

**2º Semestre Ano Letivo 2022/2023**

**Unidade Curricular de Optimização Heurística**

**Docente Anabela Costa**

**Trabalho de Grupo**

**2º ano - LCD – CDB1**

**Eliane Susso Efraim Gabriel, N° 103303**

**João Francisco Marques Gonçalves da Silva Botas, N°104782**

**Marco Delgado Esperança, N° 110451**

**Maria João Ferreira Lourenço, N° 104716**

**Umeima Adam Mahomed, N°99239**

**Lisboa, 12 de junho de 2023**

## Índice

Introdução .....	3
a) Descreva, por palavras, uma solução admissível para o problema da <i>Química_PT</i> ....	3
b) Desenvolva uma heurística para determinar uma sequência de processamento das tarefas em cada máquina que respeite a data de conclusão acordada para as tarefas T3, T4 e T5. Com base na heurística desenvolvida, apresente uma solução admissível para o problema da <i>Química_PT</i> .....	3
c) Defina um cromossoma que permita codificar uma sequência de processamento das tarefas em cada máquina.....	6
d) Tendo em conta a codificação que sugeriu em c), proponha um operador de crossover e exemplifique-o no contexto do problema da <i>Química_PT</i> .....	7
e) Tendo em conta a codificação que sugeriu em c), proponha um operador de mutação e exemplifique-o no contexto do problema da <i>Química_PT</i> .....	9
f) Os operadores propostos em d) e e) garantem a obtenção de soluções admissíveis para o problema da <i>Química_PT</i> ? Caso não garantam, justifique que tipos de inadmissibilidades os operadores indicados podem gerar e indique como pode ultrapassar esta situação na implementação do algoritmo genético .....	11
g) Com base nas alíneas anteriores, implemente um algoritmo genético que permita gerar uma solução admissível de qualidade para o problema da <i>Química_PT</i> . Dê uma breve explicação do algoritmo genético que implementou, nomeadamente dimensão da população, forma como é gerada a população inicial, método de seleção, método de substituição da população e critério(s) de paragem.....	14
h) Execute o código desenvolvido e faça uma breve análise à solução admissível obtida para o problema da <i>Química_PT</i> .....	15
Conclusão .....	17

## Introdução

O presente Trabalho de Grupo, foi realizado no âmbito da unidade curricular de Optimização Heurística, com o objetivo de aplicar e consolidar os conteúdos lecionados acerca do Capítulo II – Metaheurísticas, nomeadamente, acerca dos Algoritmos Genéticos.

Será explorado um problema que envolve a atribuição de tarefas a máquinas idênticas numa indústria do setor Químico, *Química\_PT*, que funcionam em paralelo e sem restrições de disponibilidade. O objetivo deste problema é minimizar o tempo entre a programação da primeira tarefa apresentada e a conclusão da última tarefa.

### a) Descreva, por palavras, uma solução admissível para o problema da *Química\_PT*

Uma solução admissível para o problema da *Química\_PT* tem de obedecer às seguintes condições:

- Todas as dez tarefas independentes têm de ser realizadas por uma das três máquinas idênticas M1, M2 ou M3;
- As máquinas M1 só podem ter 4 tarefas alocadas, as M2 só podem ter 3 e as M3 também só podem ter 3;
- Cada tarefa de T1 a T10, apenas pode ser processada uma única vez e apenas por uma das máquinas;
- T3, T4 e T5 têm de ser realizadas do dia 1 de junho às 8h até ao dia 5 de junho às 8h (96 horas no total), as restantes tarefas não têm data de conclusão já definida;
- As máquinas M1, M2 e M3 começam a realizar as tarefas dia 1 de junho às 8 horas.

