes com o valor mais aproximado que você puder.

Seja preciso:

- Registre o tempo em minutos.
- Contabilize todo e qualquer erro.

# Mensagem final

No PSP0, o principal objetivo é aprender a registrar com precisão e de forma completa os dados de seu trabalho.

Uma vez que você tenha aprendido o PSP inteiro, você poderá ajustar e estender o PSP para atender suas necessidades.

Até lá, faça o seu melhor esforço para seguir os scripts e instruções do processo PSP.