

UNIVERSIDADE DO MINHO

---

# SAT

---

MESTRADO INTEGRADO EM ENGENHARIA  
INFORMÁTICA

VERIFICAÇÃO FORMAL

2ºSEMESTRE 2020/21

*Aluno:*  
Etienne Costa

*Docente:*  
Maria João Frade

7 de março de 2021

# Conteúdo

<b>1</b>	<b>Resumo</b>	<b>2</b>
<b>2</b>	<b>Puzzle</b>	<b>3</b>
<b>3</b>	<b>Sudoku</b>	<b>6</b>
3.1	Restrições e Modelação . . . . .	6
3.2	Formato DIMACS CNF . . . . .	6
3.3	Resultados obtidos . . . . .	8
3.3.1	Sudoku 2x2 - Puzzle 1 . . . . .	8
3.3.2	Sudoku 2x2 - Puzzle 2 . . . . .	9
3.3.3	Sudoku 3x3 - Puzzle 1 . . . . .	10
3.3.4	Sudoku 3x3 - Puzzle 2 . . . . .	11

## 1 Resumo

O relatório apresentado diz respeito ao primeiro trabalho proposto no âmbito da unidade curricular de verificação formal. O universo de discurso recaí sobre problemas de decisão, cuja instância é uma expressão booleana escrita somente com operadores *AND* , *OR* , *NOT* e *variáveis*.

## 2 Puzzle

A "Associação Recreativa do Paraíso" tem o seguinte conjunto de regras :

- Os sócios loiros não podem ir ao Sábado.
- Quem não for adulto tem que usar chapéu.
- Cada sócio usa anel ou não usa chapéu.
- Um sócio vai ao Sábado se e só se é adulto.
- Todos os sócios adultos têm que usar anel.
- Quem usa anel tem que ser loiro.

Para a resolução deste puzzle foram definidas as seguintes variáveis proposicionais:

- loiro : 1
- sábado : 2
- adulto : 3
- chapéu : 4
- anel : 5

De seguida são apresentadas as fórmulas proposicionais e as respectivas conversões para a forma normal conjuntiva (CNF) :

1 **Regra 1** : *Os sócios loiros não podem ir ao Sábado.*

2

3 *Fórmula proposicional* :  $\text{loiro} \implies \neg \text{sábado}$

4

5 *Forma normal conjuntiva* :  $\neg \text{loiro} \vee \neg \text{sábado}$

6

7 *Número de cláusulas* : 1

1 **Regra 2** : *Quem não for adulto tem que usar chapéu.*

2

3 *Fórmula proposicional* :  $\neg \text{adulto} \implies \text{chapéu}$

4

5 *Forma normal conjuntiva* :  $\text{adulto} \vee \text{chapéu}$

6

7 *Número de cláusulas* : 1

1 **Regra 3** : *Cada sócio usa anel ou não usa chapéu*

2

3 *Fórmula proposicional* :  $\text{anel} \vee \neg \text{chapéu}$

4

5 *Forma normal conjuntiva* :  $\text{adulto} \vee \neg \text{chapéu}$

6

7 *Número de cláusulas* : 1

1 **Regra 4** : *Um sócio vai ao Sábado se e só se é adulto*

2

3 *Fórmula proposicional* :  $\text{sábado} \Leftrightarrow \text{adulto}$

4

5 *Forma normal conjuntiva* :  $(\neg \text{sábado} \vee \text{adulto}) \wedge (\text{sábado} \vee \neg \text{adulto})$

6

7 *Número de cláusulas* : 2

```

1 Regra 5 : Todos os sócios adultos têm que usar anel
2
3 Fórmula proposicional :  $adulto \implies anel$ 
4
5 Forma normal conjuntiva :  $\neg adulto \vee anel$ 
6
7 Número de cláusulas : 1

```

```

1 Regra 6 : Quem usa anel tem que ser loiro
2
3 Fórmula proposicional :  $anel \implies loiro$ 
4
5 Forma normal conjuntiva :  $\neg anel \vee loiro$ 
6
7 Número de cláusulas : 1

```

Feito isto, é escrito o resultado obtido no formato DIMACS CNF, formato esse usado para definir expressões booleanas, escritas na forma normal conjuntiva passando de seguida para o SAT Solver :

```

1 c 5 variáveis e 7 cláusulas
2 c 1 loiro
3 c 2 sábado
4 c 3 adulto
5 c 4 chapéu
6 c 5 anel
7
8 p cnf 5 7
9 -1 -2 0
10 3 4 0
11 5 -4 0
12 -2 3 0
13 2 -3 0
14 -3 5 0
15 -5 1 0

```

O resultado produzido pelo SAT solver foi o seguinte :

```

1 s SATISFIABLE
2 v 1 -2 -3 4 5 0
3
4 loiro  $\wedge$   $\neg$ sábado  $\wedge$   $\neg$ adulto  $\wedge$  chapéu  $\wedge$  anel
5
6 Podemos afirmar que o modelo é satisfazível, i.e , é possível encontrar uma
  valoração que satisfaça o modelo.