### b) Desenvolva uma heurística para determinar uma sequência de processamento das tarefas em cada máquina que respeite a data de conclusão acordada para as tarefas T3, T4 e T5. Com base na heurística desenvolvida, apresente uma solução admissível para o problema da *Química\_PT*

Relativamente à definição de uma heurística, vamos começar por definir um conjunto de passos a seguir:

Passo 1: Ordenar as tarefas T3, T4 e T5 (prioritárias) por ordem crescente de tempos de processamento, para ver qual tem o menor tempo;

Passo 2: Colocar a tarefa prioritária com menor tempo de processamento em M1, pois é a máquina à qual vamos ter de alocar mais tarefas (4). Esta tarefa vai ser a primeira a ser realizada para garantir que o prazo de entrega é cumprido;

Passo 3: Colocar a tarefa prioritária com maior tempo de processamento em M3 e a restante tarefa prioritária em M2, ou seja, alocar as 2 tarefas prioritárias sobrantes por ordem crescente de tempo de processamento. As respetivas tarefas prioritárias vão ser as primeiras a ser realizadas para garantir que os compromissos assumidos são cumpridos na sua totalidade;

Passo 4: Considerando que M1 pode ter 4 tarefas alocadas e que contém a tarefa com menor número de horas de processamento, vamos alocar-lhe as 3 tarefas com maior tempo de processamento, se houver tarefas com igual número de horas de processamento alocamos a que tem o menor número após o T. A ordem de realização das tarefas, após a realização da tarefa prioritária é por ordem decrescente;

Passo 5: Como M3 tem a tarefa prioritária com o maior número de horas de processamento, vamos alocar-lhe as 2 tarefas ainda disponíveis com menor número de horas de processamento. A ordem de realização das tarefas, após a realização da tarefa prioritária é por ordem decrescente;

Passo 6: As 2 tarefas sobrantes vão ser alocadas a M2. A ordem de realização das tarefas, após a realização da tarefa prioritária é por ordem decrescente.

Agora vamos apresentar uma solução admissível considerando a heurística desenvolvida para o problema e em cada etapa apresentar o passo correspondente:

- ✓ Passo 1: O tempo de processamento das Tarefas T3, T4 e T5 é 36, 20 e 32 horas, respetivamente. Logo, a tarefa com o menor tempo de processamento é T4.
- ✓ Passo 2: T4 é alocada a M1 e esta é a primeira a ser realizada nesta máquina, pelo que ficamos com:
  - M1: T4 (20 horas)
  - M2: ---
  - M3: ---

- ✓ Passo 3: T5 é alocado a M2 e T3 a M3, sendo estas tarefas as primeiras a serem realizadas na máquina respetiva:
  - M1: T4 (20 horas)
  - M2: T5 (32 horas)
  - M3: T3 (36 horas)
- ✓ Passo 4: As tarefas com maior tempo de processamento são T7, T10, T1 e T8, com 46, 40, 38 e 38 horas, respetivamente. Como o número 1 é menor que 8, vamos alocar T1 a M1. Ou seja, as três tarefas restantes alocadas a M1 são T7, T10 e T1. A realização destas é por ordem decrescente do número de horas de processamento:
  - M1: T4 (20 horas), T7 (46 horas), T10 (40 horas), T1 (38 horas)
  - M2: T5 (32 horas)
  - M3: T3 (36 horas)
- ✓ Passo 5: Das tarefas ainda disponíveis as duas com menores tempos de processamento são T6 (29 horas) e T2 (33 horas). A realização das tarefas agora alocadas a M3 é por ordem decrescente do número de horas de processamento:
  - M1: T4 (20 horas), T7 (46 horas), T10 (40 horas), T1 (38 horas)
  - M2: T5 (32 horas)
  - M3: T3 (36 horas), T2 (33 horas), T6 (29 horas)
- ✓ Passo 6: As restantes duas tarefas são T8 (38 horas) e T9 (34 horas). A realização das tarefas agora alocadas a M2 é por ordem decrescente do número de horas de processamento:
  - M1: T4 (20 horas), T7 (46 horas), T10 (40 horas), T1 (38 horas)
  - M2: T5 (32 horas), T8 (38 horas), T9 (34 horas)
  - M3: T3 (36 horas), T2 (33 horas), T6 (29 horas)

De forma simplificada, podemos ver as alocações na Tabela 1:

