Universidade do Minho

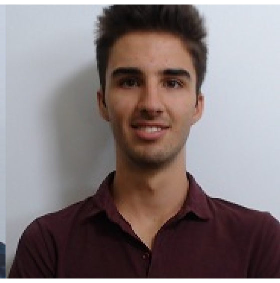
9 de janeiro de 2021

Relatório do trabalho prático 3

Métodos Determinísticos de Investigação Operacional

Mestrado Integrado em Engenharia Informática - 3º ano






Adriano Novo Soto Maior A89483	Bruno Pinto Jácome A89515	José Pedro Ribeiro Peixoto A89602	Afonso Trindade Araújo de Pascoal Faria A83920

Índice

1 | Introdução

Este relatório foi realizado no âmbito da Unidade Curricular de Métodos Determinísticos em Investigação Operacional, envolvendo o conceito programação inteira mista, com auxílio do *software* **lpsolve**. Tem como objetivos principais, abordar o raciocínio e mostrar uma solução para um problema de minimização do custo de redução do tempo de execução, sobre escalonamento de atividades.
Este relatório divide-se essencialmente em 4 partes, todas relacionadas com a resolução do enunciado do trabalho prático: Na primeira parte, identifica-se a rede obtida pelo grupo de acordo com os requisitos do enunciado e o diagrama de Gantt. Já na segunda parte, modela-se o problema e apresenta-se o ficheiro input. Depois, analisa-se os resultados obtidos pela aplicação do *lpsolve* e apresenta-se a solução ótima, isto é, o custo ótimo adicional associado a reduções nas durações nas atividades. Em último lugar, valida-se a solução ótima através da verifação do cumprimento das restrições e do valor ótimo para o custo igualar o valor obtido no ficheiro output.

2 | Resolução das questões

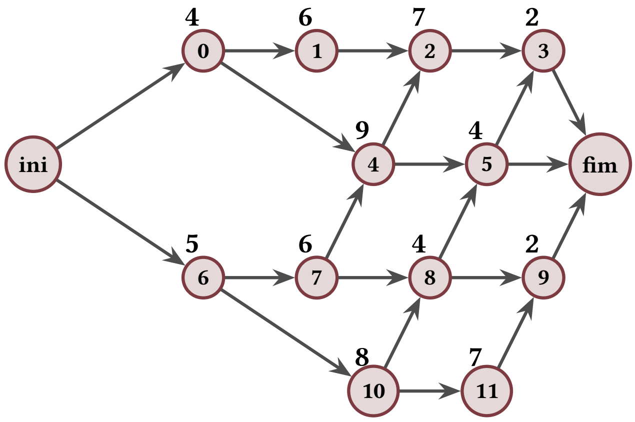
2.1 Indique o valor de ABCDE, e apresente a rede que representa o projecto depois de eliminar as actividades indicadas, identificando os vértices da rede e os arcos e respectivos custos

2.1.1 Valor de ABCDE

Comparando todos os números de aluno do grupo de trabalho (89483, 89515, 89602, 83920), verifica-se que o número maior é o 89602. Portanto, **A = 8, B = 9, C = 6, D = 0 e E = 2**.

2.1.2 Rede resultante

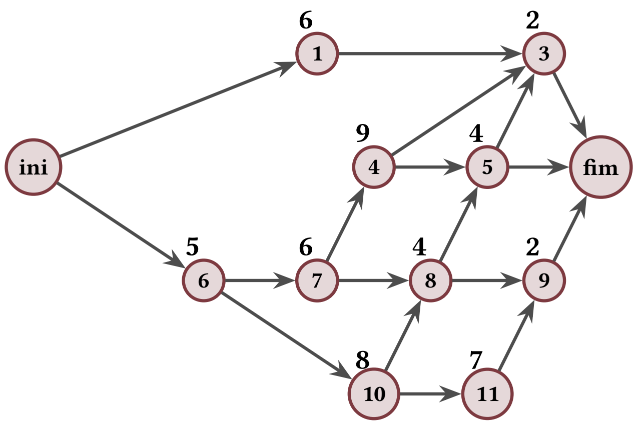
A partir do grafo original:



Removem-se os vértices D e E, respetivamente, 0 e 2.

