Modelos Determinísticos de Investigação Operacional

Trabalho prático nº3

8 de janeiro de 2020

Grupo nº 11

Filipa Alves dos Santos (A83631)

Hugo André Coelho Cardoso (A85006)

João da Cunha e Costa (A84775)

Válter Ferreira Picas Carvalho (A84464)

Mestrado Integrado em Engenharia Informática

Universidade do Minho

Índice de conteúdos

1. Introdução

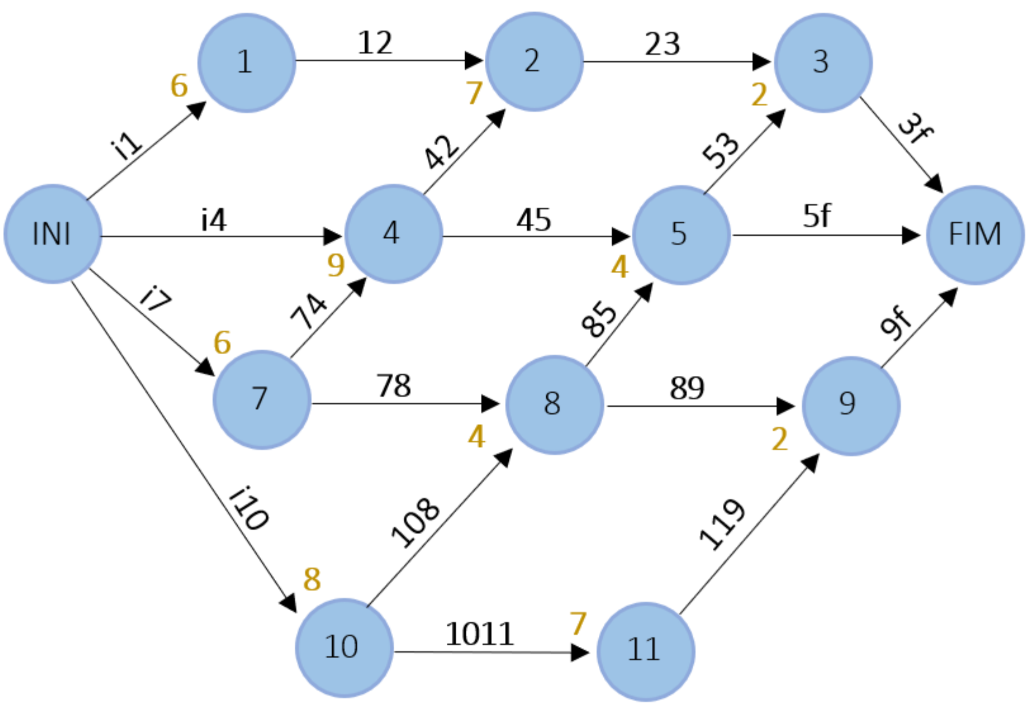
Neste trabalho prático, era pretendido trabalhar com o projeto dado no enunciado de maneira diferente nas 3 partes, de modo a estudar o método do caminho crítico. Assim, na parte 0 é apenas pedido que personalizemos a rede de acordo com dados do nosso grupo e, de seguida, achar o novo caminho crítico do novo projeto, bem como o diagrama Gantt deste. Já na parte I, era pedido que refizéssemos o problema, mas agora considerando que 3 das atividades, anteriormente simultâneas, não podiam ser realizadas ao mesmo tempo devido a limitações de equipamento. Simplesmente inserimos mais variáveis e restrições relativas a esse desafio ao modelo da parte 0 e obtivemos os resultados pretendidos. Já na parte 2, era para considerar a existência de reduções de custo 1 e 2 e tentar reduzir o tempo em 3 unidades com estas, sabendo que a 2ª só podia ser realizada se a 1ª fosse realizada no seu máximo valor. De modo parecido à parte 1, criamos também variáveis binárias e fizemos restrições com estas para tentar resolver o problema.

