**Relatório Trabalho Prático 2**

Modelos Determinísticos de Investigação Operacional

Mestrado Integrado em Engenharia Informática

Braga, 27 de novembro de 2017

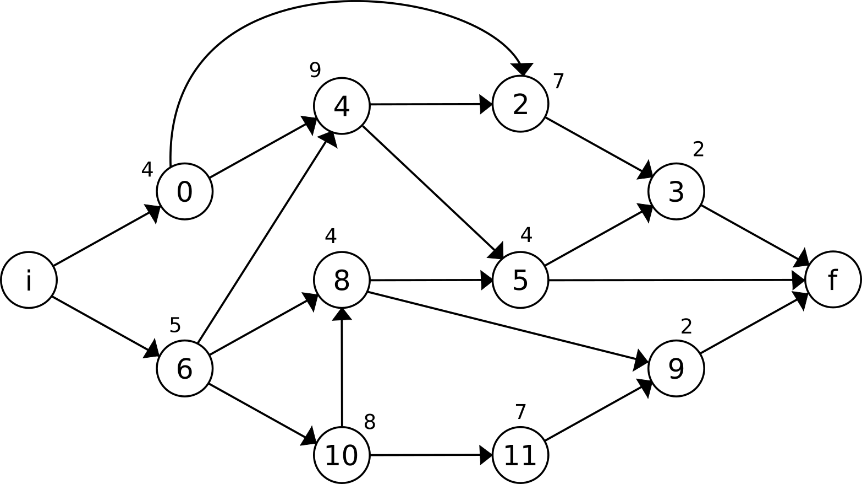
**Grupo 33**

Francisco Oliveira – 78416

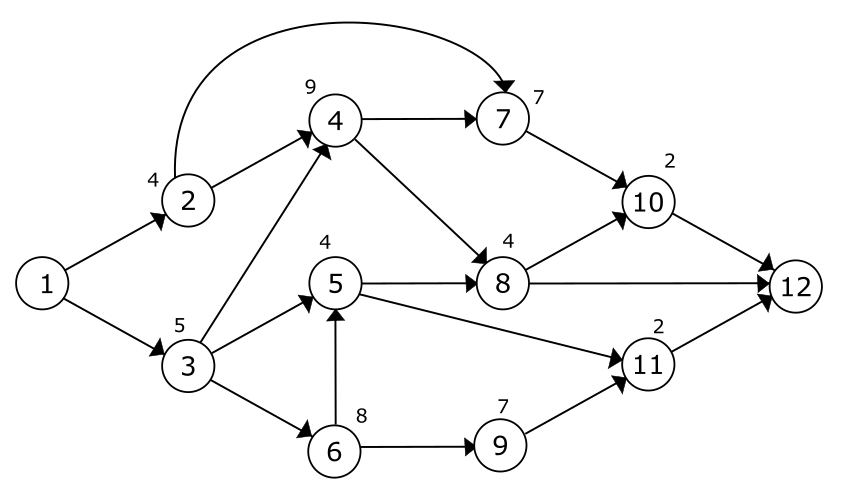
Raul Vilas Boas – 79617

Vitor Peixoto – 79175

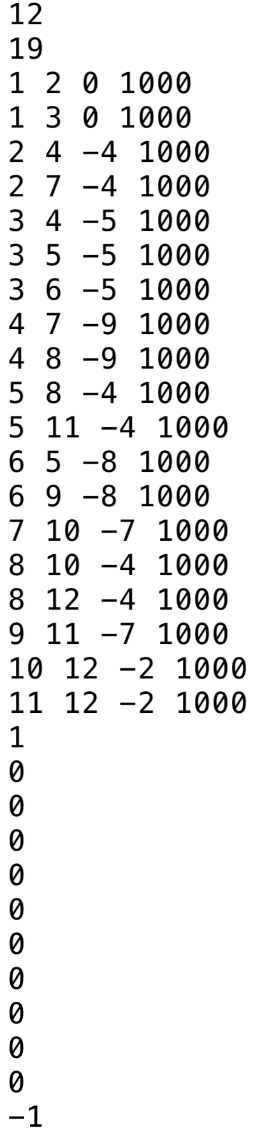
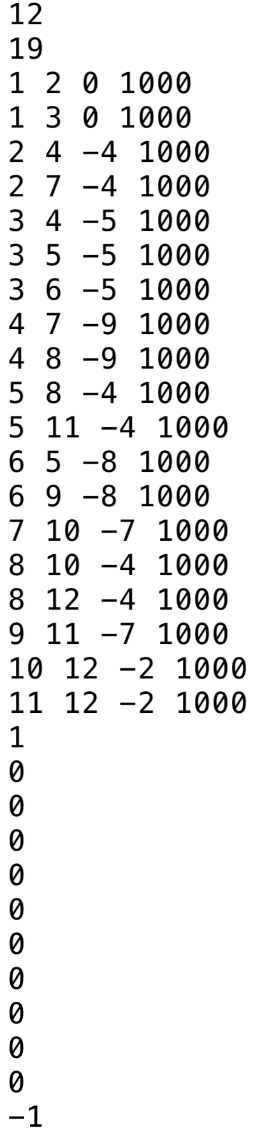
**PARTE I**