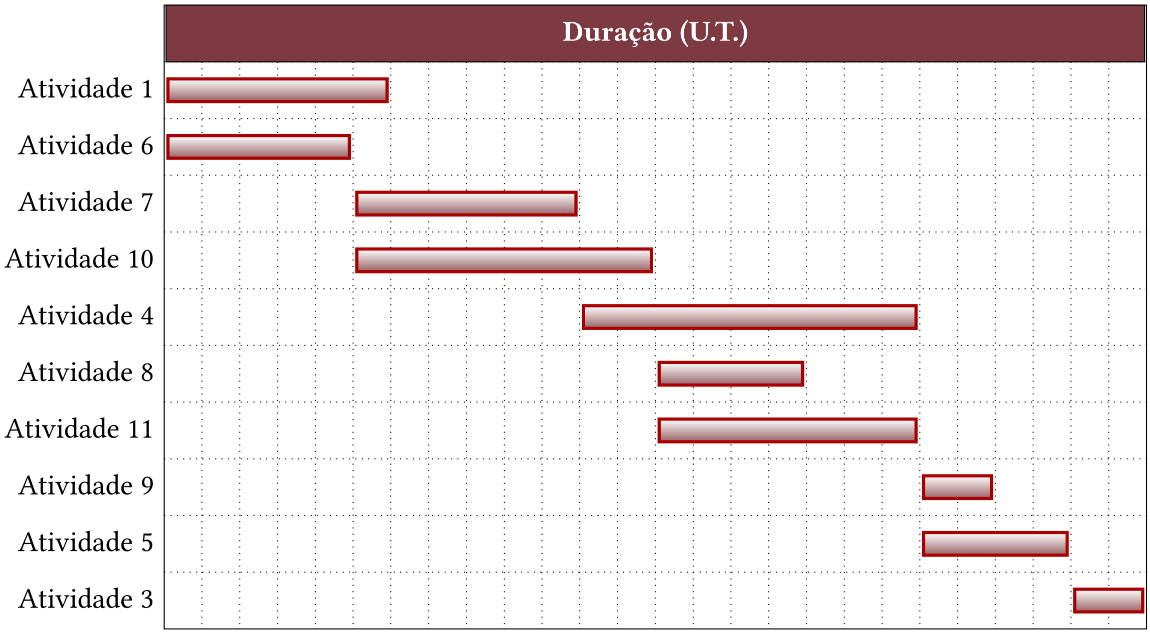
- Como a atividade 0 é removida, ini é a nova precedência das atividades 1 e 4 contudo, como 4 ainda depende de 7, então ignora-se o arco ini → 4. Ou seja, **ini é a nova precedência da atividade 1**.
- Como a atividade 2 é removida, **1 e 4 são as novas precedências da atividade 3**.

Originando o seguinte grafo:



2.2 Apresente o diagrama de Gantt (que resulta de resolver o modelo com as variáveis de decisão), e indique a duração do projecto

2.2.1 Diagrama de Gantt



2.2.2 Duração do projeto

Como se pode ver no diagrama de Gantt a duração mínima do projeto é 26 U.T., que é o custo (duração) do caminho crítico ($6 \rightarrow 7 \rightarrow 4 \rightarrow 5 \rightarrow 3$).

2.3 Apresente a formulação apenas das novas restrições e da nova função objectivo do novo modelo de programação linear inteira mista

2.3.1 Contextualização

Tempo de execução

O problema de otimização consiste em minimizar o custo total da realização de atividades, com redução no tempo de execução do projeto em 3 U.T.. Estas atividades têm durações e as suas realizações podem depender de outras:

Atividade	Duração (U.M.)	Precedências
1	6	
3	2	1, 4, 5
4	9	
5	4	4,8
6	5	
7	6	6
8	4	7, 10
9	2	8, 11
10	8	6
11	7	10

Custos de atividade e reduções nas durações das atividades

Além do tempo de execução de cada atividade, também é necessário considerar o custo de cada atividade.

Todavia, poderá reduzir-se o tempo de atividade, em troca de um custo adicional ao custo normal dessa mesma atividade. Por isso, para cada atividade $i \in 1, 3, 4, 5, 6, 8, 10, 11$, é possível:

- Não reduzir, i.e., escolher custos/durações normais.
- Reduzir a duração da atividade i em r_{1i} U.T, $0 <= r_{1i} <= m_{1i}$, com um custo adicional de $r_{1i} * c_{1i}$ U.M..
- Reduzir a duração da atividade i em $m_{1i} + r_{2i}$ U.T., $0 <= r_{2i} <= m_{2i}$, com um custo adicional de $m_{1i} * c_{1i} + r_{2i} * c_{2i}$ U.M..