	1ª Tarefa	2ª Tarefa	3ª Tarefa	4ª Tarefa
M1	T4	T7	T10	T1
M2	T5	T8	T9	Não aplicável
M3	T3	T2	T6	Não aplicável

*Tabela 1-Atribuição das tarefas às máquinas*

Estas alocações resultam nos seguintes tempos (Tabela 2):

	Nº horas 1ª Tarefa	Nº horas 2ª Tarefa	Nº horas 3ª Tarefa	Nº horas 4ª Tarefa	Total horas por máquina
M1	20	46	40	38	144
M2	32	38	34	Não aplicável	104
M3	36	33	29	Não aplicável	98

*Tabela 2-Tempo de processamento de cada tarefa*

O total de horas que demoramos até realizarmos todas as tarefas foram 144 horas, ou seja, 6 dias, acabando-as no dia de 7 junho às 8 horas.

### c) Defina um cromossoma que permita codificar uma sequência de processamento das tarefas em cada máquina

Antes de começar a definir um cromossoma específico para o problema, é importante percebermos em que este consiste. Um cromossoma é uma solução pertencente à população, para um problema específico.

O cromossoma que permite codificar a sequência de processamento das tarefas em cada máquina, tem tamanho 10, pois temos 10 tarefas para alocar a 3 máquinas.

Devido ao compromisso estabelecido e para conseguirmos saber a sequência de processamento de cada máquina, vamos definir que os valores dos primeiros 4 alelos, referentes aos genes nas posições 1 a 4 representam a ordem de execução das tarefas alocadas a M1. Os valores dos 3 alelos seguintes correspondentes aos genes nas posições 5 a 7 têm as tarefas alocadas a M2 por ordem de realização. Os valores dos últimos 3 alelos, nas posições 8 a 10 contém a sequência de tarefas a serem realizadas por M3.

Por exemplo, considerando a solução admissível presente na secção b), podemos definir o cromossoma da seguinte forma:

<b>M1</b>	<b>M1</b>	<b>M1</b>	<b>M1</b>	<b>M2</b>	<b>M2</b>	<b>M2</b>	<b>M3</b>	<b>M3</b>	<b>M3</b>
T4	T7	T10	T1	T5	T8	T9	T3	T2	T6

Ou seja, o cromossoma obtido seria (T4, T7, T10, T1, T5, T8, T9, T3, T2, T6).

Pelo que podemos conferir que M1 tem de realizar as suas 4 tarefas seguindo a ordem: T4, T7, T10 e T1; M2 tem de realizar as tarefas T5, T8 e T9, por ordem sequencial e M3 tem de realizar as tarefas T3, T2 e T6, por esta ordem.

d) Tendo em conta a codificação que sugeriu em c), proponha um operador de crossover e exemplifique-o no contexto do problema da *Química\_PT*

Antes de aplicarmos o operador crossover, vamos começar por perceber no que este consiste e os três tipos que existem.

O operador crossover é similar ao crossover que ocorre nos processos reprodutivos biológicos. Este é realizado, selecionando dois cromossomas pais que irão dar origem a dois cromossomas filhos utilizando a informação contida nos primeiros. O crossover é aplicado num algoritmo genético com elevada probabilidade - probabilidade de crossover.

Existindo 3 operadores de crossover que podemos utilizar neste trabalho:

- Crossover a um Ponto
- Crossover a  $k=2$  Pontos
- Crossover Uniforme

Decidimos aplicar o Crossover a um Ponto. Para o aplicarmos temos que:

- 1º. Definir de forma aleatória um número ( $m$ ), que varie entre 1 e a dimensão do cromossoma ( $n$ ), neste caso,  $n = 10$
- 2º. Dividir os dois cromossomas pai, que vão ser apelidados por Pai 1 e Pai 2, em duas “caudas”, considerando o  $m$  que foi escolhido anteriormente. A “cauda” esquerda contém os genes de 1 a  $m$  e a “cauda” direita contém os genes de  $m + 1$  a  $n$ .
- 3º. Cada filho fica com uma das “caudas” dos cromossomas pai. Ou seja, por exemplo, o primeiro cromossoma fica com a “cauda” esquerda do cromossoma Pai 1 e com a “cauda” direita do cromossoma do Pai 2. O segundo cromossoma fica com a “cauda” direita do cromossoma Pai 1 e com a “cauda” esquerda do cromossoma Pai 2.