2. Desenvolvimento

2.1. Parte 0

2.1.1. Rede Atualizada

Considerando que 85006 é o maior número de aluno dos membros do nosso grupo, determinou-se que o as atividades a eliminar eram a 0 e a 6. Assim, a rede que representa o nosso projeto é a seguinte:



Atividade	Duração	Precedências
1	6	-
2	7	1,4
3	2	2,5
4	9	-
5	4	4,8
6	6	-
7	4	7,10
9	2	8,11
10	8	-
11	7	10

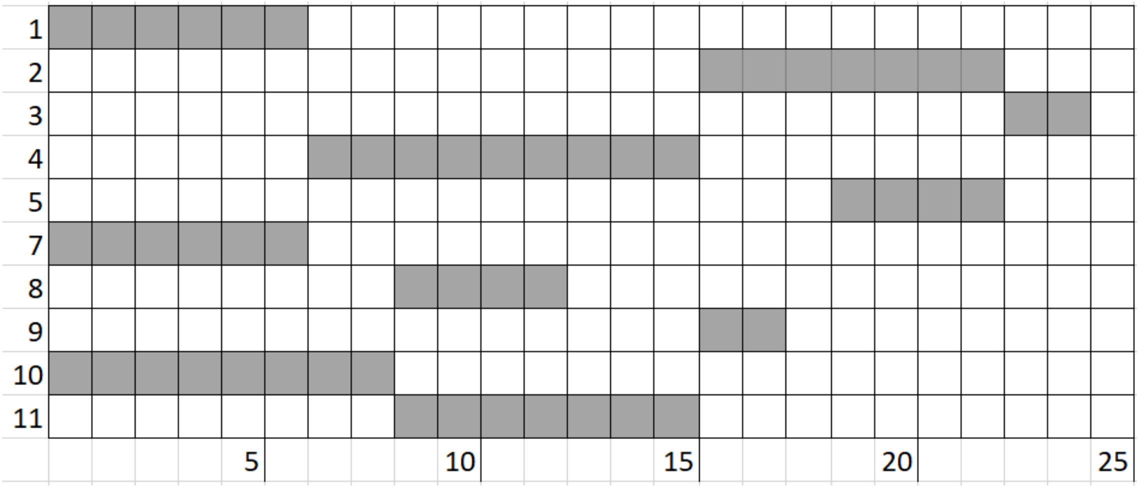
2.1.2. Input LPSolve

```
/* Função Objectivo */
min: tf;


/* Restrições */
arco_i1:   t1 >= ti + 0;
arco_12:   t2 >= t1 + 6;
arco_23:   t3 >= t2 + 7;
arco_3f:   tf >= t3 + 2;
arco_i4:   t4 >= ti + 0;
arco_42:   t2 >= t4 + 9;
arco_45:   t5 >= t4 + 9;
arco_53:   t3 >= t5 + 4;
arco_5f:   tf >= t5 + 4;
arco_i7:   t7 >= ti + 0;
arco_74:   t4 >= t7 + 6;
arco_78:   t8 >= t7 + 6;
arco_85:   t5 >= t8 + 4;
arco_89:   t9 >= t8 + 4;
arco_9f:   tf >= t9 + 2;
arco_i10:  t10 >= ti + 0;
arco_108:  t8 >= t10 + 8;
arco_1011: t11 >= t10 + 8;
arco_119:  t9 >= t11 + 7;
```

As restrições que introduzimos no programa são restrições disjuntivas: garantem que o instante do fim da execução de uma tarefa é anterior ao instante de início de outra.

Através do LPSolve, determinamos que a duração do projeto, isto é, o menor tempo possível para completar todas as atividades, é 24 unidades de tempo. O caminho crítico responsável por tal duração é 7-4-2-3, que é visível no diagrama apresentado no tópico seguinte.