Atividade	Custo Normal	c_1 (U.M./U.T.)	Redução Máxima 1	c_2 (U.M./U.T.)	Redução Máxima 2
1	1000	600	1	300	1
3	300	200	0.5	100	0.5
4	2000	800	2	400	1
5	1000	1600	0.5	800	0.5
6	800	180	1	90	1
7	900				
8	600	200	0.5	100	0.5
9	300				
10	1600	1000	0.5	500	0.5
11	1400	600	1	300	1

Custos alternativos

Para cada uma das atividades 7 e 9 - notemos que são as únicas, na tabela anterior, sem reduções - existem 2 reduções alternativos. Por isso, para cada atividade $i \in 7, 9$, é possível:

- Não reduzir, i.e., escolher custos/durações normais.
- Reduzir a duração da atividade i em t_{ai} U.T, com um custo adicional de c_{ai} U.M.
- Reduzir a duração da atividade i em t_{bi} U.T., com um custo adicional de c_{bi} U.M.

Atividade	Custo Normal	c_a	Redução *a* (U.T.)	c_b	Redução *b* (U.T.)
7	900	300	1	1100	2
9	300	200	1	400	2

2.3.2 Considerações

Objetivo

Explicou-se ,anteriormente, a possibilidade de reduzir o tempo de execução de atividades individuais, em troca de um custo adicional. Então, o verdadeiro objetivo do problema é agora **minimizar o custo adicional**, para uma redução no tempo de execução de 3 U.T.

Nota 1: a soma de todos os custos normais das atividades é **9900 U.M.**.

Nota 2: o tempo de execução mínimo normal do projeto é **26 U.T.**.

Otimização: Minimizar o custo total adicional da relização do projeto.

Para isto, utilizou-se a ferramenta de otimização *lpsolve* e enunciou-se o problema no ficheiro **GestaoProjetoCrashingTimes.lp**, gerando-se o ficheiro *output* **solution**.

2.4 Apresente o modelo de programação linear inteira mista

2.4.1 Variáveis de decisão

$r_{xi} \in \left[0,\ m_{xi}\right],$ redução do tipo x para a atividade i
$k_{yj} \in \{0,1\}$, se escolheu-se, ou não, a redução do tipo y para a atividade j
Onde, $i \in \{1,3,4,5,8,10,11\}, \quad j \in \{7,9\}, \quad x \in \{1,2\}, \quad y \in \{\text{a},\text{b}\}$

2.4.2 Parâmetros

Reduções máximas

Considere-se,

$m_{xi},$ a redução temporal máxima do tipo x para a atividade i
Onde, $i \in \{1,3,4,5,8,10,11\}, \quad x \in \{1,2\}$

Dependências nas reduções nas atividades $\in \{1,3,4,5,6,8,10,11\}$

Notemos que, a redução do tipo 2 (a um nodo i) só é válida quando a redução do tipo 1 (ao mesmo nodo i) é máxima. Por exemplo, $r_{21} = 0.5 \implies r_{11} = 1$: a redução do tipo 2 da duração da atividade 1 em 0.5 U.T. obriga a que a redução do tipo 1 da duração da atividade 1 é máxima (1 U.T.). Ora, esta implicação pode ser traduzida para:

$$r_{2i} > 0 \implies r_{1i} = m_{1i} \tag{2.1}$$

Considere-se ainda uma variável k_i binária tal que:

$k_i = 1,$ se $r_{1i} = m_{1i}$, isto é, se a redução do tipo 1 à atividade i é máxima
$k_i = 0,$ senão

Então, podemos rescreever 2.1 como:

$$r_{2i} > 0 \implies k_i = 1$$

Uma vez que, $r_{2i} \leq m_{2i}$, então:

$$r_{2i} \leq m_{2i} \times k_i$$

- Se $r_{2i} > 0$, então $\frac{r_{2i}}{m_{2i}} \in \left]0,1\right]$ e, por isso, $k_i = 1$ (ou seja, $r_{1i} = m_{1i}$).
- Senão ($r_{2i} = 0$), então $\frac{r_{2i}}{m_{2i}} = 0$ e, por isso, $k_i \in \{0,1\}$ (ou seja, r_{1i} pode ser, ou não, máximo).