O cromossoma Pai 1 pode ser o definido na alínea c), ou seja, (T4, T7, T10, T1, T5, T8, T9, T3, T2, T6).

Considerando os critérios definidos em a) para que tenhamos uma solução admissível e os critérios utilizados em d) para definir um cromossoma, vamos definir o Pai 2.

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T1	T4	T7	T10	T3	T9	T2	T5	T6	T8

Ou seja, o cromossoma Pai 2 é (T1, T4, T7, T10, T3, T9, T2, T5, T6, T8). Logo as tarefas de M1 por ordem sequencial são T1, T4, T7 e T10, as de M2 são T3, T9 e T2 e as de M3 são T5, T6 e T8.

Deste modo, ambos os cromossomas Pai 1 e Pai 2 são admissíveis.

Agora vamos considerar para o Crossover a um Ponto,  $m = 4$ . Para ser mais perceptível o crossover e como é que os cromossomas filhos foram originados vamos colocar cores diferentes nas tarefas dos cromossomas Pai 1 e Pai 2, para percebermos a influência de cada pai e os podermos distinguir.

#### Cromossoma Pai 1

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T7	T10	T1	T5	T8	T9	T3	T2	T6

#### Cromossoma Pai 2

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T1	T4	T7	T10	T3	T9	T2	T5	T6	T8

#### Cromossoma Filho 1

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T7	T10	T1	T3	T9	T2	T5	T6	T8

#### Cromossoma Filho 2

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T1	T4	T7	T10	T5	T8	T9	T3	T2	T6

Os cromossomas Filho 1 e Filho 2 são admissíveis, pois as tarefas T1 a T10, apenas são realizadas uma e uma só vez, o número de tarefas que podem ser alocadas a cada máquina e o compromisso são respeitados.



e) Tendo em conta a codificação que sugeriu em c), proponha um operador de mutação e exemplifique-o no contexto do problema da *Química\_PT*

Antes de mais, vamos perceber o que é uma mutação. Esta, é um pequeno ajuste aleatório no cromossoma, para obter uma nova solução. O operador de mutação é usado para manter e introduzir diversidade na população. Este operador é aplicado com uma probabilidade baixa - probabilidade de mutação -, pois se a probabilidade for muito elevada o Algoritmo Genético restringe-se a uma pesquisa aleatória. Assim, a mutação é a componente dos Algoritmos Genéticos que está associada à “exploração” do espaço de pesquisa.

Existem 3 operadores de mutação que podemos utilizar neste trabalho:

- Mutação Bit Flip
- Mutação por Troca
- Mutação por Inversão

É de notar que como o nosso Algoritmo Genético não está codificado em binários (0/1), nunca poderíamos utilizar a Mutação Bit Flip.

Decidimos então aplicar a Mutação por Troca. Esta é utilizada em Algoritmos Genéticos codificados como permutações, tal como é o nosso caso. Os passos para a aplicação deste operador são:

- 1º. Gerar aleatoriamente um número com distribuição uniforme no intervalo  $[0, 1]$ , cujo símbolo é ***u***;
- 2º. Se o número gerado ***u***, for inferior à probabilidade de mutação (***pm***), então selecionam-se aleatoriamente dois genes e trocam-se o valor dos seus alelos.

Para gerarmos o valor de ***u*** e escolhermos aleatoriamente os valores dos alelos a trocar criamos um código com semente fixa (com valor de 123), para que os resultados obtidos sejam sempre os mesmos. Os resultados estão presentes no ficheiro *codigo\_final\_grupo1* e aplicamos o algoritmo criado aos cromossomas Filho 1 e Filho 2, criados na secção d).

