

Trabalho 1 LC-20/21 LCC

Grupo 5
Filipe Barbosa A77252
Hugo Ferreira A78555

```
In [53]: from z3 import *
```

Problema 1

Queremos construir um horário semanal para o plano de reuniões de projeto de uma “StartUp”.

Começamos por definir as condições: para salas ordenados de 1 a S , dias de 1 a D , tempos de 1 a T , projetos de 1 a P , e colaboradores de 1 a C .

Criamos de seguida inputs com auxilio da ferramenta *numpy.random.randint* definindo os valores das constantes S, D, T, P, C , a quantidade de reuniões de cada projeto e o seu lider.

```
In [54]: import numpy

S = numpy.random.randint(1,5)
D = numpy.random.randint(1,5)
T = numpy.random.randint(1,5)
P = numpy.random.randint(1,5)
C = numpy.random.randint(2,6)

projetos = []

for i in range(P):
    aux = []
    colabs = numpy.random.randint(1,5,size=numpy.random.randint(1,C))
    aux.append(i)
    aux.append(list(set(colabs)))
    aux.append(numpy.random.choice(colabs))
    aux.append(numpy.random.randint(1,5))
    projetos.append(aux)

print("S:",S,"D:",D,"T:",T,"P:",P,"C:",C)
print(projetos)

S: 4 D: 2 T: 4 P: 3 C: 2
[[0, [3], 3, 1], [1, [3], 3, 3], [2, [4], 4, 2]]
```

De seguida criamos o solver *horario*.

```
In [55]: horario = Solver()

x={}
y={}

Exemplos:
```

```
In [56]: '''
S, D, T, P, C= 2, 2, 2, 2, 6
myInput = [[0,[1,2,3],2,1],[1,[1,2,3,4],3,1]]

col = {}
lid = {}
reu = {}

for i in myInput:
    col[i[0]] = i[1]
    lid[i[0]] = i[2]
    reu[i[0]] = i[3]
'''

Out[56]: '\nS, D, T, P, C= 2, 2, 2, 2, 6\nmyInput = [[0,[1,2,3],2,1],[1,[1,2,3,4],3,1]]\nncol = {}\nlid = {}
\nreu = {}\n\nfor i in myInput:\n    col[i[0]] = i[1]\n    lid[i[0]] = i[2]\n    reu[i[0]] = i[3]\n'
```

Separamos os inputs por projeto e por tipo (colaboradores, reuniões e lider).

```
In [57]: col = {}
lid = {}
reu = {}

for i in projetos:
    col[i[0]] = i[1]
    lid[i[0]] = i[2]
    reu[i[0]] = i[3]

print("col",col)
print("lid",lid)
print("reu",reu)

col {0: [3], 1: [3], 2: [4]}
lid {0: 3, 1: 3, 2: 4}
reu {0: 1, 1: 3, 2: 2}
```

Em seguida, declaramos a matriz de alocação, x . Definimos a nossa matriz com todas as variáveis do programa.

$$\forall_p \forall_r \forall_d \forall_s \forall_t \forall_c \cdot 0 \leq x_{p,r,d,s,t,lid_p,c} \leq 1$$

```
In [58]: for p in range(P):
    for r in range(reu[p]):
        for d in range(D):
            for s in range(S):
                for t in range(T):
                    for c in col[p]:
                        x[p,r,d,s,t,lid[p],c] = Int("[p:"+str(p)+" r:"+str(r)+" d:"+str(d)+" s:"+str(s)
+" t:"+str(t)+" l:"+str(lid[p])+" c:"+str(c)+"]")
                        horario.add(x[p,r,d,s,t,lid[p],c] >= 0,x[p,r,d,s,t,lid[p],c] <= 1)
```

Criamos agora as seguintes restrições:

1.O número total de reuniões não pode ser maior do que o número do projeto vezes o número de colaboradores.

$$\forall_p \cdot \sum_{r,d,s,t,c} x_{p,r,d,s,t,lid_p,c} \leq Reu_p \times |Col_p|$$

```
In [59]: for p in range(P):
    horario.add(Sum([x[p,r,d,s,t,lid[p],c] for r in range(reu[p]) for d in range(D) for s in range(S) f
or t in range(T) for c in col[p]]) <= reu[p]*len(col[p]))
```

2.Reunião apenas acontece num dia, numa sala, num tempo.