Além disso, notemos que se $k_i = 1$, ou seja, se $r_{2i} > 0$, então $r_{1i} = m_{1i}$. Notemos que $r_{1i} = m_{1i}$ é o mesmo que $r_{1i} \leq m_{1i}$ e $r_{1i} \geq m_{1i}$. Como, $r1i \leq m_{1i}$, então:

$$r_{1i} \geq m_{1i} \times k_i$$

Em suma, e notando que a redução total a uma atividade i é a soma dos dois tipos de reduções:

$$\boxed{\begin{aligned} r_i &= r_{1i} + r_{2i} \\ r_{1i} &\geq m_{1i} \times k_i \\ r_{2i} &\leq m_{2i} \times k_i \end{aligned}}$$

Através do quadro seguinte, confirmam-se as afirmações anteriores:

r_{2i}	r_{2i} / m_{2i}	k_i ($r_{2i} \leq m_{2i} \times k_i \Leftrightarrow k_i \geq r_{2i} / m_{2i}$)	$m_{1i} \times k_i$	r_{1i} ($r_{1i} \geq m_{1i} \times k_i$)
$]0, m_{2i}]$	$]0, 1]$	1	m_{1i}	m_{1i}
0	0	{0,1}	$\{0, m_{1i}\}$	$[0, m_{1i}]$

2.4.3 Função Objetivo

Recapitulando, sejam:

- $r_{xi} \in [0, m_{xi}]$, redução do tipo x para a atividade i
- c_{xi} o custo adicional (U.M/U.T.) do tipo x de realizar a tarefa i
- $k_{yj} \in \{0, 1\}$, se escolheu-se, ou não, a redução do tipo y para a atividade j
- c_{yj} o custo adicional do tipo y de realizar a tarefa j

Onde, $i \in \{1, 3, 4, 5, 8, 10, 11\}$, $j \in \{7, 9\}$, $x \in \{1, 2\}$, $y \in \{\mathrm{a}, \mathrm{b}\}$

Então, o custo total adicional da realização das atividades é:

$$\sum_{ijxy} [c_{xi} r_{xi} + c_{yj} k_{yj}]$$

Portanto, minimizar o custo total adicional é:

$$\boxed{\min : \sum_{ijxy} [c_{xi} r_{xi} + c_{yj} k_{yj}]}$$

A partir da tabela de custos e reduções, enunciou-se a seguinte função objetivo:

```
min:  600 r1_1  +  300 r2_1      +
      200 r1_3  +  500 r2_3      +
      800 r1_4  +  400 r2_3      +
     1600 r1_5  +  800 r2_5      +
      180 r1_6  +   90 r2_6      +
      300 ka7     + 1100 kb7      +
      300 r1_8  +  100 r2_8      +
      200 ka9     +  400 kb9      +
     1000 r1_10 +  500 r2_10     +
      600 r1_11 +  300 r2_11     ;
```

2.4.4 Restrições

Redução total

Lembra-se que é exigido que o tempo de execução do projeto seja reduzido em 3 U.T.. logo, o tempo total de execução do programa será de 26 - 3 = 23 U.T.

```
tf = 23;
```

Dependências nas reduções nas atividades $\{1, 3, 4, 5, 6, 8, 10, 11\}$

Conforme foi enunciado anteriormente em 2.4.2, enunciaram-se as seguintes restrições, no ficheiro de input:

```
r1  = r1_1  + r2_1  ;   r1_1  <= 1   ;   r1_1  >= 1   * k1  ;   r2_1  <= 1   * k1  ;
r3  = r1_3  + r2_3  ;   r1_3  <= 0.5 ;   r1_3  >= 0.5 * k3  ;   r2_3  <= 0.5 * k3  ;
r4  = r1_4  + r2_4  ;   r1_4  <= 2   ;   r1_4  >= 2   * k4  ;   r2_4  <= 1   * k4  ;
r5  = r1_5  + r2_5  ;   r1_5  <= 0.5 ;   r1_5  >= 0.5 * k5  ;   r2_5  <= 0.5 * k5  ;
r6  = r1_6  + r2_6  ;   r1_6  <= 1   ;   r1_6  >= 1   * k6  ;   r2_6  <= 1   * k6  ;
r8  = r1_8  + r2_8  ;   r1_8  <= 0.5 ;   r1_8  >= 0.5 * k8  ;   r2_8  <= 0.5 * k8  ;
r10 = r1_10 + r2_10 ;   r1_10 <= 0.5 ;   r1_10 >= 0.5 * k10 ;   r2_10 <= 0.5 * k10 ;
r11 = r1_11 + r2_11 ;   r1_11 <= 1   ;   r1_11 >= 1   * k11 ;   r2_11 <= 1   * k11 ;
```

