

TP 1 EX 1

Grupo 5:

Breno Fernando Guerra Marrão A97768

Tales André Rovaris Machado

Inicialização

Usamos as bibliotecas Or-tools (com interface SCIP) e random para resolver o problema de alocação proposto

```
In [1]: from ortools.linear_solver import pywraplp
        from random import randrange, choice
        horario = pywraplp.Solver.CreateSolver('SCIP')
```

Gerador de dados aleatorios

Para gerar os dados teste para o problema criamos um gerador que pode ser alterado para de acordo com os objetivos desejados de simulação Disp é a matriz 2x2 que para cada elemento i,j tem o valor 0 ou 1 com a disponibilidade do colaborador i no turno j Proj é o array de tuplos onde o primeiro elemento é a constituição do grupo, o segundo é o lider do grupo e o terceiro as horas minimas por projeto

In [2]:

```

S,T,P,C = 5,20,10,100

#A disponibilidade de cada participante, incluindo o líder. Essa disp
#representada numa matriz booleana de acessibilidade com uma linha po
#uma coluna por "slot" 1..T

disp = []

h= [0,1,1,1]
for c in range(C):
    disp.append([])
    for t in range(T):
        disp[c].append(choice(h))

#O conjunto de colaboradores de cada projeto, o seu líder e o número
#Nesse caso para um projeto de 0 - P , está associado os colaboradores
proj = []
lideres = []
for p in range(P):
    colab = set()
    for col in range(14):
        colab.add(randrange(C))
    cola = list(colab)
    l = choice(cola)
    lideres.append(l)
    hora = randrange(1,5)
    proj.append((cola,l,hora))

```

Inicialização da matriz

In [3]:

```

x = {}
for s in range(S):
    x[s] = {}
    for t in range(T):
        x[s][t] = {}
        for p in range(P):
            x[s][t][p] = {}
            for c in range(C):
                x[s][t][p][c] = horario.BoolVar(f'x[{s},{t},{p},{c}]')

def X(s,t,p,c): # abreviatura
    return x[s][t][p][c]

```

$$\forall_{c < C} \forall_{t < T} \quad \text{se } disp_{c,t} = 0 \quad \text{então} \quad \forall_{p < P} \forall_{s < S} \quad X_{s,t,p,c} = 0$$

```
In [4]: #Adiciona condições para que se um colaborador não esteja disponível
#ele não posso ser alocada a esse spot
for c in range(C):
    for t in range(T):
        if disp[c][t] == 0:
            for p in range(P):
                for s in range(S):
                    horario.Add((X(s,t,p,c)) == 0 )
```

$$\forall c < C \quad \forall p < P \quad \text{se } c \notin \text{proj}_p \quad \text{então } \forall s < S \quad \forall t < T \quad X_{s,t,p,c} = 0$$

```
In [5]: #cada colaborador a cada dia e nao pode participar de um projeto que
for c in range(C):
    for p in range(P):
        if c not in proj[p][0]:
            for s in range(S):
                for t in range(T):
                    horario.Add((X(s,t,p,c)) == 0)
```

$$\forall c < C \quad \forall t < T \quad \sum_{p < P, s < S} X_{s,t,p,c} \leq 1$$

```
In [6]: #checa se o mesmo colaborador esta fazendo o trabalho em 2 projetos
for c in range(C):
    for t in range(T):
        horario.Add(sum(X(s,t,p,c) for p in range(P) for s in range(S))
```

$$\forall p < p \quad \forall l \in \text{lideres} \quad \forall t < T \quad \text{se } \text{disp}_{l,t} \quad \text{então } \forall s < S \quad \forall c \in \text{proj}_p \quad X_{s,t,p,c} = 0$$

```
In [7]: #Adiciona condições para que se o lider não esteja disponível não alocar
for p in range(P):
    l = proj[p][1]
    for t in range(T):
        if disp[l][t] == 0:
            for s in range(S):
                for c in proj[p][0]:
                    horario.Add((X(s,t,p,c)) == 0 )
```