2.1.3. Diagrama de Gantt



2.2. Parte I

2.2.1. Reformulação do problema

Para esta pergunta, escolhemos as seguintes atividades 1, 7 e 10, que se encontram em paralelo e onde uma delas, a atividade 7, pertence ao caminho crítico. Como agora existe apenas um equipamento para realizar as 3 atividades, vão ter de ser executadas sequencialmente. Como tal, vamos exigir ao LPSolve que teste as diferentes precedências possíveis entre elas, em busca da solução que resulte na menor duração possível do projeto.

Criamos variáveis binárias para expressar dicotomias, impedindo assim as duas atividades que dizem respeito à respetiva variável de ocupar simultaneamente a máquina:

- bin y17, y110, y710;

De seguida, criamos restrições de não-simultaneidade entre as várias atividades, como exemplificamos abaixo, que, tal como o nome indica, impedem que as respetivas atividades se realizem simultaneamente. As restrições entre as restantes variáveis são análogas:

- Atividades 1 e 10:

  t1 + M * y110 >= t10 + 8;

  t10 + M * (1 – y110) >= t1 + 6;

Ao testar em LPSolve, assumimos o valor 100 para M e adequamos a sintaxe das restrições ao programa, como é possível observar mais abaixo, no ficheiro de input.

A função objetivo mantém-se porque, citando o enunciado, “o objectivo continua a ser realizar o projecto na menor duração possível”.

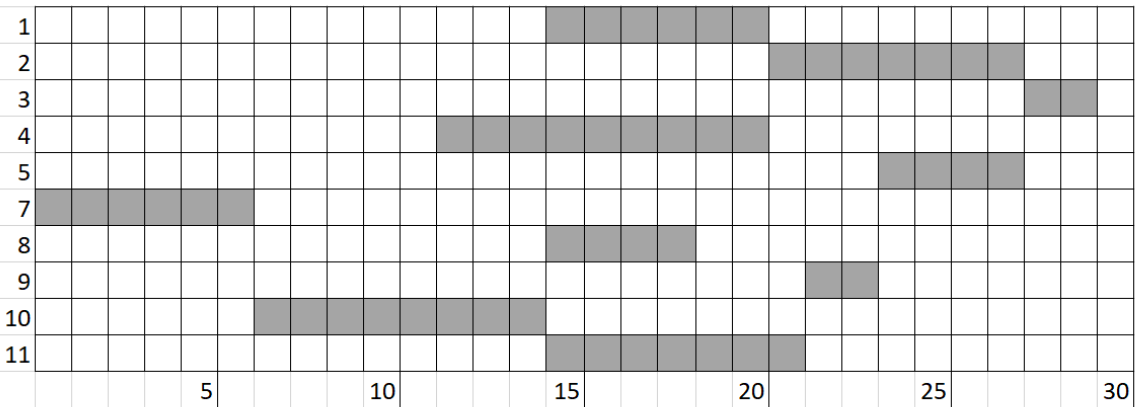
```
/* Função Objetivo */

min: tf;

/* Restrições */

arco_i1:   t1 >= ti + 0;

arco_12:   t2 >= t1 + 6;

arco_23:   t3 >= t2 + 7;

arco_3f:   tf >= t3 + 2;

arco_i4:   t4 >= ti + 0;

arco_42:   t2 >= t4 + 9;

arco_45:   t5 >= t4 + 9;

arco_53:   t3 >= t5 + 4;

arco_5f:   tf >= t5 + 4;

arco_i7:   t7 >= ti + 0;

arco_74:   t4 >= t7 + 6;

arco_78:   t8 >= t7 + 6;

arco_85:   t5 >= t8 + 4;

arco_89:   t9 >= t8 + 4;

arco_9f:   tf >= t9 + 2;

arco_i10:  t10 >= ti + 0;

arco_108:  t8 >= t10 + 8;

arco_1011: t11 >= t10 + 8;

arco_119:  t9 >= t11 + 7;


/* Atividades 1, 7 e 10 sequencialmente */

t1 + 100 y17 >= t7 + 6;

t7 + 100 - 100 y17 >= t1 + 6;

t1 + 100 y110 >= t10 + 8;
```

```
t10 + 100 - 100 y110 >= t1 + 6;

t10 + 100 y710 >= t7 + 6;
t7 + 100 - 100 y710 >= t10 + 8;

bin y17, y110, y710;
```