Dependências nas reduções nas atividades $\{7, 9\}$

Ao contrário das dependências nas reduções anteriores, as reduções agora são alternativas exclusivas (xor). Isto é, para cada atividade $j \in \{7, 9\}$ apenas se poderá, no máximo, optar por um dos dois tipos (a ou b) de reduções.

Considere-se $k_{yj} \in \{0, 1\}, y \in \{a, b\}, j \in \{7, 9\}$ o parâmetro que indica que se optou pela redução do tipo y para a atividade j. Desta forma, tem-se que:

$$\sum_{y} k_{yj} \leq 1$$

Com isto, enunciaram-se as seguintes restrições no ficheiro de input:

```
r7 = ka7 + 2 kb7; ka7 + kb7 <= 1;
r9 = ka9 + 2 kb9; ka9 + kb9 <= 1;
```

2.5 Apresente o ficheiro de input

2.5.1 GestaoProjetoCrashingTimes.lp

```
min:  600 r1_1  +  300 r2_1      +
      200 r1_3  +  500 r2_3      +
      800 r1_4  +  400 r2_3      +
     1600 r1_5  +  800 r2_5      +
      180 r1_6  +   90 r2_6      +
      300 ka7   + 1100 kb7       +
      300 r1_8  +  100 r2_8      +
      200 ka9   +  400 kb9       +
     1000 r1_10 +  500 r2_10     +
      600 r1_11 +  300 r2_11     ;

// tempo máximo para concluir o projecto
tf = 23;

// relações de precedência
// na restrição ti >= di + tj - ri, a função di - ri representa
// o tempo de conclusão da actividade i após a redução da duração
// de di para di - ri

arco_i_1:   t1  >=     -  r1  + 6  ;
arco_1_3:   t3  >= t1  -  r3  + 2  ;
arco_3_f:   tf  >= t3              ;
arco_4_3:   t3  >= t4  -  r3  + 2  ;
arco_4_5:   t5  >= t4  -  r5  + 4  ;
arco_5_3:   t3  >= t5  -  r3  + 2  ;
arco_5_f:   tf  >= t5              ;
arco_i_6:   t6  >=     -  r6  + 5  ;
arco_6_7:   t7  >= t6  -  r7  + 6  ;
arco_7_4:   t4  >= t7  -  r4  + 9  ;
arco_7_8:   t8  >= t7  -  r8  + 4  ;
arco_8_9:   t9  >= t8  -  r9  + 2  ;
arco_9_f:   tf  >= t9              ;
arco_6_10:  t10 >= t6  -  r10 + 8  ;
arco_10_8:  t8  >= t10 -  r8  + 4  ;
arco_10_11: t11 >= t10 -  r11 + 7  ;
arco_11_9:  t9  >= t11 -  r9  + 2  ;

// reduções máxima permitidas do tipo 1 e 2 (só possível se a redução do tipo 1 é max)
r1  = r1_1  + r2_1  ;   r1_1  <= 1   ;   r1_1  >= 1   * k1  ;   r2_1  <= 1   * k1  ;
r3  = r1_3  + r2_3  ;   r1_3  <= 0.5 ;   r1_3  >= 0.5 * k3  ;   r2_3  <= 0.5 * k3  ;
r4  = r1_4  + r2_4  ;   r1_4  <= 2   ;   r1_4  >= 2   * k4  ;   r2_4  <= 1   * k4  ;
r5  = r1_5  + r2_5  ;   r1_5  <= 0.5 ;   r1_5  >= 0.5 * k5  ;   r2_5  <= 0.5 * k5  ;
r6  = r1_6  + r2_6  ;   r1_6  <= 1   ;   r1_6  >= 1   * k6  ;   r2_6  <= 1   * k6  ;
r8  = r1_8  + r2_8  ;   r1_8  <= 0.5 ;   r1_8  >= 0.5 * k8  ;   r2_8  <= 0.5 * k8  ;
r10 = r1_10 + r2_10 ;   r1_10 <= 0.5 ;   r1_10 >= 0.5 * k10 ;   r2_10 <= 0.5 * k10 ;
r11 = r1_11 + r2_11 ;   r1_11 <= 1   ;   r1_11 >= 1   * k11 ;   r2_11 <= 1   * k11 ;

// redução do tipo a (kai) ou b (kbi)
r7 = ka7 + 2 kb7; ka7 + kb7 <= 1;
r9 = ka9 + 2 kb9; ka9 + kb9 <= 1;

bin k1, k3, k4, k5, k6, ka7, kb7, k8, ka9, kb9, k10, k11;
```

