



Programação procedimental

Programação procedural ou **programação procedimental** é um paradigma de programação que se baseia em procedimentos, que são executados numa sequência.^[1] Ela é derivada da programação imperativa, mas acrescentando procedimentos (em inglês, *procedure calls*), de forma a organizar o código em blocos que possam ser reutilizados.^[2]

Os procedimentos, também conhecidos como rotinas,^[3] sub-rotinas,^[4] métodos,^[5] ou funções^[4] simplesmente contêm um conjunto de passos computacionais a serem executados. Um dado procedimento pode ser chamado a qualquer hora durante a execução de um programa, inclusive por outros procedimentos ou por si mesmo.

A programação procedural é geralmente uma escolha melhor que a programação sequencial e não estruturada em muitas situações que envolvem uma complexidade média e requerem facilidade de manutenção. Possíveis benefícios são a habilidade de reutilizar o mesmo código em diferentes lugares no programa sem copiá-lo, ser uma forma mais fácil de organizar o fluxo do programa que uma coleção de comandos "goto" ou "jump" (que podem transformar um programa extenso e complexo num código espaguete), e a habilidade de ser fortemente modular e estruturado.

Procedimentos e modularidade

A modularidade é uma característica geralmente desejável, especialmente em programas grandes e complicados. Ela pode ser alcançada com a utilização de procedimentos com canais de entrada e saída estritamente definidos, usualmente acompanhados de regras claras sobre quais tipos de entrada e saída são permitidos ou esperados. As entradas costumam ser especificadas sintaticamente na forma de *argumentos*, e as saídas entregues na forma de *valores de retorno*.

O gerenciamento de escopo é outra técnica que ajuda a manter procedimentos fortemente modulares. Ela impede que o procedimento acesse variáveis de outros procedimentos (e vice-versa), incluindo instâncias anteriores de si mesmo, sem autorização explícita. Isto ajuda a impedir confusões entre variáveis com o mesmo nome sendo utilizadas em locais diferentes, e também que os procedimentos atrapalhem a execução um do outro.

Procedimentos menos modulares, frequentemente utilizados em programas pequenos ou escritos rapidamente, tendem a interagir com um grande número de variáveis no ambiente de execução, que também podem ser modificadas por outros procedimentos. O fato de que muitas variáveis agem como pontos de contato entre as várias partes do programa é o que o torna menos modular.

Por causa da habilidade de especificar uma interface simples, de serem auto-contidos, e de serem reutilizados, procedimentos facilitam a criação de programas ou bibliotecas de programação por várias pessoas ou grupos diferentes.

Comparação com a programação imperativa

A maioria das linguagens procedurais também são linguagens imperativas,^{[[carece de fontes?](#)]} pois fazem referências explícitas ao estado do ambiente de execução. Isto pode significar desde variáveis (que podem corresponder aos registradores do processador) a algo como a posição da "tartaruga" na linguagem de programação Logo (que por sua vez pode ser desde um cursor na tela a um dispositivo físico que se move no chão de uma sala).

Algumas formas de programação imperativa, como a programação orientada a objetos não são necessariamente procedurais.

Linguagens de programação procedural

Para ser considerada procedural, uma linguagem de programação deve suportar o conceito de procedimentos, e possuir uma sintaxe para defini-los. Idealmente, ela deve suportar a especificação de tipos de argumentos, variáveis locais, chamadas recursivas e o uso de procedimentos em módulos distintos de um programa. Ela também pode suportar a distinção entre argumentos de entrada e de saída.

O exemplo canônico de uma linguagem de programação procedural é ALGOL. Uma linguagem em que a única forma de procedimento é um método é geralmente considerada orientada a objetos ao invés de procedural, e não será incluída nesta lista. Isto se aplica a C# e Java, mas não a C++.

- | | | |
|--|-----------------------------|-----------------------|
| ▪ <u>Ada</u> | ▪ <u>Forth</u> | <u>Oberon-2)</u> |
| ▪ <u>ALGOL</u> | ▪ <u>Fortran</u> | ▪ <u>occam</u> |
| ▪ <u>BASIC</u> | ▪ <u>F</u> | ▪ <u>M</u> |
| ▪ <u>C</u> | ▪ <u>Go (Golang)</u> | ▪ <u>Pascal</u> |
| ▪ <u>ColdFusion</u> | ▪ <u>Lasso</u> | ▪ <u>Perl</u> |
| ▪ <u>COBOL</u> | ▪ <u>Linoleum</u> | ▪ <u>PHP</u> |
| ▪ <u>Component Pascal</u> | ▪ <u>Lua</u> | ▪ <u>PL/C</u> |
| ▪ <u>D</u> | ▪ <u>Maple</u> | ▪ <u>Python</u> |
| ▪ <u>Delphi</u> | ▪ <u>Mathematica</u> | ▪ <u>PL/I</u> |
| ▪ <u>ECMAScript</u> (e.g.,
<u>ActionScript</u> , <u>DMDScript</u> ,
<u>JavaScript</u> , <u>JScript</u>) | ▪ <u>MATLAB</u> | ▪ <u>Rapira</u> |
| | ▪ <u>Modula-2</u> | ▪ <u>VBScript</u> |
| | ▪ <u>Oberon</u> (Oberon-1 e | ▪ <u>Visual Basic</u> |

Ver também

- Módulo (programação)
- Linguagem de programação
- Paradigma de programação
- Programação estruturada
- Programação funcional
- Programação imperativa

- Programação orientada a objetos

Referências

1. «Procedural programming - Contemporary trends in software development - CCEA - GCSE Digital Technology (CCEA) Revision» (<https://www.bbc.co.uk/bitesize/guides/zc8pjty/revisio>
[n/2](https://www.bbc.co.uk/bitesize/guides/zc8pjty/revisio)). *BBC Bitesize* (em inglês). Consultado em 14 de dezembro de 2022
2. Mueller, John Paul (29 de outubro de 2019). *Programação Funcional Para Leigos* (https://books.google.com/books?id=bJ_BDwAAQBAJ&newbks=0&printsec=frontcover&pg=PA13&dq=Programa%C3%A7%C3%A3o+procedural&hl=pt-BR). [S.l.]: Alta Books. p. 13. ISBN 9788550813509
3. Rohit, Khurana. *Object Oriented Programming with C++, 2nd Edition* (<https://books.google.com/books?id=zJRDDAAQBAJ&newbks=0&printsec=frontcover&pg=PA3&dq=procedural+programming&hl=pt-BR>) (em inglês). [S.l.]: Vikas Publishing House. p. 2. ISBN 9789325975644
4. «Procedural programming» (https://isaacomputerscience.org/topics/procedural_programming?examBoard=all&stage=all). *Isaac Computer Science* (em inglês). Consultado em 14 de dezembro de 2022
5. Rai, Laxmisha (20 de maio de 2019). *Programming in C++: Object Oriented Features* (<https://books.google.com/books?id=o3acDwAAQBAJ&newbks=0&printsec=frontcover&pg=PA8&dq=procedural+programming&hl=pt-BR>) (em inglês). [S.l.]: Walter de Gruyter GmbH & Co KG. ISBN 9783110593846

Ligações externas

Obtida de "https://pt.wikipedia.org/w/index.php?title=Programação_procedimental&oldid=69786712"