É de notar que o valor de ***u*** obtido foi de cerca de 0.05 e consideramos ***pm*** = 0.1, pois quanto menor o valor melhor. O valor do alelo que foi trocado está realçado, para ser mais fácil de perceber as mudanças realizadas. Deste modo, obtivemos os seguintes resultados:

### **Cromossoma Filho 1 inicial**

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T7	T10	T1	T3	T9	T2	T5	T6	T8

### **Cromossoma Filho 1 com Mutação por Troca**

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T2	T10	T1	T3	T9	T7	T5	T6	T8

Deste modo, relativamente à mutação no Cromossoma Filho 1 inicial o valor do alelo referente às tarefas T7 e T2 é trocado, ficando T2 no gene na posição 2 e T7 no gene na posição 7. O cromossoma Filho 1 com a Mutação por Troca continua a ser admissível, pois todas as tarefas só são realizadas uma e uma única vez, o número de tarefas por máquina é respeitado e as tarefas prioritárias T3, T4 e T5 são realizadas no tempo acordado.

### **Cromossoma Filho 2 inicial**

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T1	T4	T7	T10	T5	T8	T9	T3	T2	T6

### **Cromossoma Filho 2 com Mutação por Troca**

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T9	T7	T10	T5	T8	T4	T3	T2	T6

Deste modo, relativamente à mutação no Cromossoma Filho 2 inicial, o valor do alelo referente às tarefas T4 e T9 é trocado, ficando T9 no gene na posição 2 e T4 no gene na posição 7. O cromossoma Filho 2 com a Mutação por Troca continua a ser admissível, pois todas as tarefas só são realizadas uma e uma única vez, o número de tarefas por máquina é respeitado e as tarefas prioritárias T3, T4 e T5 são realizadas no tempo acordado.

f) Os operadores propostos em d) e e) garantem a obtenção de soluções admissíveis para o problema da *Química\_PT*? Caso não garantam, justifique que tipos de inadmissibilidades os operadores indicados podem gerar e indique como pode ultrapassar esta situação na implementação do algoritmo genético

Antes de concluirmos se os operadores propostos em d) e e) são admissíveis, vamos perceber o que faz uma solução ser considerada admissível neste caso específico.

Uma solução admissível necessita de cumprir todos os requisitos definidos em a), ou seja, a solução necessita de alocar todas as tarefas em consideração, sendo que cada uma só pode ser executada uma e uma só vez e apenas por uma máquina, sendo que M1, M2 e M3 só podem ter 4, 3 e 3 tarefas designadas, respetivamente. Além disso, as tarefas T3, T4 e T5 têm de ser finalizadas no dia 5 de junho às 8 horas, sendo que todas as máquinas iniciam o seu funcionamento dia 1 de junho às 8 horas.

Tal como já foi explicado nas alíneas d) e e), os cromossomas que criamos foram admissíveis, mas caso não fossem tínhamos duas opções, consoante admitíssemos ou não soluções não admissíveis na população. Se só admitíssemos soluções admissíveis na população, teríamos de desenvolver um procedimento que transformasse as soluções não admissíveis em soluções admissíveis, pelo que iríamos identificar a tarefa que não está a ser realizada e a tarefa que está a ser realizada mais do que uma vez e identificar as tarefas prioritárias que não cumprem o compromisso, substituindo-as por outra tarefa que ainda estivesse disponível. Esta opção vai ser explicada em seguida.

Se por sua vez, admitíssemos soluções admissíveis e não admissíveis na população, quando tivéssemos um cromossoma não admissível poderíamos incluí-lo na população, mas o seu valor de aptidão sofreria uma penalização. Ou seja, desenvolveríamos uma função de avaliação que contabilizasse não só o tempo de processamento de cada tarefa, mas também uma parcela adicional que penalizasse a avaliação da solução, por ser inadmissível.