2.6 Apresente o ficheiro de output produzido pelo programa

2.6.1 solucao.txt

```

Value of objective function: 570.00000000

Actual values of the variables:
r1_1                            0
r2_1                            0
r1_3                            0
r2_3                            0
r1_4                            0
r1_5                            0
r2_5                            0
r1_6                            1
r2_6                            1
ka7                             1
kb7                             0
r1_8                            0
r2_8                            0
ka9                             0
kb9                             0
r1_10                           0
r2_10                           0
r1_11                           0
r2_11                           0
tf                             23
t1                              6
r1                              0
t3                             23
r3                              0
t4                             17
t5                             21
r5                              0
t6                              3
r6                              2
t7                              8
r7                              1
r4                              0
t8                             15
r8                              0
t9                             20
r9                              0
t10                            11
r10                             0
t11                            18
r11                             0
k1                              0
k3                              0
r2_4                            0
k4                              0
k5                              0
k6                              1
k8                              0
k10                             0
k11                             0
```

2.7 Apresente o plano de execução (diagrama de Gantt) do projecto representando as actividades com as durações que elas têm após a respectiva redução

2.7.1 Análise do ficheiro output

- Reduziu-se 1 U.T. (máximo) do tipo 1 à atividade 6.

```
r1_6                                1
```

- Reduziu-se 1 U.T. (máximo) do tipo 2 à atividade 6.

```
r2_6                                1
```

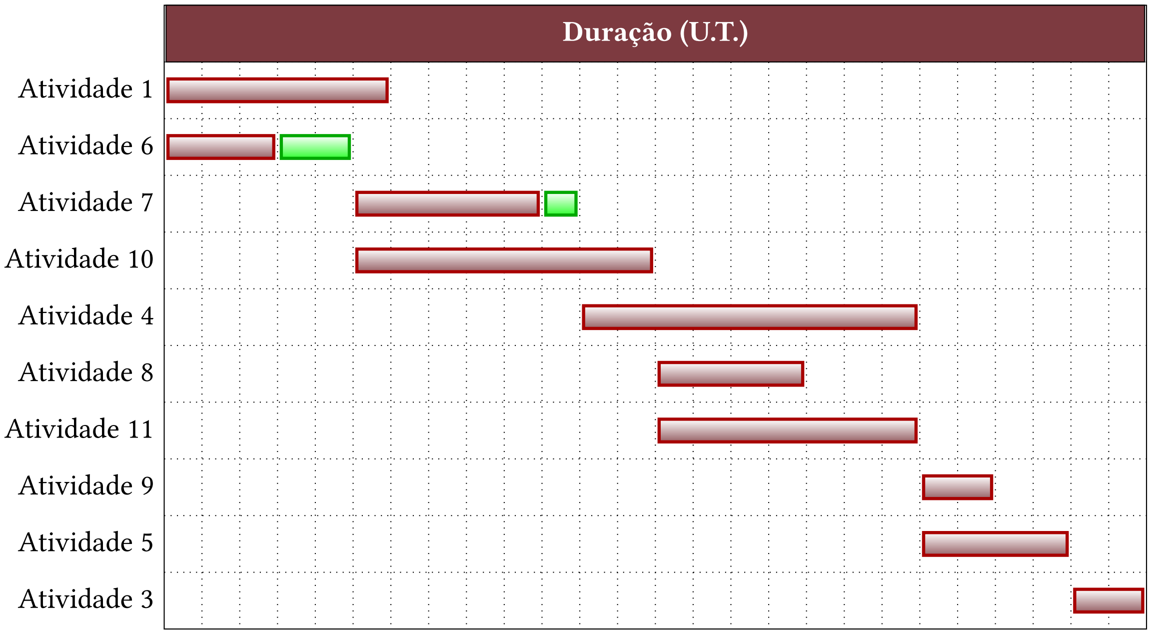
- Escolheu-se a redução do tipo *a* à atividade 7, reduzindo-se 1 U.T.

```
ka7                                 1
```