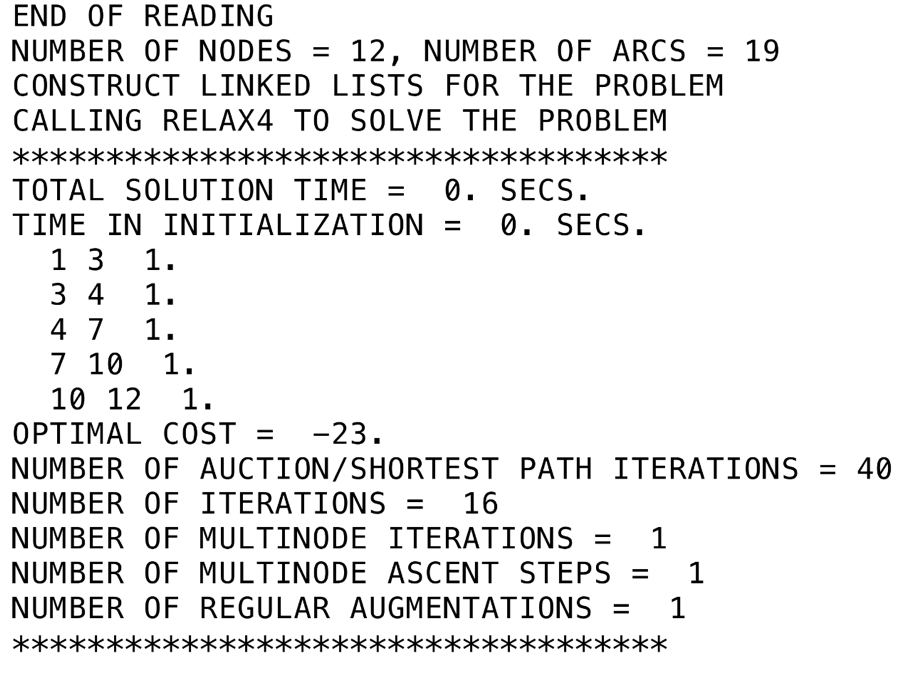
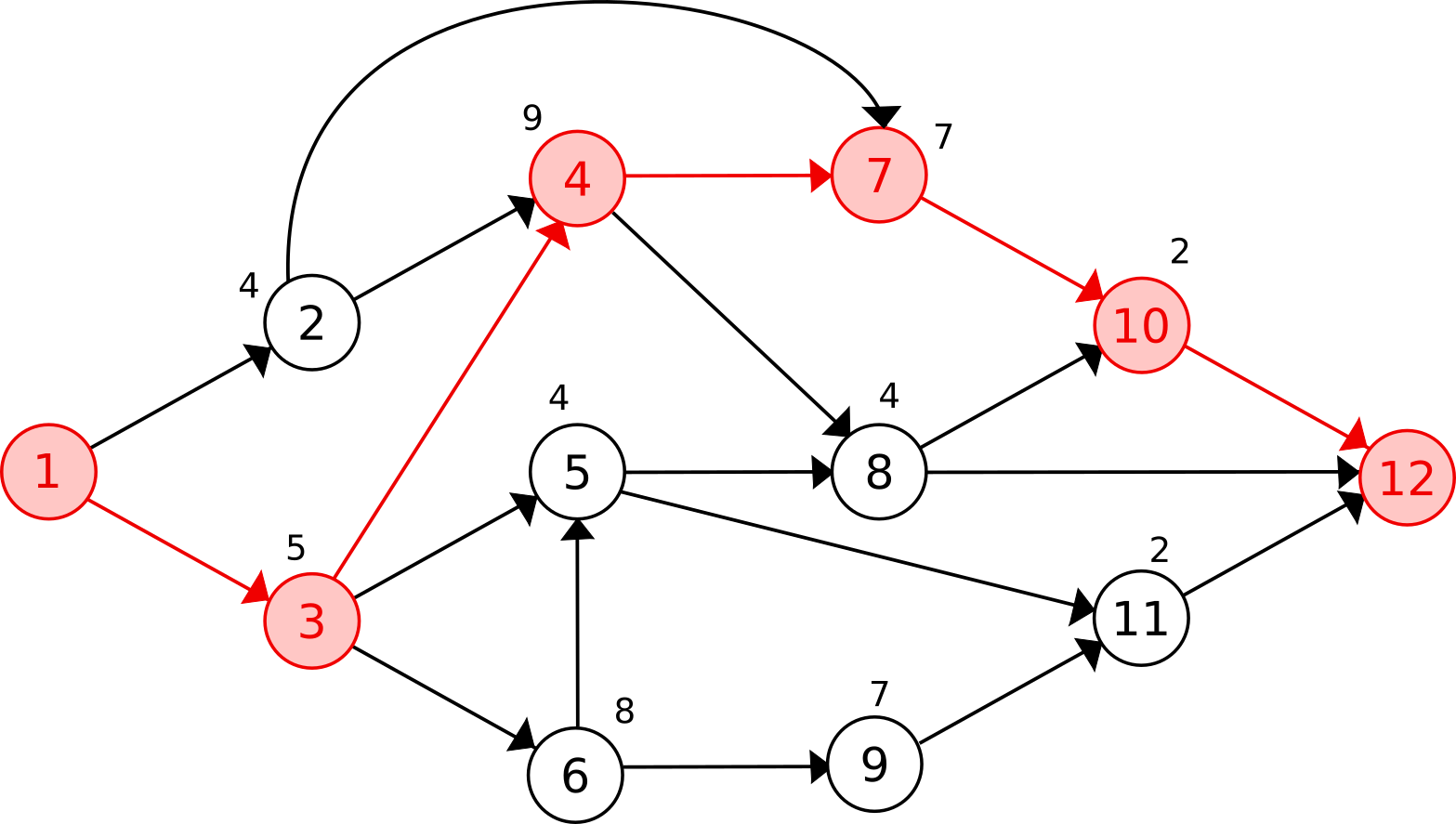
1. Após eliminar as atividades indicadas na secção “Determinação da lista de atividades” com base no número de aluno ABCDE = 79617, obteve-se o seguinte grafo:
2. Assim, com base no grafo orientado com 12 vértices e 19 arcos contruiu-se uma matriz, em que em cada coluna estão representados os seus arcos e em cada linha os seus vértices. Para além disso, também se adicionou uma linha (linha c) com os custos de cada vértice e uma coluna com as ofertas e procuras de cada vértice:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Ai0 | Ai6 | A02 | A04 | A64 | A68 | A610 | A42 | A45 | A85 | A89 | A1011 | A23 | A53 | A5f | A119 | A9f | A108 | A3f | Res |
| i | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | -1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | -1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 0 | -1 | -1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | -1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | -1 | 0 | 0 | 1 | -1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | -1 | 1 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| f | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | -1 | 0 | -1 | -1 |
| c | 0 | 0 | 4 | 4 | 5 | 5 | 5 | 9 | 9 | 4 | 4 | 8 | 7 | 4 | 4 | 7 | 2 | 8 | 2 |  |

