

Cap. 1.1 - Conceitos introdutórios de programação

Aprenda a Programar com C# 2018 — Edições Sílabo
<http://www.silabo.pt/livros.asp?num=606>

António Trigo, Jorge Henriques
{antonio.trigo,jmvhenriques}@gmail.com

17 de Abril de 2018

Introdução

Fases

Linguagens de programação

Comunicação Homem-Computador

Desenvolvimento de programas

Introdução

- ▶ Existem dois componentes distintos no computador:
 - ▶ Hardware, componentes físicos que constituem o computador;
 - ▶ Software, vários programas de computador e divide-se em:
 - ▶ software de sistema, responsável pela interação homem computador;
 - ▶ software de aplicação, que permite executar tarefas específicas.
- ▶ Um programa de computador, é composto por um conjunto de instruções que descrevem uma tarefa a ser realizada pelo computador;
- ▶ E programar?

Programar

- ▶ Mais do que escrever código numa linguagem de programação, como o C, programar engloba um conjunto de atividades que têm por finalidade a obtenção de software, designadamente:
 - ▶ Análise;
 - ▶ Conceção;
 - ▶ Implementação;
 - ▶ Validação;
 - ▶ Manutenção.

Ciclo de desenvolvimento de software - modelo cascata

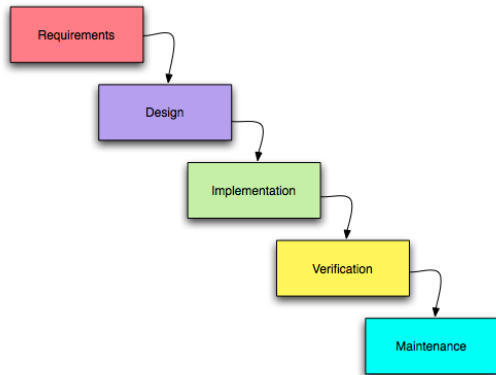


Figura: Modelo em cascata

Definição

- ▶ É um conjunto limitado de instruções (vocabulário), associado a um conjunto de regras (sintaxe) que define como as instruções podem ser associadas, ou seja, como se podem compor os programas para a resolução de um determinado problema;
- ▶ É um método normalizado para expressar instruções para um computador;
- ▶ É um conjunto de regras sintáticas e semânticas utilizadas para definir um programa de computador;
- ▶ Permite que um programador especifique com precisão sobre que dados irá atuar e como estes serão armazenados/transmitidos e que ações tomar sobre eles.

Níveis de linguagens

- ▶ As linguagens de programação podem ser classificadas em níveis de linguagens, sendo que as linguagens de nível mais baixo estão mais próximas da linguagem interpretada pelo processador e mais distante das linguagens naturais;
- ▶ Linguagem máquina, composta por conjuntos de bits;
- ▶ Linguagem hexadecimal, mais próxima do ser humano;
- ▶ Linguagem Assembly, versão legível da linguagem máquina;

Níveis de linguagens (cont.)

- ▶ Linguagem Alto Nível
 - ▶ Apresenta uma sintaxe mais próxima da linguagem natural, fazendo uso de palavras reservadas extraídas do vocabulário corrente (como READ, WRITE, TYPE, etc.);
 - ▶ Permite a manipulação dos dados nas mais diversas formas (números inteiros, reais, vetores, etc.), enquanto que a linguagem Assembly trabalha com bits, bytes, palavras, armazenadas em memória;
 - ▶ Teoricamente, os programas escritos nestas linguagens podem ser utilizados em diferentes máquinas.
- ▶ Linguagem Estruturada, Pascal, C;
- ▶ Linguagem Orientadas a Objetos, **C#**, Java e C++.

Interpretação/Compilação

- ▶ Uma linguagem de programação pode ser convertida, ou traduzida, em código máquina por dois métodos:
 - ▶ Compilação ,em que o texto é todo traduzido de uma só vez para um programa executável, caso do Pascal e do C;
 - ▶ Interpretação, em que o texto vai sendo traduzido à medida que é executado, como é o caso, do JavaScript, Python ou Perl.

Ergonomia de software

- ▶ Ciência que estuda conforto, utilização, organização e documentação do software.
- ▶ Tem por objetivo facilitar e otimizar o trabalho do utilizador com o computador.
- ▶ Propõe normas para:
 - ▶ Apresentação de ecrãs;
 - ▶ Caixas de diálogos;
 - ▶ Ferramentas de trabalho;
 - ▶ Menus;
 - ▶ Documentação;
 - ▶ Técnicas de ficheiros;
 - ▶ Técnicas de otimização;
 - ▶ Ajuda.

Princípios da ergonomia

1. Minimizar o esforço do utilizador (ex.: eliminar trabalho duplicado)
2. Minimizar a utilização de memória do utilizador (ex.: o utilizador deve memorizar o mínimo possível);
3. Minimizar a frustração do utilizador (ex.: evitar demora nas tarefas, acesso direto às tarefas, etc.);
4. Maximizar o uso de padrões e hábitos (ex.: utilizar as mesmas teclas para as mesmas funções);
5. Maximizar a tolerância a diferenças humanas (ex.: criação de perfis por utilizador, acessibilidade, utilização de botões, etc.);

Princípios da ergonomia (cont.)

6. Maximizar o suporte a mudanças de ambiente de hardware/software (ex.: suportar a utilização em diferentes sistemas operativos, reconfiguração do software, suportar diferentes bases de dados, etc.);
7. Notificação imediata de problemas, o programa deve notificar o utilizador de erros que ocorrem no programa (ex.: ficheiro não existente), bem como de alterações importantes aos dados (ex.: operação de eliminação de registos)
8. Maximizar o controlo das tarefas pelo utilizador (ex.: o utilizador deve poder decidir a forma como realiza as tarefas, definir prioridades de processamento, poder definir as operações mais comuns, etc.)
9. Maximizar o suporte às tarefas (ex.: todos os recursos devem estar disponíveis para a realização de uma determinada tarefa, existência de documentação do sistema, designadamente de manuais de utilização, etc.)

Desenvolvimento de programas

- ▶ O desenvolvimento de programas está associado ao uso de ferramentas ou ambiente de desenvolvimento que acompanham o programador desde a etapa de codificação até à geração e teste do código executável.
 - ▶ Criação do código fonte (codificação);
 - ▶ Tradução do código fonte;
 - ▶ Debuggers, auxiliam o programador a eliminar (ou a reduzir) a quantidade de erros (bugs) de execução do programa;
 - ▶ Execução passo-a-passo;
 - ▶ Execução de programas;
 - ▶ Visualização dos valores das variáveis;
 - ▶ Alteração em tempo de execução de conteúdos de memória, de variáveis ou de instruções, etc.