```

De seguida foram propostas algumas questões que podiam ser respondidas à custa do modelo definido:

```

1 A afirmação " Quem usa anel não pode ir ao Sábado." é correcta ?
2
3 Para responder a isso foi feita a negação da fórmula proposicional obtendo o
  seguinte :
4
5 Fórmula proposicional :  $\neg(\text{anel} \implies \neg\text{sábado})$ 
6
7 Forma normal conjuntiva :  $(\text{anel} \wedge \text{sábado})$ 
8
9 Número de cláusulas : 2
10
11 Adicionando as duas novas cláusulas ao modelo:
12
13 p cnf 5 9
14 -1 -2 0
15 3 4 0
16 5 -4 0
17 -2 3 0
18 2 -3 0
19 -3 5 0
20 -5 1 0
21 5 0
22 2 0
23
24 Obtem-se como resposta UNSATISFIABLE, com base nisso podemos afirmar que
  a afirmação é VERDADEIRA.

```

```

1 Pode um sócio de chapéu ser loiro ?
2
3 Com base na valoração obtida no modelo inicial, chegamos ao seguinte:
4
5 s SATISFIABLE
6 v 1 -2 -3 4 5 0
7 ≡
8 loiro  $\wedge$   $\neg$ sábado  $\wedge$   $\neg$ adulto  $\wedge$  chapéu  $\wedge$  anel
9
10 Podemos confirmar à custa da fórmula proposicional acima que é possível ter sócios
  de chapéu e cabelo loiro.

```

```

1 A afirmação " Afinal a associação não pode ter sócios adultos. " é correcta ?
2
3 Para responder a isso foi feita a negação da fórmula proposicional obtendo o
  seguinte :
4
5 Fórmula proposicional :  $\neg(\neg\text{adulto})$ 
6
7 Forma normal conjuntiva : adulto
8
9 Número de cláusulas : 1
10
11 Adicionando a nova cláusula ao modelo:
12
13 p cnf 5 8
14 -1 -2 0
15 3 4 0
16 5 -4 0
17 -2 3 0
18 2 -3 0
19 -3 5 0
20 -5 1 0
21 3 0
22
23 Obtem-se como resposta UNSATISFIABLE, com base nisso podemos afirmar que
  a afirmação é VERDADEIRA.

```

### 3 Sudoku

Os puzzles Sudoku são problemas de colocação de números inteiros entre 1 e  $N^2$  numa matriz quadrada de dimensão  $N^2$ , por forma a que **cada coluna e cada linha contenha todos os números, sem repetições**. Além disso, cada matriz contém  $N^2$  sub-matrizes quadradas disjuntas, de dimensão  $N$ , **que deverão também elas conter os números entre 1 e  $N^2$** . Cada problema é dado por uma matriz parcialmente preenchida, cabendo ao jogador completá-la.

#### 3.1 Restrições e Modelação

Com base nesta breve introdução é possível especificar algumas restrições correspondentes às regras do puzzle bem com uma possível representação do modelo:

1	<i>Restrição 1: Pelo menos um dígito em cada célula</i>
2	$\bigwedge_{1 \leq linha \leq n, 1 \leq coluna \leq n} (x_{linha,coluna,1} \vee x_{linha,coluna,2} \vee \dots \vee x_{linha,coluna,n})$
1	<i>Restrição 2 : No máximo um dígito em cada célula</i>
2	$\bigwedge_{1 \leq linha \leq n, 1 \leq coluna \leq n, 1 \leq valor \leq valor' \leq n} (\neg x_{linha,coluna,valor} \vee \neg x_{linha,coluna,valor'})$
1	<i>Restrição 3 : Cada linha possui todos os dígitos</i>
2	$\bigwedge_{1 \leq linha \leq n, 1 \leq valor \leq n} (x_{linha,1,valor} \vee x_{linha,2,valor} \vee \dots \vee x_{linha,n,valor})$
1	<i>Restrição 4 : Cada coluna possui todos os dígitos</i>
2	$\bigwedge_{1 \leq coluna \leq n, 1 \leq valor \leq n} (x_{1,coluna,valor} \vee x_{2,coluna,valor} \vee \dots \vee x_{n,coluna,valor})$
1	<i>Restrição 5 : Cada sub – matriz possui todos os dígitos</i>
2	$\bigwedge_{1 \leq linha' \leq \sqrt{n}, 1 \leq coluna' \leq \sqrt{n}, 1 \leq valor \leq n} (\bigvee_{(linha,coluna) \in B_n(linha',coluna')} x_{linha,coluna,valor})$
3	<i>where</i>
4	$B_n(linha',coluna') = \{(linha'\sqrt{n} + i, coluna'\sqrt{n} + j)   0 \leq i < \sqrt{n}, 0 \leq j < \sqrt{n}\}$
1	<i>Variável Proposicional:</i>
2	
3	$X_{l,c,v} = 1$ : Se o valor(v) encontra-se na linha (l) e coluna(c)
4	$X_{l,c,v} = 0$ : Se o valor(v) não se encontra na linha (l) e coluna(c) .

