



Programação de Computadores e seus conceitos básicos

Materiais de Formação

Subcapítulo 1.4. – Como funciona o programa

WP3: Materiais de Formação Code4SP

Preparado por:



CITIZENS
IN POWER



Center for Social
Innovation



ZAUG
gGmbH



Subcapítulo 1.4. : Como funciona um programa



Como funciona um programa

Como funciona um programa

Existem muitos tipos diferentes de programas de computador, mas todos eles têm os mesmos componentes básicos: uma interface de utilizador, um processador e memória. A interface de utilizador permite que este introduza informações e instruções no programa, o processador executa as instruções, e a memória armazena o programa e os dados que este último processa. A maioria dos programas de computador são escritos numa linguagem de programação de alto nível, uma linguagem concebida para ser de fácil leitura e escrita pelo ser humano. Contudo, o processador só pode compreender o código da máquina, que é uma série de uns e zeros. Portanto, antes de um programa poder ser executado, deve ser convertido em código de máquina. Ou seja, terá de ser feito por um programa chamado *compilador*. O compilador lê o programa e converte-o em código de máquina. Depois armazena o código da máquina num ficheiro chamado *executável*. Quando o utilizador executa o programa, o executável é carregado na memória e o processador executa as instruções.



Como funciona um programa

O ciclo *fetch-decode-execute*

O ciclo *fetch-decode-execute* é o processo básico que um computador utiliza para executar instruções. O ciclo começa quando o computador vai buscar uma instrução à memória. Em seguida, descodifica a instrução para determinar o que é suposto fazer. Finalmente, executa a instrução. O ciclo repete-se então, recuperando a instrução seguinte da memória.



Como funciona um programa

Da linguagem de máquina à linguagem de montagem

Como a programação em linguagem de máquina, que consiste apenas em código binário, é demasiado complexa para um ser humano, foi criada uma linguagem de montagem. A linguagem de montagem (ou assembly) trata-se de uma linguagem de programação de baixo nível para um computador, microprocessador, ou outro dispositivo programável, na qual o programador utiliza as instruções da linguagem assembly para controlar o funcionamento do dispositivo. A linguagem de montagem é específica para um determinado microprocessador ou família de microprocessadores. Consiste numa série de códigos mnemónicos, nomes simbólicos para as operações que o microprocessador pode realizar, e os operandos (dados) sobre os quais estas operações devem ser realizadas. A linguagem de montagem é convertida em código de máquina, uma forma de código binário específico de um determinado tipo de computador e que pode ser compreendido pelo processador do computador. O código da máquina é a única forma de código que o processador pode executar diretamente.

Mesmo sendo mais fácil do que a programação em linguagem de máquina, a programação em linguagem de montagem não é suficientemente útil para produzir código fonte rápido e fácil de ler. Por conseguinte, as linguagens de programação de alto nível (tais como C# ou python) foram criadas.

Atualmente, existem muitas linguagens de programação de alto nível. Algumas das mais comuns são Java, Python e Ruby. As linguagens de programação de alto nível são mais fáceis de utilizar do que as linguagens de programação de baixo nível. Permitem que se concentre na tarefa em mãos, em vez de se concentrar nos detalhes do computador. Isto torna-as ideais para a criação de aplicações e programas. As linguagens de programação de alto nível também tendem a ser mais tolerantes do que as linguagens de programação de baixo nível. Se cometer um erro ao escrever código numa linguagem de alto nível, o compilador será normalmente capaz de o corrigir por si. Isto pode poupar-lhe muito tempo e frustração ao “bater código”





Como funciona um programa

Palavras-chave, Operadores, e Sintaxe: uma visão geral

Há muitas linguagens de programação de alto nível disponíveis hoje em dia. Cada uma tem o seu conjunto único de palavras-chave, operadores e sintaxe. Para ser eficaz com uma linguagem de programação de alto nível, é importante estar familiarizado com as palavras-chave, operadores, e sintaxe específicos utilizados por essa linguagem. Algumas das palavras-chave mais comuns utilizadas nas linguagens de programação de alto nível incluem: if, then, else, while, for, do, break, continue. Estas palavras-chave são utilizadas para controlar o fluxo da execução do programa. Os operadores são símbolos que representam operações que podem ser realizadas sobre valores. Os operadores mais comuns incluem: + (addition), - (subtraction), * (multiplication), / (division), e % (modulus). Estes operadores podem ser utilizados para calcular os resultados das expressões. A sintaxe de uma linguagem de programação é o conjunto de regras que dirigem a forma como o código deve ser escrito para ser interpretado pelo compilador ou pelo intérprete. A sintaxe de uma linguagem de programação de alto nível é tipicamente mais tolerante do que a sintaxe de uma linguagem de nível inferior. Isto pode facilitar aos principiantes a aprendizagem da programação.





Como funciona um programa

Compiladores e Intérpretes

Os compiladores e intérpretes de computador são ferramentas importantes para os programadores de software. Um compilador pega no código escrito numa língua e converte-o em código que pode ser executado numa máquina diferente. Um intérprete pega no código escrito numa língua e executa-o tal como está, sem o compilar primeiro. Os compiladores são tipicamente utilizados para linguagens que têm muita estrutura, como C ou Java. Os intérpretes são tipicamente utilizados para linguagens que são mais flexíveis, como Python ou Ruby. Os compiladores produzem normalmente códigos mais rápidos do que os intérpretes. No entanto, os intérpretes são tipicamente mais portáteis, o que significa que podem funcionar em mais tipos de máquinas. O instrumento a utilizar depende da situação. Se a velocidade é importante, um compilador é uma melhor escolha. Se a portabilidade é importante, um intérprete é uma melhor alternativa.





OBRIGADO!

PRÓXIMO CAPÍTULO: Programas e linguagens de programação

Co-funded by the
Erasmus+ Programme
of the European Union

