

Perspectiva histórica e motivação para a POO

- ❑ POO é um paradigma de programação
- ❑ Muitas linguagens e plataformas de programação modernas suportam este paradigma
 - Java, C++, .NET,
- ❑ OOP nasceu nos finais dos anos 60 quando a área da Engenharia de Software (então nascente) se defrontava com a chamada “software crisis”
 - Ver o artigo “[The humble programmer](http://www.cs.utexas.edu/users/EWD/ewd03xx/EWD340.PDF)” de Dijkstra, disponível em <http://www.cs.utexas.edu/users/EWD/ewd03xx/EWD340.PDF>

Perspectiva histórica e motivação para a POO

- ❑ Esta “software crisis” foi causada pelo desenvolvimento do hardware e dos requisitos de software, aos quais os programadores tinham dificuldade em dar resposta com as ferramentas e metodologias disponíveis
 - Era difícil escrever programas correctos, compreensíveis e verificáveis
 - Complexidade, expectativas, mudança,...

Perspectiva histórica e motivação para a POO

- ❑ OOP respondeu em parte a esses problemas, uma vez que promove fortemente a modularização dos programas
 - Um programa pode ser visto como uma colecção de unidades (ou objectos), com os seus atributos e capacidades próprias, que interactuam entre si
 - ❑ A aproximação tradicional descrevia um programa como uma lista de funções ou mesmo de instruções
 - Cada objecto é capaz de receber mensagens, processar informação e enviar mensagens a outros objectos
 - Um objecto pode ser visto como uma pequena máquina ou actor com o seu papel e responsabilidade no contexto do programa

Perspectiva histórica e motivação para a POO

- ❑ O conceito de objecto apareceu pela primeira vez numa linguagem de programação no final dos anos 60
 - A linguagem era o [Simula 67](#)
 - Criada na Noruega por Ole-Johan Dahl e Kristen Nygaard
 - Trabalhavam na simulação de navios
 - A ideia foi agrupar os navios em diversas classes de objectos, sendo cada uma delas responsável pela definição dos seus dados e comportamentos
 - Esta linguagem introduziu pela primeira vez os conceitos de classe e de instância de classe

Perspectiva histórica e motivação para a POO

- A linguagem Smalltalk apareceu nos anos 70
 - Desenvolvida no Xerox PARC
 - Introduziu o termo “**object oriented programming**” para definir a utilização extensiva de objectos e mensagens como base dos programas
 - Foi desenhada como uma linguagem completamente dinâmica em que os objectos podem ser criados, modificados e utilizados dinamicamente (e não de forma estática)
 - Incluiu ainda o conceito de **herança**

Perspectiva histórica e motivação para a POO

- A POO tornou-se o “paradigma dominante” nos anos 80
 - Aparecimento do C++
 - Importância crescente dos interfaces gráficos e a adequação da POO ao seu desenvolvimento (a interface é composta de objectos)
 - Apareceram diversos toolkits baseados em POO e programação por eventos
 - Outras linguagens foram propostas, como Modula 2, Eiffel e versões OO de linguagens já existentes (Pascal, Ada, Basic, etc.). Algumas destas procuraram manter compatibilidade com a programação procedimental

Perspectiva histórica e motivação para a POO

- Mais recentemente o Java ganhou grande popularidade
 - Utilização de uma máquina virtual -> compatibilidade
 - Semelhanças com C e C++
- Ainda mais recentemente apareceram plataformas e linguagens baseadas em POO, mas algumas procurando manter compatibilidade com metodologias procedimentais
 - Python, Ruby
 - Visual Basic.NET, C# (desenhadas para a plataforma .NET)