$$\forall_p \forall_r \cdot \sum_{d,s,t,c} x_{p,r,d,s,t,lid_p,c} \leq |Col_p|$$

```
In [60]: for p in range(P):
    for r in range(reu[p]):
        horario.add(Sum([x[p,r,d,s,t,lid[p],c] for d in range(D) for s in range(S) for t in range(T) fo
r c in col[p]]) <= len(col[p]))
```

3.Certo colaborador apenas está presente numa sala num determinado dia, hora, tempo.

$$\forall_p \forall_c \forall_d \forall_t \cdot \sum_{r,s} x_{p,r,d,s,t,lid_p,c} \leq 1$$

```
In [61]: for d in range(D):
    for t in range(T):
        for p in range(P):
            for c in col[p]:
                horario.add(Sum([x[p,r,d,s,t,lid[p],c] for s in range(S) for r in range(reu[p])]) <= 1)
```

4. Para realizar-se uma reunião deve ter pelo menos 50% dos colaboradores do projeto a participar.

$$\forall_p \forall_r \cdot \sum_{d,s,t,c} x_{p,r,d,s,t,lid_p,c} \leq |col_p|$$

De seguida testamos se o número de colaboradores é par ou impar porque o resultado da divisão de um número inteiro impar por 2 é um número arredondado para baixo e neste caso para a reunião ter pelo menos 50% assistência temos de adicionar mais 1.

Se for par fazemos:

$$\forall_p \forall_r \cdot \sum_{d,s,t,c} x_{p,r,d,s,t,lid_p,c} \geq |col_p| \div 2$$

Se for impar fica então:

$$\forall_p \forall_r \cdot \sum_{d,s,t,c} x_{p,r,d,s,t,lid_p,c} \geq (|col_p| \div 2) + 1$$

```
In [62]: for p in range(P):
    for r in range(reu[p]):
        horario.add(Sum([x[p,r,d,s,t,lid[p],c] for d in range(D) for s in range(S) for t in range(T) fo
r c in col[p]]) <= len(col[p]))
        if (len(col[p])%2)==0:
            horario.add(Sum([x[p,r,d,s,t,lid[p],c] for d in range(D) for s in range(S) for t in range(T)
for c in col[p]]) >= (len(col[p])/2))
        else:
            horario.add(Sum([x[p,r,d,s,t,lid[p],c] for d in range(D) for s in range(S) for t in range(T)
for c in col[p]]) >= (len(col[p])/2)+1)
```

5.Lider comparece em todas as reuniões dos projetos que lidera.

$$\forall_p \cdot \sum_{r,s,d,t} x_{p,r,d,s,t,lid_p,lid_p} = reu_p$$

```
In [63]: for p in range(P):
    horario.add(Sum([x[p,r,d,s,t,lid[p],lid[p]] for r in range(reu[p]) for s in range(S) for d in range
(D) for t in range(T) ]) == reu[p]))
```

6.Colaborador não pode ir à mesma reunião em varios dias, horas e tempos.

$$\forall_p \forall_r \forall_c \cdot \sum_{s,d,t} x_{p,r,d,s,t,l,c} \leq 1$$

```
In [64]: for p in range(P):
    for r in range(reu[p]):
        for c in col[p]:
            horario.add(Sum([x[p,r,d,s,t,lid[p],c] for d in range(D) for s in range(S) for t in range(T)
])) <= 1)
```

Finalizamos agora testando se é possível satisfazer as restrições e imprimimos o horário.

```
In [65]: r = horario.check()
print(r)

if r == sat:
    m = horario.model()

for p in range(P):
    for r in range(reu[p]):
        for c in col[p]:
            for d in range(D):
                for t in range(T):
                    for s in range(S):
                        if (m[x[p,r,d,s,t,lid[p],c]] == 1):
                            print("projeto",p,"Reunião",r,"Lider",lid[p],"Colaborador",c,"Dia",d,"Hora"
,t,"Sala",s)

sat
projeto 0 Reunião 0 Lider 3 Colaborador 3 Dia 0 Hora 0 Sala 0
projeto 1 Reunião 0 Lider 3 Colaborador 3 Dia 0 Hora 0 Sala 3
projeto 1 Reunião 1 Lider 3 Colaborador 3 Dia 1 Hora 2 Sala 0
projeto 1 Reunião 2 Lider 3 Colaborador 3 Dia 0 Hora 2 Sala 3
projeto 2 Reunião 0 Lider 4 Colaborador 4 Dia 0 Hora 2 Sala 2
projeto 2 Reunião 1 Lider 4 Colaborador 4 Dia 1 Hora 3 Sala 3
```