

Trabalho 1

Grupo 06

- Tomás Vaz de Carvalho Campinho A91668
- Miguel Ângelo Alves de Freitas A91635

▼ Problema 1

1. Pretende-se construir um horário semanal para o plano de reuniões de projeto de uma “StartUp” de acordo com as seguintes condições:
 1. Cada reunião ocupa uma sala (enumeradas 1..S) durante um “slot” (tempo, dia). Assume-se os dias enumerados 1..De, em cada dia, os tempos enumerados 1..T.
 2. Cada reunião tem associado um projeto (enumerados 1..P) e um conjunto de participantes. Os diferentes colaboradores são enumerados 1..C.
 3. Cada projeto tem associado um conjunto de colaboradores, dos quais um é o líder. Cada projeto realiza um dado número de reuniões semanais. São “inputs” do problema o conjunto de colaboradores de cada projeto, o seu líder e o número de reuniões semanais.
 4. O líder do projeto participa em todas as reuniões do seu projeto; os restantes colaboradores podem ou não participar consoante a sua disponibilidade, num mínimo (“quorum”) de 50% do total de colaboradores do projeto. A disponibilidade de cada participante, incluindo o líder, é um conjunto de “slots” (“inputs” do problema).

Instalar e importar bibliotecas necessárias:

```
!pip3 install ortools
```

```
from ortools.linear_solver import pywraplp
```

▼ Inicialização

Inicializamos o solver horário, a matriz de alocação e todos os inputs do problema:

```
#por exemplo:  
horario = pywraplp.Solver.CreateSolver('SCIP')  
x = {}
```

```
P,C,S,T,D = 2,6,1,2,1
```

```
colabs = {1:[1,2],2:[3,4,5,6]} #colaboradores (numero_do_projeto:[colaborador,...])
lids = {1:2,2:3} #líderes (numero_do_projeto:lider)
reuns = {1:1,2:1} #reuniões (numero_do_projeto:numero_de_reunioes)
disps = {1:[(1,1)],2:[(1,1)],3:[(2,1)],4:[(2,1)],5:[(2,1)],6:[]} #disponibilidades (colabo
```

Definimos x (matriz de alocação) com as variáveis do problema:

```
def inicializarHorario(colabs):
    for p in range(1,P+1):
        x[p] = {}
        for c in range(1,C+1):
            x[p][c] = {}
            for s in range(1,S+1):
                x[p][c][s] = {}
                for t in range(1,T+1):
                    x[p][c][s][t] = {}
                    for d in range(1,D+1):
                        x[p][c][s][t][d] = horario.BoolVar('x [%i] [%i] [%i] [%i] [%i]' % (p,c,s,t,d))

def X(p,c,s,t,d):
    return x[p][c][s][t][d]
```

▼ Restrições

1. No mínimo 50% dos colaboradores do projeto presentes em cada reunião:

```
def nColabsPresentes(colabs):
    for p in range(1,P+1):
        horario.Add(sum([X(p,c,s,t,d) for d in range(1,D+1) for s in range(1,S+1) for t in r
```

2. Se um colaborador não estiver disponível, então esse colaborador não vai à reunião:

```
def indisponiveis(disps):
    for c in range(1,C+1):
        for t in range(1,T+1):
            for d in range(1,D+1):
                if (t,d) not in disps[c]:
                    horario.Add(sum([X(p,c,s,t,d) for p in range(1,P+1) for s in range(1,S+1)])==0)
```

3. Um colaborador não pode estar em várias reuniões ao mesmo tempo:

```
def colabUnicaReuniao():
    for t in range(1,T+1):
        for d in range(1,D+1):
            for c in range(1,C+1):
                horario.Add(sum([X(p,c,s,t,d) for s in range(1,S+1) for p in range(1,P+1)])<=1)
```

4. Para cada sala num slot (tempo,dia) acontece no máximo uma reunião:

```
def umaSalaumaReuniao(lids):
    for s in range(1,S+1):
        for t in range(1,T+1):
            for d in range(1,D+1):
                horario.Add(sum([X(p,lids[p],s,t,d) for p in range(1,P+1)])<=1)
```

5. O líder tem de estar presente em todas as reuniões do projeto:

```
def nReunioesLider(lids, reuns):
    for p in range(1,P+1):
        horario.Add(sum([X(p,lids[p],s,t,d) for s in range (1,S+1) for t in range (1,T+1) for
```

▼ Construção do horário

Imprimir horário:

```
def imprimeHorario():
    r = horario.Solve()
    if r == pywraplp.Solver.OPTIMAL:
        for d in range(1,D+1):
            print("DIA",d)
            for t in range(1,T+1):
                print("\tTEMPO",t)
                for p in range(1,P+1):
                    for c in range(1,C+1):
                        for s in range(1,S+1):
                            if (t,d) in disps[c]:
                                if c in colabs[p]:
                                    print("\t","projeto",p,"colaborador",c,"sala",s)
    else:
        print("Não é possível construir horário pois não está de acordo com todas as restrições")
```