#### 3.2 Formato DIMACS CNF

Para a geração do ficheiro DIMACS CNF , tirou-se partido das facilidades que a linguagem **Python** oferece.

A ideia principal do **script** criado, consiste na geração de todas as fórmulas proposicionais à custa da dimensão da sub-matriz e um ficheiro de entrada com a representação de um estado inicial da mesma.

Tendo em conta que boa parte dos algoritmos perdem desempenho com inputs maiores, optou-se por otimizar os ciclos gerando muitas das vezes as fórmulas proposicionais de 2 restrições numa só iteração.

Uma das condições iniciais do puzzle, está relacionada com o estado inicial que o puzzle apresenta, de forma a facilitar essa inserção é passado um ficheiro de entrada no seguinte formato :

4.1.  
 .2..  
 ..3..  
 .4.1

4		1	
	2		
		3	
	4		1

Ficheiro de texto

Matriz correspondente

Após a geração de todas as cláusulas e variáveis proposicionais, é feita a passagem do ficheiro gerado para o SAT Solver e feita a filtragem dos valores positivos à custa do seguinte comando **grep -Eo '[0-9]+' ficheiro**, pois só estes correspondem as variáveis proposicionais do nosso interesse.



### 3.3 Resultados obtidos

#### 3.3.1 Sudoku 2x2 - Puzzle 1

4		1	
	2		
		3	
	4		1

4	3	1	2
1	2	4	3
2	1	3	4
3	4	2	1

1	
2	<i>Pode ser feita a interpretação da solução à custa do último valor, i.e,</i>
	<i><math>m[1][1]=4</math> <math>m[1][2]=3</math> <math>m[1][3]=1</math> <math>m[1][4]=2</math>.</i>
3	
4	114
5	123
6	131
7	142
8	211
9	222
10	234
11	243
12	312
13	321
14	333
15	344
16	413
17	424
18	432
19	441
20	0

### 3.3.2 Sudoku 2x2 - Puzzle 2

2			
		3	
	1		
			3

2	3	4	1
1	4	3	2
3	1	2	4
4	2	1	3

1	
2	<i>Pode ser feita a interpretação da solução à custa do último valor, i.e.,</i>
	<i><math>m[1][1]=2</math> <math>m[1][2]=3</math> <math>m[1][3]=4</math> <math>m[1][4]=1</math>.</i>
3	
4	112
5	123
6	134
7	141
8	211
9	224
10	233
11	242
12	313
13	321
14	332
15	344
16	414
17	422
18	431
19	443
20	0

### 3.3.3 Sudoku 3x3 - Puzzle 1

8		2				6		9
1			7		2			8
5								1
	1			8	3		5	
	7		1	6			4	
6								2
2			4		7			5
7		1				8		4

8	3	2	5	4	1	6	7	9
1	6	4	7	9	2	5	3	8
5	9	7	6	3	8	4	2	1
4	1	6	2	8	3	9	5	7
3	2	5	9	7	4	1	8	6
9	7	8	1	6	5	2	4	3
6	4	3	8	5	9	7	1	2
2	8	9	4	1	7	3	6	5
7	5	1	3	2	6	8	9	4

The solution is correct  
 You took 04 minutes 25 seconds to solve the puzzle

### 3.3.4 Sudoku 3x3 - Puzzle 2

5								2
2	7		9		5		4	8
	4		3		2		5	
	1	9				5	2	
				1				
	5	4				7	3	
	6		1		8		9	
1	9		4		6		7	5
4								3

5	3	8	7	4	1	9	6	2
2	7	1	9	6	5	3	4	8
9	4	6	3	8	2	1	5	7
3	1	9	8	7	4	5	2	6
6	2	7	5	1	3	4	8	9
8	5	4	6	2	9	7	3	1
7	6	3	1	5	8	2	9	4
1	9	2	4	3	6	8	7	5
4	8	5	2	9	7	6	1	3

The solution is correct  
You took 03 minutes 33 seconds to solve the puzzle