1. Visto que o *relax4* apenas aceita números inteiros positivos, foi necessário alterar os números do grafo de forma a ser possível utilizar o programa. Por esse motivo, o grafo utilizado no *relax4* foi o que se encontra representado em baixo:

No input do ficheiro *relax4* tivemos que ter alguns cuidados, isto porque ele está preparado para descobrir o caminho mais curto de um grafo. Como neste problema o objetivo é calcular o caminho crítico do grafo em questão, decidimos colocar os custos a negativo, para assim calcular o pretendido.



1. No output do *relax4* obteve-se um custo ótimo de -23 devido aos custos utilizados serem negativos. Contudo descobre-se que o custo do caminho critico é 23 e qual o seu percurso.
2. O caminho crítico calculado pelo relax foi 1-3-4-7-10-12. Esse caminho é representado na seguinte imagem do grafo:

**PARTE II**

1. Para esta parte já se voltou a usar o grafo com os valores originais dos vértices.

Após a mudança de variável *ri = ti - si* obtivemos o seguinte modelo alternativo.

*Min z: 100 t0 - 100 s0 + 500 t2 - 500 s2 + 100 t3 - 100 s3 + 400 t4 - 400 s4 + 800 t5 - 800 s5 + 90 t6 - 90 s6 + 100 t8 - 100 s8 + 500 t10 - 500 s10 + 300 t11 - 300 s11;*

Como o objetivo do problema nesta parte é diminuir o tempo de execução, encontrado na Parte I, em 3 U.T. e visto que o tempo máximo obtido foi de 23 pretende-se então que o tempo final seja de 20. Para isso, acrescentou-se a seguinte restrição:

*tf = 20*

|  |  |
| --- | --- |
| *t2 ≥ s0 + 4* | *t4 ≥ s6 + 5* |
| *t3 ≥ s2 + 7* | *t5 ≥ s8 + 4* |
| *t0 ≥ ti + 0* | *tf ≥ s9 + 2* |
| *t4 ≥ s0 + 4* | *t8 ≥ s6 + 5* |
| *t2 ≥ s4 + 9* | *t9 ≥ s8 + 4* |
| *t3 ≥ s5 + 4* | *t10 ≥ s6 + 5* |
| *tf ≥ s3 + 2* | *t8 ≥ s10 + 8* |
| *t5 ≥ s4 + 9* | *t9 ≥ s11 + 7* |
| *tf ≥ s5 + 4* | *t11 ≥ s10 + 8* |
| *t6 ≥ ti + 0* | *t4 ≥ s6 + 5* |

Para além disso, é necessário introduzir as restrições necessárias:

Como existe um limite máximo de reduções também é necessário escrever essa restrição, assim como a de que os valores das reduções (*ti - si*) são maiores ou iguais a zero.