2.2.3. Output LPSolve

Variables	MILP Feasible	MILP Better	MILP ...	result
	38	30	29	29
tf	38	30	29	29
t1	0	0	14	14
ti	0	0	0	0
t2	29	21	20	20
t3	36	28	27	27
t4	20	12	11	11
t5	32	24	23	23
t7	14	6,00000000000002	0	0
t8	20	20	14	14
t9	24	27	21	21
t10	6,00000000000002	12	6	6
t11	17	20	14	14
y17	1	1	0	0
y110	1	1	0	0
y710	1	0	0	0

2.2.4. Diagrama de Gantt



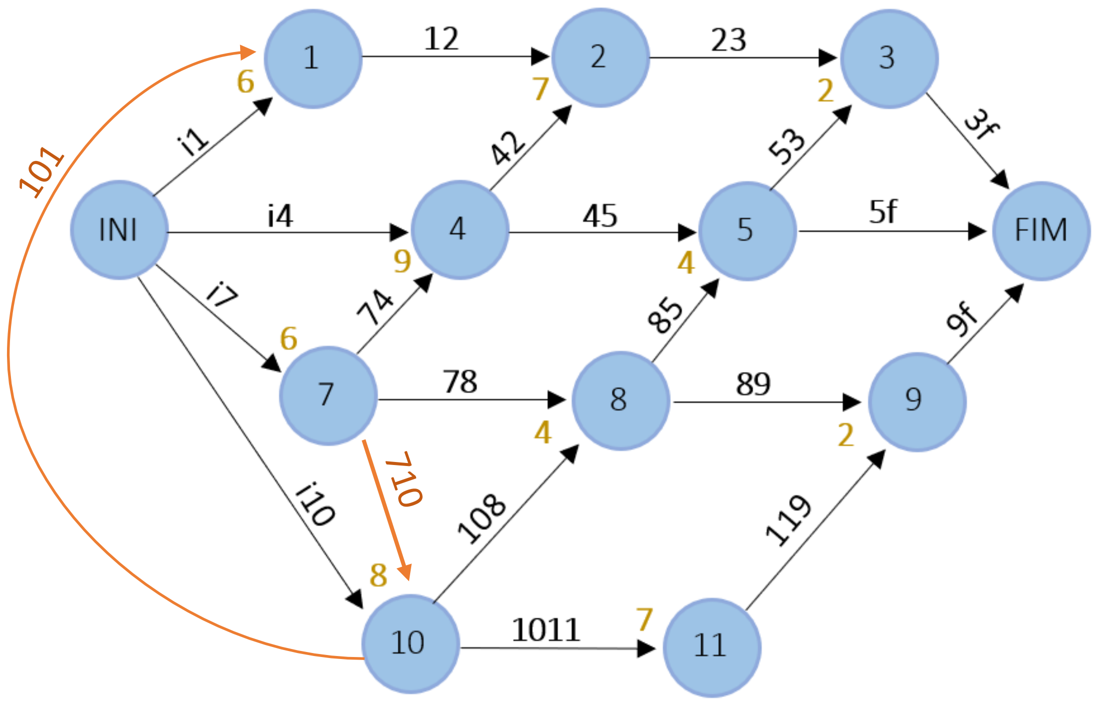
Substituindo os valores das variáveis nas novas restrições, vemos que estas se verificam:

- t1 + 100 y17 >= t7 + 6 -----> 14 + 100 * 0 >= 0 + 6 -----> 14 >= 6
- t7 + 100 – 100 y17 >= t1 + 6 -----> 0 + 100 – 100 * 0 >= 14 + 6 -----> 100 >= 20
- t1 + 100 y110 >= t10 + 8 -----> 14 + 100 * 0 >= 6 + 8 -----> 14 >= 14
- t10 + 100 - 100 y110 >= t1 + 6 -----> 6 + 100 – 100 * 0 >= 14 + 6 -----> 106 >= 20
- t10 + 100 y710 >= t7 + 6 -----> 6 + 100 * 0 >= 0 + 6 -----> 6 >= 6
- t7 + 100 - 100 y710 >= t10 + 8 -----> 0 + 100 – 100 * 0 >= 6 + 8 -----> 100 >= 14

Nova duração do projeto: 29 unidades de tempo

Novo caminho crítico: 7-10-1-2-3

Na prática, isto equivale a criar um arco de 7 para 10 e outro de 10 para 1 no grafo. Logo, a rede que representa a nova solução é a seguinte:



2.3. Parte II

2.3.1. Reformulação do problema

Primeiramente, criamos as variáveis ri_1 e ri_2, para todas as atividades i ∈ {1,2,3,4,5,7,8,9,10,11}, que correspondem respetivamente à redução de custo 1 e à redução de custo 2 da atividade i. A nova função objetivo vai ser em função destas novas variáveis, sendo que queremos minimizar apenas o custo suplementar, obtendo obrigatoriamente um tempo de execução 3 unidades mais baixo que o obtido na parte 0 (tf ≤ 21). Assim, a função vai ter este aspeto:

$$min \sum_{i \in \{1,2,3,4,5,7,8,9,10,11\}} c1_{unitário} \times ri\_1 + c2_{unitário} \times ri\_2$$

Além disto, também restringimos todos as reduções aos seus máximos (por exemplo, para a redução de custo 1 da atividade 1: r1_1 ≤ 1. Para mantermos a precedência dos arcos, criamos uma restrição para cada atividade que indica que o instante em que uma atividade que precede outra, com as reduções aplicadas, tem de ser sempre inferior ou igual ao desta. Exemplificando, com a restrição t2 ≥ t1 – r1_1 – r1_2 + 6 quer dizer que a atividade 2 é sempre posterior à atividade 1, agora com as reduções de tempo r1_1 e r1_2.

No enunciado é também indicado que a redução de preço 2 só pode ser aplicada se a redução de preço 1 da mesma atividade estiver no seu máximo. De modo a resolver este problema, começamos por criar uma variável binária para cada atividade que determina se a redução de preço 1 é máxima ou não: bin y1, y2, y3, y4, y5, y7, y8, y9, y10, y11. Para cada uma destas variáveis, caso a 1ª redução de preço da atividade correspondente seja máxima, é lhe atribuído o valor 0 e, caso contrário, é lhe atribuído o valor 1. Para garantir que este nosso raciocínio é alcançado na prática, criamos 2 restrições que forçam este comportamento:

$$\forall i_{\in\{1,2,3,4,5,7,8,9,10,11\}}\ ri\_1 + M \times yi \geq max_{ri\_1}, M \to \infty$$

e

$$\forall i_{\in\{1,2,3,4,5,7,8,9,10,11\}}\ ri\_1 + M \times yi < max_{ri\_1} + M, M \to \infty$$

(no LPSolve consideramos o M = 100, sendo que só tem de ser um número muito grande)

Dando um exemplo, se considerarmos a atividade 2 e substituirmos por ri_1 = 1, que é a máxima redução 1 possível para essa atividade, y obrigatoriamente será 0 e, se substituirmos por qual quer valor inferior a esse, y será sempre 1. Com estas restrições determinadas, definimos uma última, também aplicada a cada atividade, que só deixa que a redução de custo 2 seja efetuada se a 1 estiver no seu máximo, ou seja, no nosso caso, se y = 0. Eis a sua fórmula:

$$\forall i_{\in\{1,2,3,4,5,7,8,9,10,11\}}\ ri\_2 \leq max_{rc2}(1 - yi)$$

Pegando no exemplo anterior, quando o y = 0 (ri_1 máximo), rc2 vai ser ≤ max_{rc2} e quando y = 1 (ri_1 menor que o máximo), rc2 vai ser ≤ 0, confirmando o nosso raciocínio.