$$\forall s < S \quad \forall t < T \quad \sum_{p < P, c < C} X_{s,t,p,c} \leq 1$$

```
In [8]: # Cada sala não pode ter mais que uma reunião ao mesmo tempo
for s in range(S):
    for t in range(T):
        horario.Add(sum(X(s,t,p,c) for p in range(P) for c in lideres
```

$$\forall p < P \quad \forall t < T \quad \forall s < s \quad \sum_{c < C} x_{s,t,p,c} \geq |\text{proj}_{p,0}| \div 2$$

```
In [9]: #Adiciona condições para que pelo menos 50% dos colaboradores de um projeto
#alocado a este projeto
for p in range(P):
    for t in range(T):
        for s in range(S):
            horario.Add(sum(X(s,t,p,c) for c in proj[p][0]) >= (X(s,
```

$$\forall_{p < P}. \quad proj_{p,2} \leq \sum_{s < S, t < T} X_{s,t,p,lideres_p}$$

```
In [10]: #Adiciona condições para que o número de reuniões por projeto seja maior
for p in range(P):
    horario.Add ( proj[p][2] <= sum(X(s,t,p,lideres[p]) for s in range(S))
```

Para otimizar a solução tivemos de maximizar o número total de reuniões através do `Maximize(h)` e minimizar o numero de pessoas por reunião, assim diminuindo a media de pessoas por projeto

```
In [11]: for p in range(P):
        for s in range(S):
            for t in range(T):
                horario.Minimize(sum(X(s,t,p,c) for c in range(C)))

h = sum(X(s,t,p,proj[p][1]) for s in range(S) for t in range(T) for p in range(P))

horario.Maximize(h)
```

```
In [12]: status = horario.Solve()
print(status)
if status == pywraplp.Solver.OPTIMAL or status == pywraplp.Solver.FEASIBLE:
    for s in range(S):
        print("Sala ", s, ":")
        for t in range(T):
            for p in range(P):
                if X(s,t,p,proj[p][1]).solution_value() == 1:

                    a = []
                    for c in proj[p][0] :
                        if X(s,t,p,c).solution_value() == True:
                            a.append(c)
                    print("Slot ", t , " Projeto:", p, " Colaboradores:"
```

```
0
Sala 0 :
Slot 0 Projeto: 9 Colaboradores: [38, 39, 73, 75, 52, 87]
Slot 1 Projeto: 1 Colaboradores: [1, 33, 12, 50, 93, 51]
Slot 2 Projeto: 2 Colaboradores: [65, 71, 74, 80, 18, 89, 30]
Slot 3 Projeto: 4 Colaboradores: [0, 6, 11, 76, 78, 48, 49, 19,
85, 58]
Slot 4 Projeto: 2 Colaboradores: [32, 65, 14, 80, 18, 93, 31]
Slot 5 Projeto: 6 Colaboradores: [37, 70, 75, 95, 86, 92, 61]
Slot 6 Projeto: 0 Colaboradores: [72, 81, 55, 24, 58, 94]
Slot 7 Projeto: 5 Colaboradores: [64, 99, 70, 45, 84, 31]
Slot 8 Projeto: 8 Colaboradores: [32, 33, 8, 10, 21, 85]
Slot 9 Projeto: 9 Colaboradores: [35, 9, 73, 75, 52, 87]
Slot 10 Projeto: 2 Colaboradores: [32, 65, 71, 18, 89, 93, 30]
Slot 11 Projeto: 2 Colaboradores: [65, 71, 74, 14, 80, 18, 89]
Slot 12 Projeto: 0 Colaboradores: [70, 47, 55, 24, 58, 94]
Slot 13 Projeto: 0 Colaboradores: [67, 72, 55, 24, 58, 94]
Slot 14 Projeto: 7 Colaboradores: [96, 2, 68, 66, 72, 41, 46]
Slot 15 Projeto: 4 Colaboradores: [0, 78, 48, 49, 85, 89]
Slot 16 Projeto: 0 Colaboradores: [35, 9, 6, 11, 52, 87]
```