*t0 - s0* ≤ *1*

*t2 - s2* ≤ *4*

*t3 - s3* ≤ *1*

*t4 - s4* ≤ *3*

*t5 - s5* ≤ *1*

*t6 - s6* ≤ *2*

*t8 - s8* ≤ *1*

*t9 - s9* ≤ *0*

*t10 - s10* ≤ *1*

*t11 - s11* ≤ *2*

*t0 - s0, t2 - s2, t3 - s3, t4 - s4, t5 - s5, t6 - s6, t8 - s8, t9 - s9, t10 - s10, t11 - s11 ≥ 0*

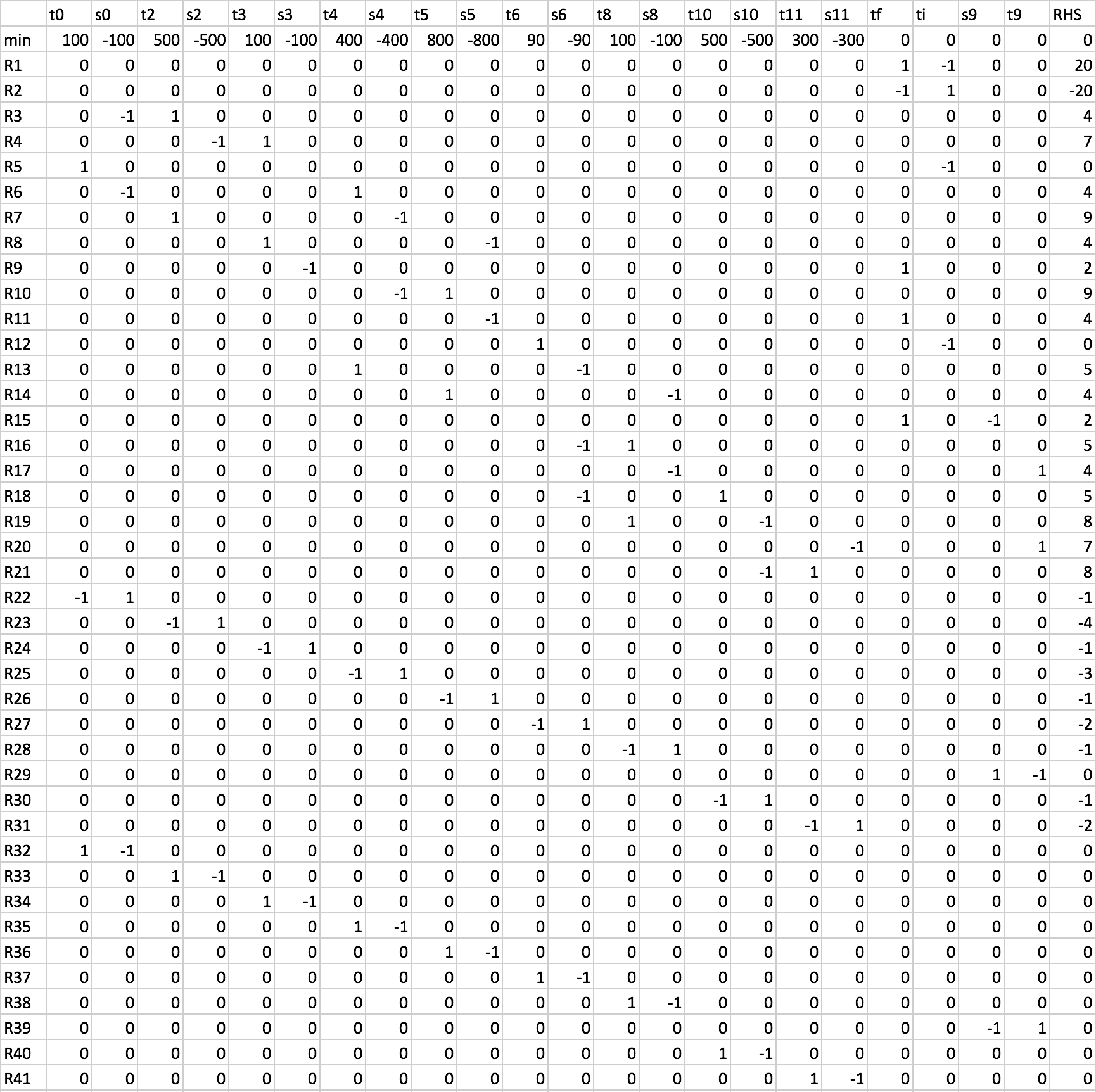
Por último, também é necessário permitir que os valores dos *si* possam ter valores negativos, para isso, utiliza-se o ‘*free*’.