Apesar de nas alíneas anteriores esta situação não ocorrer, quando criamos o código da alínea g) tivemos em consideração a possibilidade de ocorrência de soluções não admissíveis e decidimos que na nossa população, apenas iríamos considerar cromossomas

admissíveis, pelo que optamos por desenvolver um procedimento que transformasse as soluções não admissíveis em soluções admissíveis ao alterar posições aleatoriamente.

Ou seja, no Crossover a um ponto, verificamos se todas as tarefas estavam alocadas e se existissem tarefas repetidas, a primeira posição onde esta aparece é mantida e na segunda posição esta é alterada, por uma das tarefas que ainda não esteja alocada. O mesmo procedimento é realizado para o número de horas, caso as tarefas prioritárias não sejam cumpridas a tempo.

Deste modo, para ser mais fácil a compreensão deixamos um exemplo prático. Começamos com dois cromossomas Pais admissíveis e utilizamos  $m = 4$ .

#### Cromossoma Pai 1

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T6	T9	T10	T5	T1	T2	T3	T7	T8

#### Cromossoma Pai 2

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T2	T10	T4	T8	T1	T5	T9	T3	T6	T7

#### Cromossoma Filho 1 inicial

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T6	T9	T10	T1	T5	T9	T3	T6	T7

Podemos observar de imediato que o Cromossoma Filho 1 inicial não é admissível, pois as tarefas T6 e T9 estão repetidas duas vezes e as tarefas T2 e T8 não são realizadas. Relativamente, à realização das tarefas prioritárias, estas são cumpridas no tempo devido.

#### Cromossoma Filho 2 inicial

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T2	T10	T4	T8	T5	T1	T2	T3	T7	T8

Podemos observar que o Cromossoma Filho 2 inicial não é admissível, pois as tarefas T2 e T8 aparecem duas vezes, não dando lugar à realização das tarefas T6 e T9. Relativamente, à realização das tarefas prioritárias, estas são cumpridas no tempo devido.

Assim, o procedimento criado tem de entrar em ação, para obtermos soluções admissíveis. Os resultados obtidos estão apresentados em baixo:

#### **Cromossoma Filho 1 correto**

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T4	T6	T9	T10	T1	T5	T8	T3	T2	T7

Verificamos que todas as tarefas são realizadas uma e uma única vez, sendo que as tarefas T6 e T9 que eram realizadas duas vezes são mantidas nas primeiras posições em que apareciam no Cromossoma Filho 1 inicial, ou seja, nos genes nas posições 2 e 3, respetivamente. T9 que aparecia repetida na posição 7 do cromossoma é substituída por T8 e T6 que era repetida na posição 9 é substituída pela tarefa restante, ou seja, T2. Continua a não haver problema relativamente ao cumprimento das tarefas prioritárias.

#### **Cromossoma Filho 2 correto**

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T2	T10	T4	T8	T5	T1	T6	T3	T7	T9

Neste cromossoma todas as tarefas são realizadas apenas uma vez. T2 e T8 são mantidas nas posições iniciais na primeira vez que aparecem no cromossoma, ou seja, nos genes nas posições 1 e 4, respetivamente. Na posição 7 do cromossoma, T2 é substituído por T6 e T8 na posição 10 por T9. Relativamente, à realização das tarefas prioritárias, estas são cumpridas no tempo devido.

Relativamente à Mutação por Troca, decidimos apenas verificar a admissibilidade, considerando o número de horas decorridas até ao fim do processamento das tarefas prioritárias, pois consideramos que o operador de crossover garantia sempre a admissibilidade das soluções no que respeita a cada uma ser apenas realizada uma vez e na realização das tarefas prioritárias no tempo devido. Pelo que a nossa única preocupação foi garantir que as tarefas prioritárias ainda eram cumpridas, ou seja, sempre que a troca envolvia as tarefas T3, T4 e T5 verificamos se esta troca implica o não cumprimento destas tarefas no número de horas previstas. Se assim for, o cromossoma inicial não é

alterado. Pelo que temos uma resposta muito parecida à situação em que a mutação não ocorre porque  $u > pm$ . Vamos em seguida, apresentar um exemplo prático:

