

# FlowShop Scheduling

Fabiano Fidêncio  
Filipe Oshiro  
Ricardo Miranda  
Vitor Massaru Iha

Instituto de Computação - UNICAMP

24 de maio de 2011

# Sumário

- 1 O Problema
- 2 É um problema de IA
- 3 Como resolver
- 4 Outras abordagens

# O Problema

O que ele é

- Problema de otimização combinatória.

# O Problema

## O que ele é

- Problema de otimização combinatória.
- Fluxo unidirecional de produtos através de unidades de produção.

# O Problema

## O que ele é

- Problema de otimização combinatória.
- Fluxo unidirecional de produtos através de unidades de produção.
- Não há preempção.

# O Problema

## O que ele é

- Problema de otimização combinatória.
- Fluxo unidirecional de produtos através de unidades de produção.
- Não há preempção.
- Produto pode pular um estágio de produção mas sempre mantendo o mesmo sentido de fluxo.

# O Problema

## O que ele é

- Problema de otimização combinatória.
- Fluxo unidirecional de produtos através de unidades de produção.
- Não há preempção.
- Produto pode pular um estágio de produção mas sempre mantendo o mesmo sentido de fluxo.
- Sequencias de Permutação.

# O Problema

## O que ele é

- Problema de otimização combinatória.
- Fluxo unidirecional de produtos através de unidades de produção.
- Não há preempção.
- Produto pode pular um estágio de produção mas sempre mantendo o mesmo sentido de fluxo.
- Sequencias de Permutação.
- Objetivo: obter a melhor sequencia de operações minimizando o tempo total de produção.

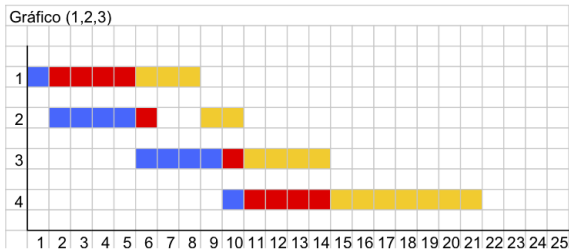


# O Problema

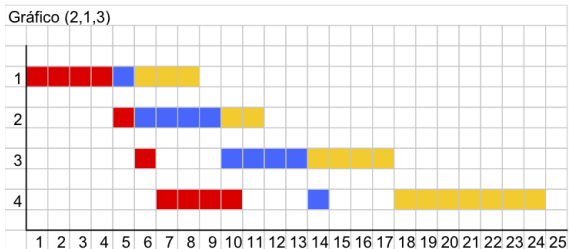
## O que ele é

- Problema de otimização combinatória.
- Fluxo unidirecional de produtos através de unidades de produção.
- Não há preempção.
- Produto pode pular um estágio de produção mas sempre mantendo o mesmo sentido de fluxo.
- Sequencias de Permutação.
- Objetivo: obter a melhor sequencia de operações minimizando o tempo total de produção.

# Exemplos

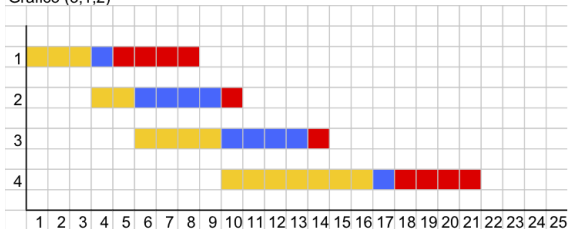


# Exemplos

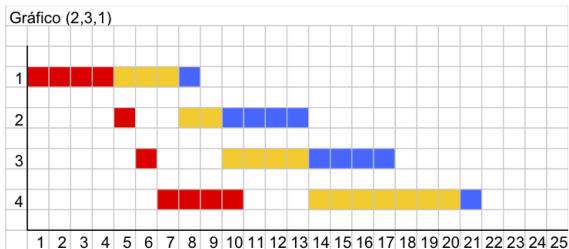


# Exemplos

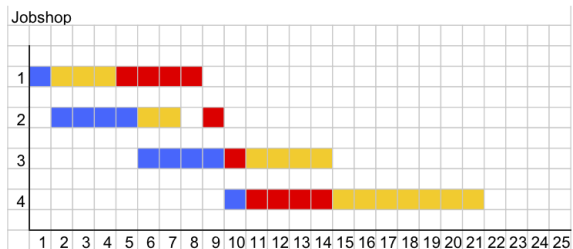
Gráfico (3,1,2)



# Exemplos



# Exemplos



# Sumário

- 1 O Problema
- 2 É um problema de IA
- 3 Como resolver
- 4 Outras abordagens

# É um problema de IA

- Encontrar a sequência que minimiza o makespan, é um problema NP-difícil



# É um problema de IA

- Encontrar a sequência que minimiza o makespan, é um problema NP-difícil
- Isto inviabiliza a procura pela solução ótima através de algoritmos exatos

# É um problema de IA

- Encontrar a sequência que minimiza o makespan, é um problema NP-difícil
- Isto inviabiliza a procura pela solução ótima através de algoritmos exatos
- Abre caminho para a abordagem do problema através de métodos aproximados que procuram encontrar soluções aceitáveis, eventualmente ótimas, em um tempo razoável

# É um problema de IA

- Encontrar a sequência que minimiza o makespan, é um problema NP-difícil
- Isto inviabiliza a procura pela solução ótima através de algoritmos exatos
- Abre caminho para a abordagem do problema através de métodos aproximados que procuram encontrar soluções aceitáveis, eventualmente ótimas, em um tempo razoável

# Sumário

- 1 O Problema
- 2 É um problema de IA
- 3 Como resolver**
- 4 Outras abordagens

# Como resolver

Utilizaremos Algoritmos Genéticos para resolução do problema, seguindo o seguinte algoritmo base:

Inicializar a população de cromossomos

Repita até atender ao critério de parada

    Avaliar cada cromossomo (função-objetivo).

    Selecionar e Aplicar Operadores (Recombinação, Mutação).

Fim-Repita Retornar o melhor cromossomo.

# Sumário

- 1 O Problema
- 2 É um problema de IA
- 3 Como resolver
- 4 Outras abordagens

# Outras abordagens

Há algumas outras formas de resolvermos o problema citado, entre elas:

- Algoritmo  $A^*$  (Branch and Bound)

# Outras abordagens

Há algumas outras formas de resolvermos o problema citado, entre elas:

- Algoritmo A\* (Branch and Bound)
- Simulated Annealing



# Outras abordagens

Há algumas outras formas de resolvermos o problema citado, entre elas:

- Algoritmo A\* (Branch and Bound)
- Simulated Annealing
- Busca Tabu

# Outras abordagens

Há algumas outras formas de resolvermos o problema citado, entre elas:

- Algoritmo A\* (Branch and Bound)
- Simulated Annealing
- Busca Tabu