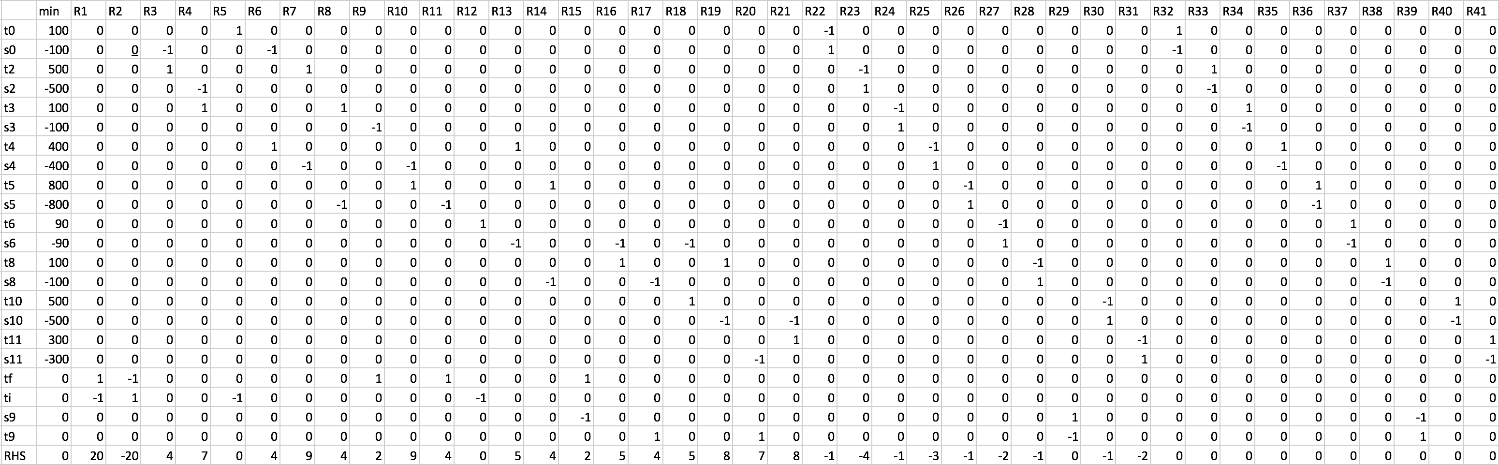
*free s0, s2, s3, s4, s5, s6, s8, s9, s10, s11;*

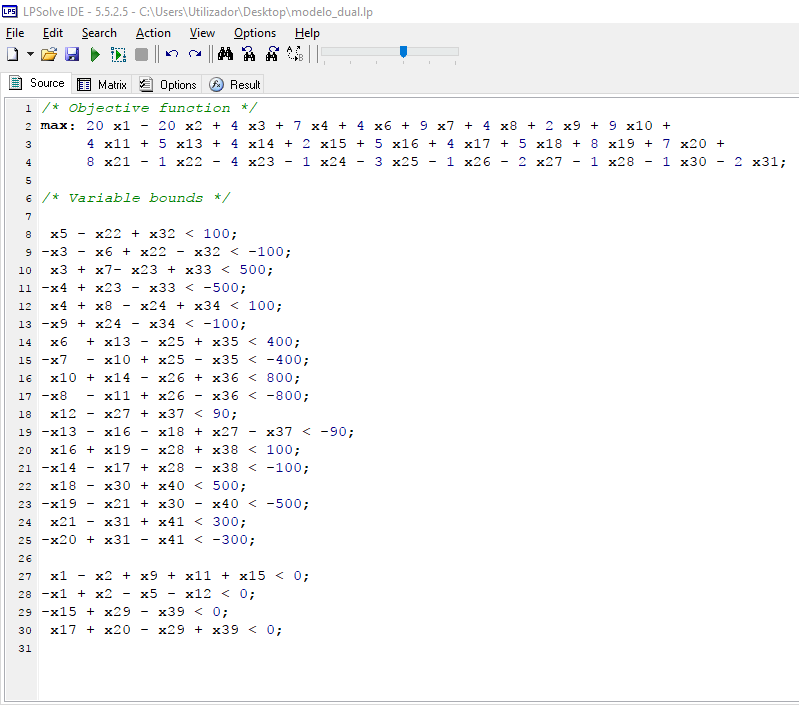
1. **& 4.** Para obter o modelo dual do modelo alternativo é necessário transpor a matriz desse mesmo modelo. Para isso, foi necessário alterar algumas das restrições do modelo, visto que, como era uma minimização todas as restrições têm de ser maiores ou iguais (*≥*).