#### Cromossoma Filho inicial

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T2	T5	T4	T8	T9	T1	T10	T3	T7	T6

#### Cromossoma Filho com a mutação corrigida

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T2	T5	T4	T8	T9	T1	T10	T3	T7	T6

Neste exemplo, podemos verificar que apesar do nosso valor de  $u$  obtido ser de 0.05 e  $pm = 0.1$ , não ocorreu mutação porque a troca deveria envolver T3, T4 ou T5 e implicava que estas tarefas não iam ser realizadas na sua totalidade no tempo acordado, logo o Cromossoma Filho inicial ficou igual ao Cromossoma Filho com a mutação corrigida.

g) Com base nas alíneas anteriores, implemente um algoritmo genético que permita gerar uma solução admissível de qualidade para o problema da *Química\_PT*. Dê uma breve explicação do algoritmo genético que implementou, nomeadamente dimensão da população, forma como é gerada a população inicial, método de seleção, método de substituição da população e critério(s) de paragem

Vamos agora implementar um Algoritmo Genético com base também no que já realizamos nas alíneas anteriores.

- Dimensão da população: deve ser igual em todas as iterações do Algoritmo Genético e deve assegurar a sua diversidade. Devem ser testados vários valores para a dimensão da população. Este problema tem cerca de  $C_4^{10} * C_3^6 * C_3^3 = 4200$  soluções admissíveis, isto se não contarmos com as restrições temporais de T3, T4 e T5. Inicialmente, consideramos que a população é constituída por 20 cromossomas. Esta dimensão está relacionada com o facto de ser mais fácil

perceber exatamente o que está a acontecer e corrigir eventuais erros. Após assegurarmos que o algoritmo estava bem desenvolvido decidimos experimentar para uma população de dimensão 200. Esta dimensão foi escolhida, pois acreditamos que se se estivéssemos na presença de uma população maior o algoritmo iria ficar muito pesado e seria muito difícil percebermos exatamente o que o nosso algoritmo estava a fazer. Os resultados apresentados na alínea h) foram para uma população de dimensão 200.

- Como se gera a população inicial: a geração da população inicial foi feita de forma aleatória, mas tinha de cumprir algumas condições. Consideramos que o cromossoma teria de ter dimensão 10 e o valor de cada alelo teria de ser diferente, ou seja, cada tarefa estava apenas presente uma vez. Além disso, respeita-se a data de entrega acordada para as tarefas T3, T4 e T5, pois garantimos que decorrem sempre no máximo 96 horas até que estas sejam efetuadas na sua totalidade. A população inicial é composta por 200 cromossomas.
- Método de seleção: utilizamos o método de seleção dos pais por Torneio. Ou seja, selecionamos ao acaso K, em que K é um valor inteiro, indivíduos da população. Entre esses K indivíduos, escolhe-se o melhor, ou seja, o mais apto, para ser um dos pais. Repete-se o mesmo processo para se escolher o outro pai do par. Neste caso, escolhemos K=200, pois assim pode escolher os melhores pais dentro de toda a população e caso o pai escolhido seja igual, repete-se o processo, pelo que nunca vamos ter dois pais iguais escolhidos.
- Método de substituição da população: utilizamos o Modelo Estacionário, em cada iteração. Ou seja, em cada iteração existe a substituição de 10% da população, ou seja, neste caso de 20 cromossomas. Substituímos pelos filhos gerados utilizando o Crossover a um ponto e mutações destes usando a Mutação por Troca (com  $pm = 0.1$ ), caso o valor de 10% não seja alcançado apenas com os filhos gerados.
- Critérios de paragem: quando chegarmos ao número máximo de iterações possíveis que definimos, neste caso, 40 ou quando não conseguirmos melhorias ao fim de 7 iterações.