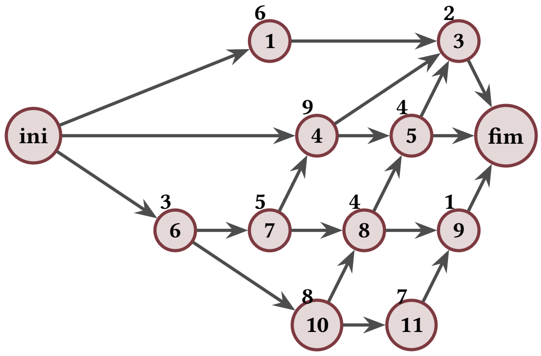
- A redução total ao projeto foi de 3 U.T. (26 - 23 = 3), tal como era desejado.

```
tf                                 23
```

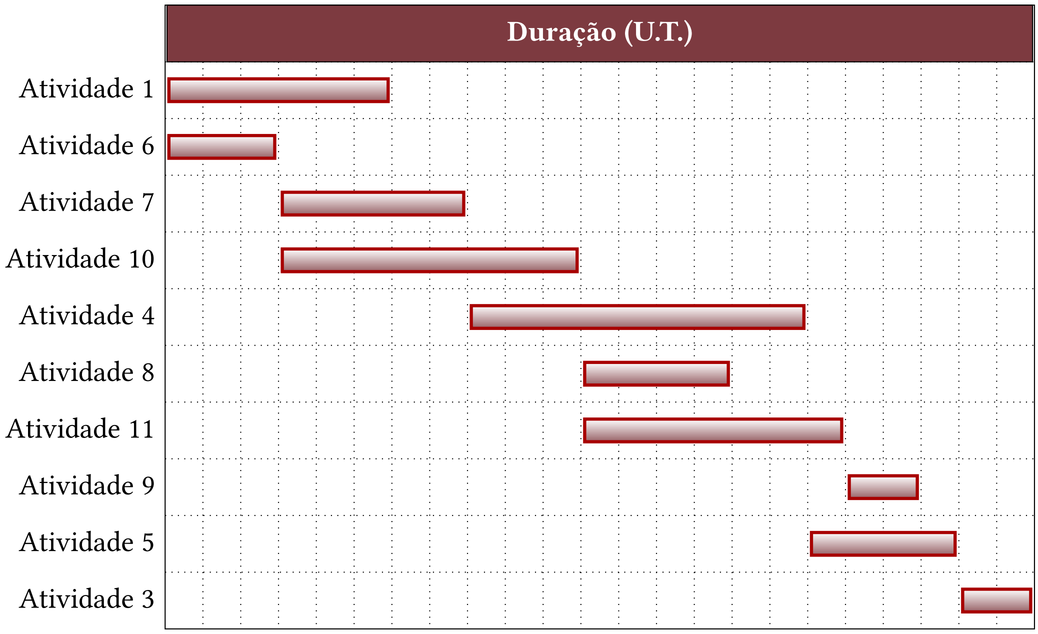
2.7.2 Diagrama de Gantt (Antes da Redução)