As restrições alteradas foram:

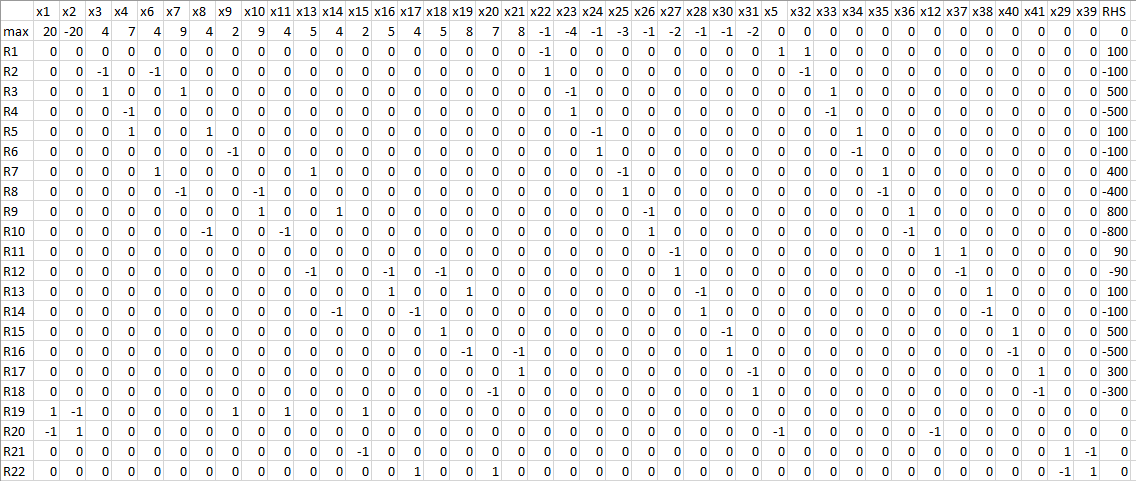
|  |  |  |
| --- | --- | --- |
| *tf* ≤ *20* | *Para* | *tf - ti ≥ 20;*  *-tf + ti ≥ -20;* |
| *t0 - s0* ≤ *1* | *-t0 + s0 ≥ -1* |
| *t2 - s2* ≤ *4* | *-t2 + s2 ≥ -4* |
| *t3 - s3* ≤ *1* | *-t3 + s3 ≥ -1* |
| *t4 - s4* ≤ *3* | *-t4 + s4 ≥ -3* |
| *t5 - s5* ≤ *1* | *-t5 + s5 ≥ -1* |
| *t6 - s6* ≤ *2* | *-t6 + s6 ≥ -2* |
| *t8 - s8* ≤ *1* | *-t8 + s8 ≥ -1* |
| *t9 - s9* ≤ *0* | *-t9 + s9 ≥ 0* |
| *t10 - s10* ≤ *1* | *-t10 + s10 ≥ -1* |
| *t11 - s11* ≤ *2* | *-t11 + s11 ≥ -2* |

Após estas alterações, utilizamos a matriz fornecida pelo *LPSolve* e transpôs--se com a ajuda do *Microsoft Excel*:

Obtendo assim a seguinte matriz:

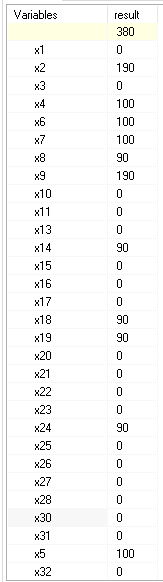
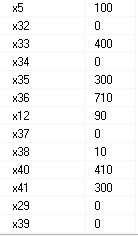
A partir da matriz obteve-se as restrições e a função objetivo. Alterou-se as variáveis para *xi*.

1. A matriz obtida na opção *matrix* do *LPSolve*:



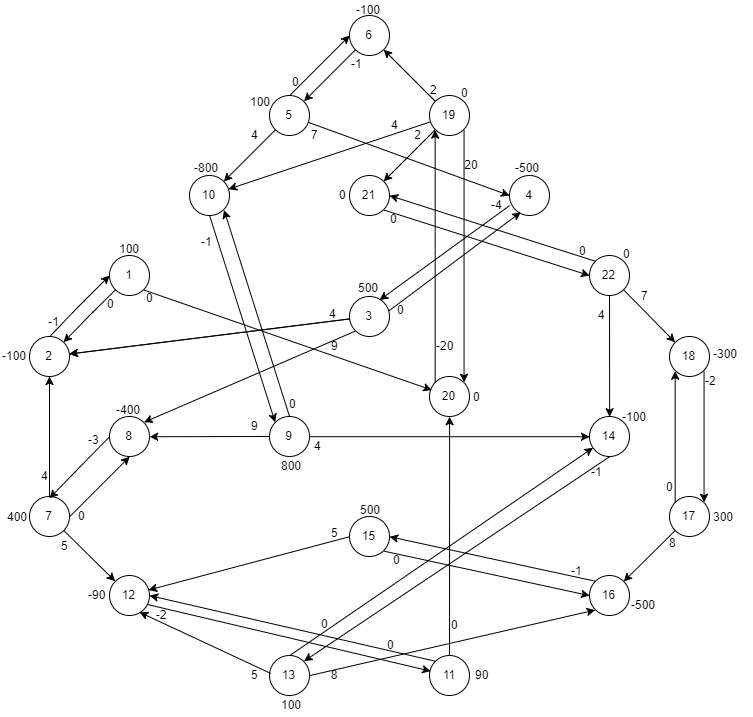
Esta matriz é uma matriz incidência vértice-arco para a rede em análise em que cada coluna corresponde a um vértice e cada linha a um arco do grafo. Para além disso, em cada coluna apenas há dois elementos diferentes de zero, que são os valores dos vértices de origem e destino, 1 origem e -1 destino.