2.3.2. Input LPSolve

```
/* Função Objetivo */
min:  600 r1_1 + 300 r1_2 + 1000 r2_1 + 500 r2_2 + 200 r3_1 +
100 r3_2 + 800 r4_1 + 400 r4_2 + 1600 r5_1 + 800 r5_2 + 200 r8_1
+ 100 r8_2 + 1000 r10_1 + 500 r10_2 + 600 r11_1 + 300 r11_2;
```

```
/* Restrições */
tf <= 21;

arco_i1:   t1 >= ti + 0;
arco_12:   t2 >= t1 - r1_1 - r1_2 + 6;
arco_23:   t3 >= t2 - r2_1 - r2_2 + 7;
arco_3f:   tf >= t3 - r3_1 - r3_2 + 2;
arco_i4:   t4 >= ti + 0;
arco_42:   t2 >= t4 - r4_1 - r4_2 + 9;
arco_45:   t5 >= t4 - r4_1 - r4_2 + 9;
arco_53:   t3 >= t5 - r5_1 - r5_2 + 4;
arco_5f:   tf >= t5 - r5_1 - r5_2 + 4;
arco_i7:   t7 >= ti + 0;
arco_74:   t4 >= t7 - r7_1 - r7_2 + 6;
arco_78:   t8 >= t7 - r7_1 - r7_2 + 6;
arco_85:   t5 >= t8 - r8_1 - r8_2 + 4;
arco_89:   t9 >= t8 - r8_1 - r8_2 + 4;
arco_9f:   tf >= t9 - r9_1 - r9_2 + 2;
arco_i10:  t10 >= ti + 0;
arco_108:  t8 >= t10 - r10_1 - r10_2 + 8;
arco_1011: t11 >= t10 - r10_1 - r10_2 + 8;
arco_119:  t9 >= t11 - r11_1 - r11_2 + 7;

r1_1 <= 1;
r2_1 <= 3;
r3_1 <= 0.5;
r4_1 <= 2;
r5_1 <= 0.5;
r7_1 <= 0;
r8_1 <= 0.5;
r9_1 <= 0;
r10_1 <= 0.5;
r11_1 <= 1;
```