2.7.3 Rede da solução ótima



2.7.4 Diagrama de Gantt (Após a Redução)



2.8 Verifique que o custo da solução está correcto

2.8.1 Verificação das restrições

Redução mínima no tempo para concluir o projeto

$$23 = 23 \quad \checkmark$$

Relações de Precedência

$$\begin{aligned}
\textbf{arco_i_1:}&\ 6 \ \geq \quad -0+6 \quad \checkmark\\
\textbf{arco_1_3:}&\ 23 \geq 6 \ \ -0+2 \quad \checkmark\\
\textbf{arco_3_f:}&\ 23 \geq 23 \quad \checkmark\\
\textbf{arco_4_3:}&\ 23 \geq 17-0+2 \quad \checkmark\\
\textbf{arco_4_5:}&\ 21 \geq 17-0+4 \quad \checkmark\\
\textbf{arco_5_3:}&\ 23 \geq 21-0+2 \quad \checkmark\\
\textbf{arco_5_f:}&\ 23 \geq 21 \quad \checkmark\\
\textbf{arco_i_6:}&\ 3 \ \geq \quad -2+5 \quad \checkmark\\
\textbf{arco_6_7:}&\ 8 \ \geq 3 \ \ -1+6 \quad \checkmark\\
\textbf{arco_7_4:}&\ 17 \geq 8 \ \ -0+9 \quad \checkmark\\
\textbf{arco_7_8:}&\ 15 \geq 8 \ \ -0+4 \quad \checkmark\\
\textbf{arco_8_9:}&\ 20 \geq 15-0+2 \quad \checkmark\\
\textbf{arco_9_f:}&\ 23 \geq 20 \quad \checkmark\\
\textbf{arco_6_10:}&\ 11 \geq 3 \ \ -0+8 \quad \checkmark\\
\textbf{arco_10_8:}&\ 15 \geq 11-0+4 \quad \checkmark\\
\textbf{arco_10_11:}&\ 18 \geq 11-0+7 \quad \checkmark\\
\textbf{arco_11_9:}&\ 20 \geq 18-0+2 \quad \checkmark
\end{aligned}$$

Reduzir (opção 1 e opção 2) no máximo

$$\begin{array}{llll}
0=0+0\,;\checkmark & 0\leq 1\ ;\checkmark & 0\geq 1\times 0\ ;\checkmark & 0\leq 1\times 0\ ;\checkmark\\
0=0+0\,;\checkmark & 0\leq 0.5\,;\checkmark & 0\geq 0.5\times 0\,;\checkmark & 0\leq 0.5\times 0\,;\checkmark\\
0=0+0\,;\checkmark & 0\leq 2\ ;\checkmark & 0\geq 2\times 0\ ;\checkmark & 0\leq 1\times 0\ ;\checkmark\\
0=0+0\,;\checkmark & 0\leq 0.5\,;\checkmark & 0\geq 0.5\times 0\,;\checkmark & 0\leq 0.5\times 0\,;\checkmark\\
2=1+1\,;\checkmark & 1\leq 1\ ;\checkmark & 1\geq 1\times 1\ ;\checkmark & 1\leq 1\times 1\ ;\checkmark\\
0=0+0\,;\checkmark & 0\leq 0.5\,;\checkmark & 0\geq 0.5\times 0\,;\checkmark & 0\leq 0.5\times 0\,;\checkmark\\
0=0+0\,;\checkmark & 0\leq 0.5\,;\checkmark & 0\geq 0.5\times 0\,;\checkmark & 0\leq 0.5\times 0\,;\checkmark\\
0=0+0\,;\checkmark & 0\leq 1\ ;\checkmark & 0\geq 1\times 0\ ;\checkmark & 0\leq 1\times 0\ ;\checkmark
\end{array}$$

Opção a ou b (xor)

$$\begin{array}{ll}
1=1+2\times 0\,;\checkmark & 1+0\leq 1\,\checkmark\\
0=0+2\times 0\,;\checkmark & 0+0\leq 1\,\checkmark
\end{array}$$

2.8.2 Verificação do custo

Seguindo as informações dadas pelo ficheiro output e pelas tabelas de custos,

$$
\begin{aligned}
\text{custo total adicional} = &\\
&600\ (0) + 300\ (0) + \\
&200\ (0) + 500\ (0) + \\
&800\ (0) + 400\ (0) + \\
&1600\ (0) + 800\ (0) + \\
&\mathbf{180}\ (\mathbf{1}) + \mathbf{90}\ (\mathbf{1}) + \\
&\mathbf{300}\ (\mathbf{1}) + 1100\ (0) + \\
&300\ (0) + 100\ (0) + \\
&200\ (0) + 400\ (0) + \\
&1000\ (0) + 500\ (0) + \\
&600\ (0) + 300\ (0)
\end{aligned}
$$

$= 180 + 90 + 300 = 570$, que é o mesmo valor que foi obtido no ficheiro output.