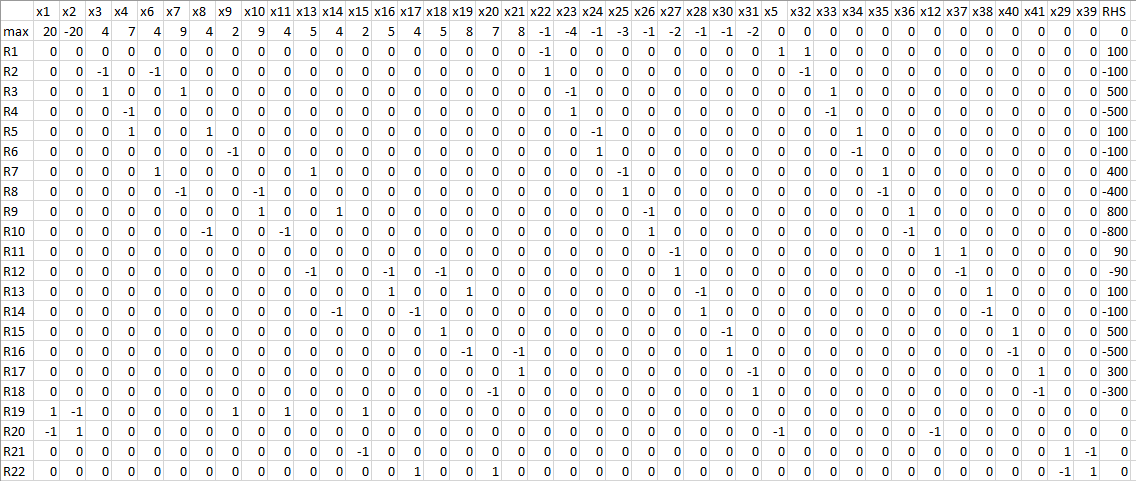
1. Output obtido pelo *LPSolve*:



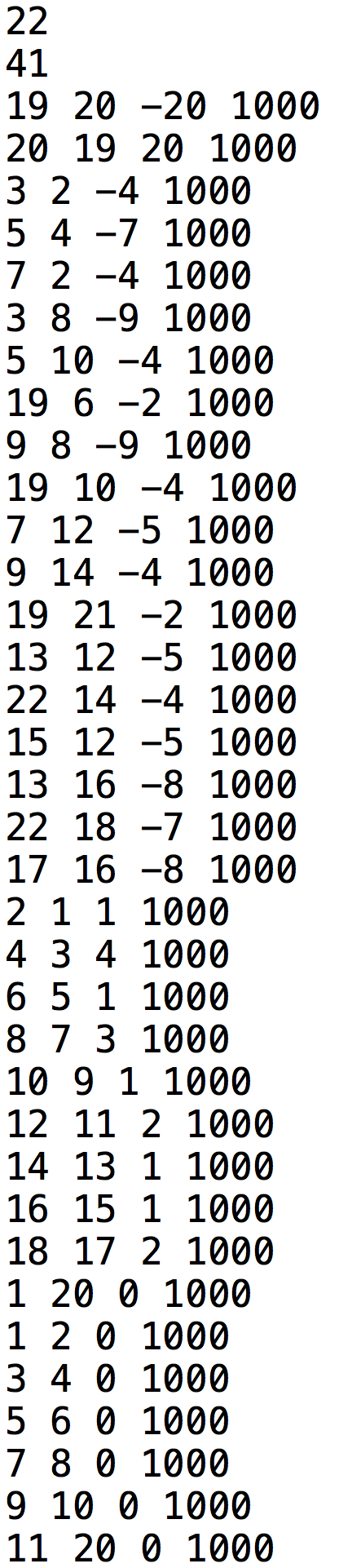
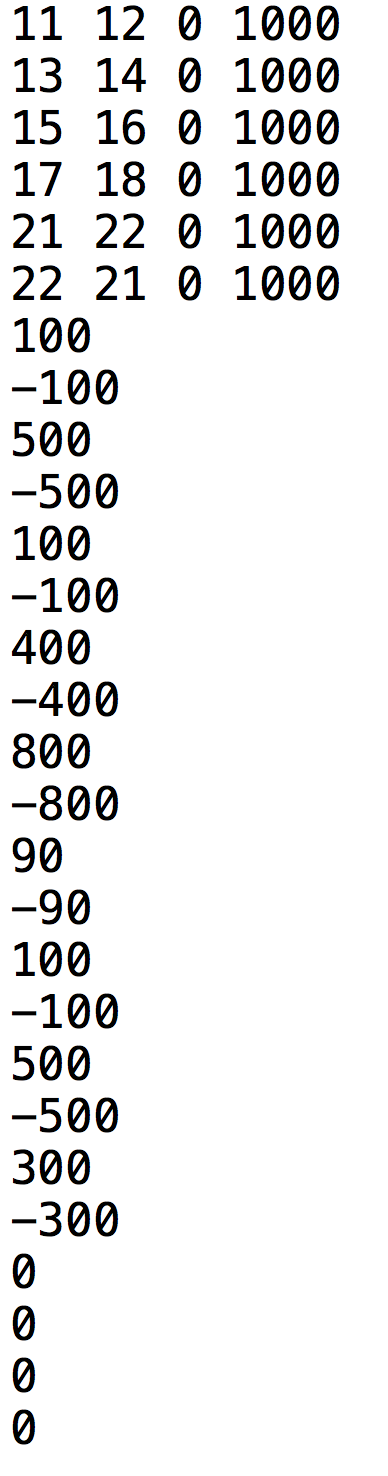
**PARTE III**