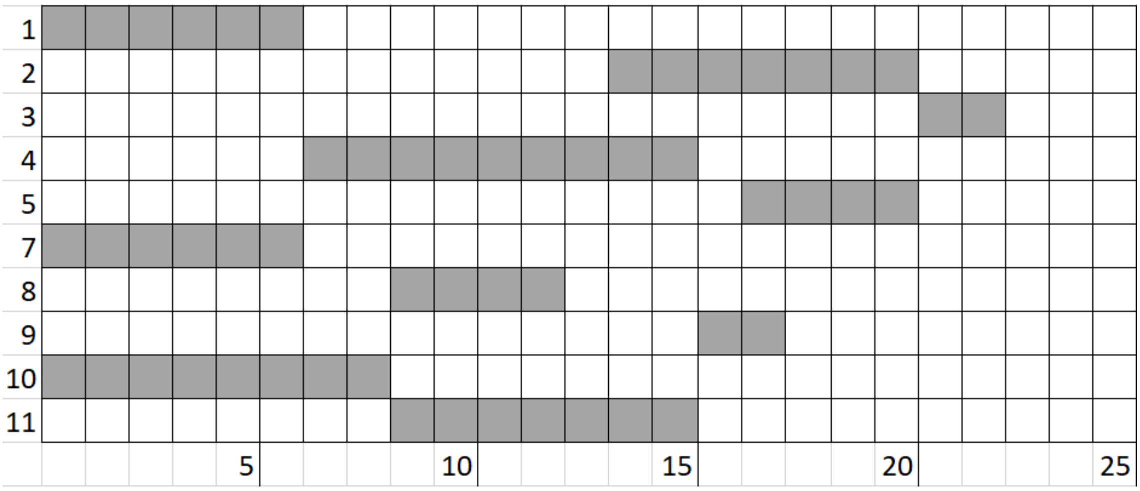
```
r1_1 + 100 y1 >= 1;
r1_1 + 100 y1 < 1 + 100;
r1_2 <= 1-y1;

r2_1 + 100 y2 >= 3;
r2_1 + 100 y2 < 3 + 100;
r2_2 <= 1 - y2;

r3_1 + 100 y3 >= 0.5;
r3_1 + 100 y3 < 0.5 + 100;
r3_2 <= 0.5 - 0.5 y3;

r4_1 + 100 y4 >= 2;
r4_1 + 100 y4 < 2 + 100;
r4_2 <= 1 - y4;

r5_1 + 100 y5 >= 0.5;
r5_1 + 100 y5 < 0.5 + 100;
r5_2 <= 0.5 -0.5 y5;

r7_1 + 100 y7 >= 0;
r7_1 + 100 y7 < 0 + 100;
r7_2 <= 0;

r8_1 + 100 y8 >= 0.5;
r8_1 + 100 y8 < 0.5 + 100;
r8_2 <= 0.5 -0.5 y8;

r9_1 + 100 y9 >= 0;
r9_1 + 100 y9 < 0 + 100;
r9_2 <= 0;

r10_1 + 100 y10 >= 0.5;
```

```
r10_1 + 100 y10 < 0.5 + 100;
r10_2 <= 0.5 -0.5 y10;

r11_1 + 100 y11 >= 1;
r11_1 + 100 y11 < 1 + 100;
r11_2 <= 1-y11;

bin y1, y2, y3, y4, y5, y7, y8, y9, y10, y11;
```

2.3.3. Output LPSolve

Variables	MILP ...	result
	1750	1750
r1_1	0	0
r1_2	0	0
r2_1	0	0
r2_2	0	0
r3_1	0,5	0,5
r3_2	0,5	0,5
r4_1	2	2
r4_2	0	0
r5_1	0	0
r5_2	0	0
r8_1	0	0
r8_2	0	0
r10_1	0	0
r10_2	0	0
r11_1	0	0
r11_2	0	0
tf	21	21
t1	0	0
ti	0	0
t2	13	13
t3	20	20
t4	6	6
t5	16	16
t7	0	0
r7_1	0	0
r7_2	0	0
t8	8	8
t9	15	15
r9_1	0	0
r9_2	0	0
t10	0	0
t11	8	8
y1	1	1
y2	1	1
y3	0	0
y4	1	1
y5	1	1
y7	0	0
y8	1	1
y9	0	0
y10	1	1
y11	1	1

2.3.4. Diagrama de Gantt



Infelizmente, na solução do LPSolve, y4 = 1 e r4_1 = 2 (a primeira redução da atividade 4 = 2, que é o máximo). Isto vai contra as restrições estabelecidas e não conseguimos entender nem resolver este erro. Pensamos que possa ser de um arredondamento interno do LPSolve, por exemplo arredondando o valor de r4_1 de 1.9 para 2, o que leva o programa a pensar que a redução é máxima quando, de facto, não o é. Por este motivo, não conseguimos obter um Diagrama de Gantt correto nem um caminho crítico possível.

3. Conclusão

Neste terceiro e último trabalho, de forma semelhante aos dois trabalhos anteriores, para além de nos ensinar sobre o tema específico, que neste caso foi o método do caminho crítico, também nos desenvolve o raciocínio para resolver problemas do género. Tivemos que descobrir as restrições associadas a cada desafio imposto no enunciado bem como interpretar resultados de modo a certificarmo-nos que fazem sentido no contexto do problema. Porém, tivemos problemas relativamente à parte 2 deste trabalho que não conseguimos resolver.

Em geral, estamos satisfeitos com todos os trabalhos realizados no âmbito desta unidade curricular e concluímos este semestre com um maior conhecimento nesta área devido a estes.