#### h) Execute o código desenvolvido e faça uma breve análise à solução admissível obtida para o problema da *Química\_PT*

Abaixo podemos ver o pseudocódigo desenvolvido para um critério de paragem de 40 iterações ou 7 iterações sem melhoria:

---

**Algorithm 1** : Algoritmo genético completo

---

```
definir seed inicial
Número de iterações ← definir iterações para o algoritmo genético
Dimensão população inicial ← definir número de cromossomas presentes na população inicial
for número de cromossomas existentes na população inicial do
    gerar solução inicial
    calcular aptidão de cada máquina
Exibir a população inicial
Max iterações sem melhoria ← definir critério de paragem para o algoritmo genético quando
não há melhoria
Melhor solução e aptidão encontradas ← começa vazio para o ciclo das iterações
for número iterações definido do
    Seleção dos pais por torneio(...)
    Aplicar o crossover a um ponto para a geração de filhos a partir dos pais(...)
    probabilidade de mutação ← define um valor de probabilidade máxima para realizar as mu-
tações
    for cromossomas na população do
        if probabilidade < probabilidade de mutação then
            Realizar a mutação por troca no cromossoma
            if tarefas prioritárias excedem limite das 96 horas then
                alterar posições das tarefas aleatoriamente
    nova população ← lista vazia para colocar a nova população gerada
    for cromossomas na população do
        calcular aptidão e adicionar cromossomas à nova população
    for cromossomas na nova população do
        if melhor solução atual for None ou inferior à guardada then
            guardar nova melhor solução
    if melhor valor da população < melhor valor encontrado then
        iteração sem melhoria passa a 0
    else
        incrementar 1 à iteração sem melhoria
    if critério de paragem atingido then
        atingido o critério de paragem: nenhum progresso atingido
        raise SystemExit: Fim Algoritmo Genético
    mostrar nova população
    mostrar melhor solução encontrada até ao momento
    if taxa de substituição não alcançada then
        substituir cromossomas por mutações diferentes de filhos gerados
mostrar a população da última iteração
melhor solução e aptidão encontrada
```

---

Através do pseudocódigo podemos perceber que a melhor solução admissível para este problema é o menor dos valores máximos obtidos em cada cromossoma, em cada iteração da população.

Neste caso, a melhor solução encontrada foi de 116. Este valor acompanha-nos desde a população inicial, pelo que o algoritmo foi interrompido, após a sétima iteração sem melhoras. É de notar que conseguimos tirar esta conclusão, pois como se pode verificar no código após a sétima iteração sem melhorias lança-se um SystemExit, que termina o programa.



O cromossoma que dá origem à melhor solução admissível com valor de 116 é:

M1	M1	M1	M1	M2	M2	M2	M3	M3	M3
T9	T2	T4	T6	T3	T10	T1	T7	T5	T8

Verificamos de imediato que esta solução é admissível, pois, temos todas as tarefas presentes e só são realizadas uma vez, as máquinas M1, M2 e M3 têm o número de tarefas a serem alocadas completas. As tarefas T3, T4 e T5 são terminadas durante o período acordado. Relativamente, aos tempos decorridos em cada máquina:

- M1:  $T9 + T2 + T4 + T6 = 34 \text{ horas} + 33 \text{ horas} + 20 \text{ horas} + 32 \text{ horas} = 116 \text{ horas}$
- M2:  $T3 + T10 + T1 = 36 \text{ horas} + 40 \text{ horas} + 38 \text{ horas} = 114 \text{ horas}$
- M3:  $T7 + T5 + T8 = 46 \text{ horas} + 32 \text{ horas} + 38 \text{ horas} = 116 \text{ horas}$

Deste modo, percebemos, que o maior número de horas decorridas numa das máquinas até à realização de todas as tarefas foi de 116 horas, sendo este o valor de aptidão do cromossoma. Ou seja, completamos todas as tarefas no dia 6 de junho por volta das 4 horas da manhã.

## Conclusão

A melhor solução admissível a que chegamos usando um Algoritmo Genético, com uma população de dimensão 200, com seleção dos Pais por Torneio, operador de Crossover a um ponto e Mutação por Troca, com critérios de paragem de 40 iterações ou 7 iterações sem melhoras (ou seja, sem conseguirmos diminuir o mínimo dos valores máximos) foi de 116.