1. Para a construção do grafo, analisou-se a matriz produzida pelo modelo dual do modelo alternativo. Com base nas colunas que continha a informação dos arcos conseguiu-se obter a origem e o destino de cada um deles, o que levou a ser possível a construção do grafo. Para além disto, uma das linhas continha o custo de cada arco e os valores de oferta e procura de cada vértice.

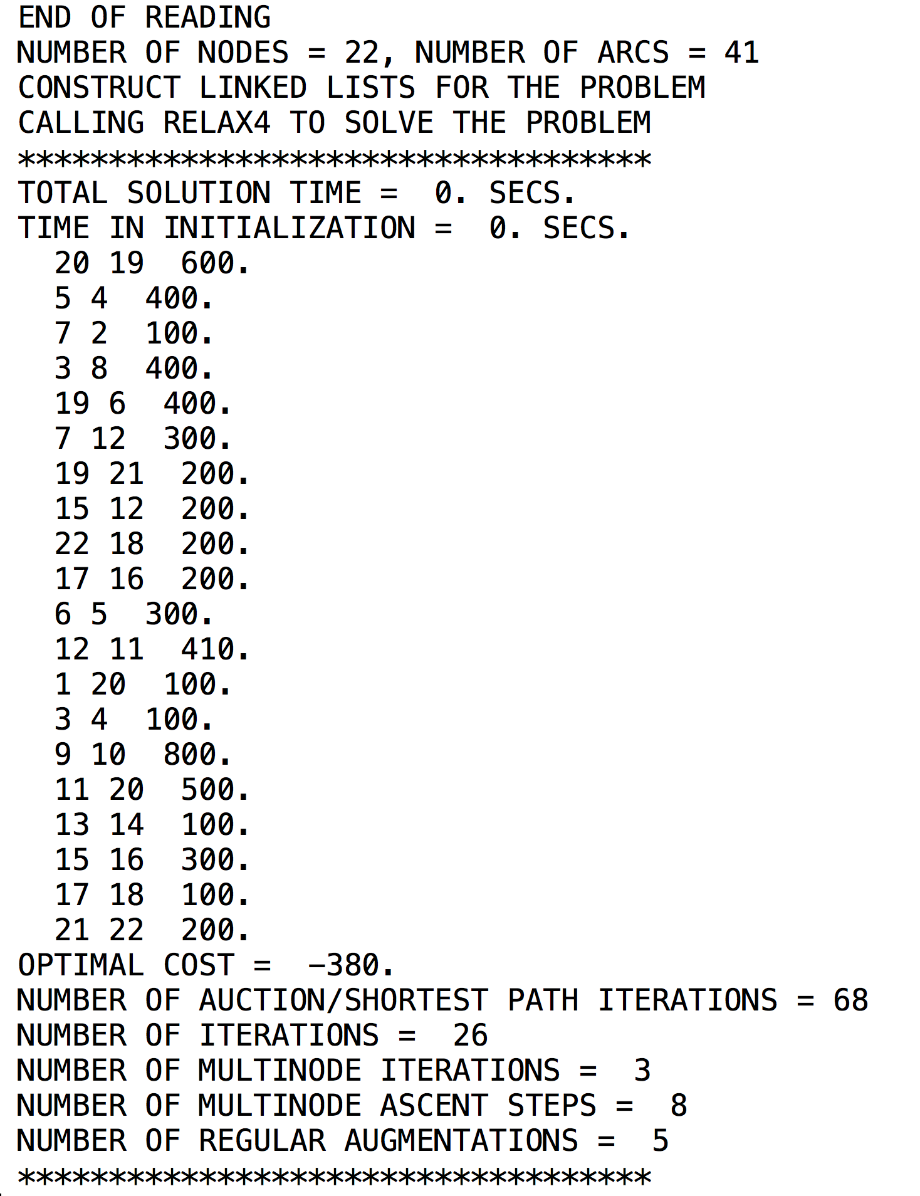




1. Input do *Relax4*:



1. Output produzido pelo *Relax4*:



1. Como a solução ótima obtida nesta parte é igual à obtida na parte II e como na Parte II se comprovou a correção dessa solução, visto que esta é igual à obtida no primal, através do Teorema Forte da Dualidade, concluímos que a solução ótima da Parte II está correta e, por conseguinte, esta é também a solução ótima para o problema na Parte III.