Construção final do horário de acordo com todos os requisitos:

```
def horarioStartUp(colabs,lids,reuns,disps):
    inicializarHorario(colabs)

    nColabsPresentes(colabs)
    indisponiveis(disps)
    colabUnicaReuniao()
    umaSalaumaReuniao(lids)
    nReunioesLider(lids,reuns)

    imprimeHorario()
```

```
horarioStartUp(colabs,lids,reuns,disps)
```

```
DIA 1
```

```
TEMPO 1
```

```
projeto 1 colaborador 1 sala 1
```

```
projeto 1 colaborador 2 sala 1
```

```
TEMPO 2
```

```
projeto 2 colaborador 3 sala 1
```

```
projeto 2 colaborador 4 sala 1
```

```
projeto 2 colaborador 5 sala 1
```

▼ Exemplos teste para cada restrição

1. Só um dos quatro colaboradores (o líder) pode estar presente, logo não haverá reunião:

```
horario = pywraplp.Solver.CreateSolver('SCIP')
```

```
x = {}
```

```
P,C,S,T,D = 1,4,1,1,1
```

```
colabs = {1:[1,2,3,4]}
```

```
lids = {1:2}
```

```
reuns = {1:1}
```

```
disps = {1:[(1,1)],2:[],3:[],4:[]}
```

```
horarioStartUp(colabs,lids,reuns,disps)
```

Não é possível construir horário pois não está de acordo com todas as restrições

2. O colaborador 3 não está disponível, logo não vai à reunião:

```
horario = pywraplp.Solver.CreateSolver('SCIP')
```

```
x = {}
```

```
P,C,S,T,D = 1,3,1,1,1
```

```
colabs = {1:[1,2,3]}
```

```
lids = {1:1}
```

```
reuns = {1:1}
```

```
disps = {1:[(1,1)],2:[(1,1)],3:[]}
```

```
horarioStartUp(colabs,lids,reuns,disps)
```

```
DIA 1
```

```
TEMPO 1
```

```
projeto 1 colaborador 1 sala 1
```

```
projeto 1 colaborador 2 sala 1
```

3. O colaborador 1 não consegue estar presente em ambas as reuniões do projeto 1 e 2 ao mesmo tempo:

```

horario = pywraplp.Solver.CreateSolver('SCIP')
x = {}
P,C,S,T,D = 2,2,2,1,1
colabs = {1:[1,2],2:[1,2]}
lids = {1:1,2:1}
reuns = {1:1,2:1}
disps = {1:[(1,1)],2:[]}

```

```

horarioStartUp(colabs,lids,reuns,disps)

```

Não é possível construir horário pois não está de acordo com todas as restrições

4. Existem duas reuniões diferentes para uma única sala, e para um único slot (tempo,dia):

```

horario = pywraplp.Solver.CreateSolver('SCIP')
x = {}
P,C,S,T,D = 2,4,1,1,1
colabs = {1:[1,2],2:[3,4]}
lids = {1:1,2:3}
reuns = {1:1,2:1}
disps = {1:[(1,1)],2:[(1,1)],3:[(1,1)],4:[(1,1)]}

```

```

horarioStartUp(colabs,lids,reuns,disps)

```

Não é possível construir horário pois não está de acordo com todas as restrições

5. O líder (colaborador 1) não está disponível para todas as reuniões do seu projeto:

```

horario = pywraplp.Solver.CreateSolver('SCIP')
x = {}
P,C,S,T,D = 1,3,1,2,2
colabs = {1:[1,2,3]}
lids = {1:1}
reuns = {1:2}
disps = {1:[(1,1)],2:[(1,1),(2,2)],3:[(1,1),(2,2)]}

```

```

horarioStartUp(colabs,lids,reuns,disps)

```

Não é possível construir horário pois não está de acordo com todas as restrições

Outros exemplos de horários que mostram o bom funcionamento do programa:

```

horario = pywraplp.Solver.CreateSolver('SCIP')
x = {}
P,C,S,T,D = 2,5,1,2,1
colabs = {1:[1,2,5],2:[3,4]}
lids = {1:2,2:4}
reuns = {1:1,2:1}

```

```
disps = {1:[(1,1)],2:[(1,1)],3:[(2,1)],4:[(2,1)],5:[]}
```

```
horarioStartUp(colabs,lids,reuns,disps)
```

```
DIA 1
```

```
TEMPO 1
```

```
projeto 1 colaborador 1 sala 1
```

```
projeto 1 colaborador 2 sala 1
```

```
TEMPO 2
```

```
projeto 2 colaborador 3 sala 1
```

```
projeto 2 colaborador 4 sala 1
```

```
horario = pywraplp.Solver.CreateSolver('SCIP')
```

```
x = {}
```

```
P,C,S,T,D = 1,3,1,3,1
```

```
colabs = {1:[1,2,3]}
```

```
lids = {1:2}
```

```
reuns = {1:3}
```

```
disps = {1:[(1,1),(3,1)],2:[(1,1),(2,1),(3,1)],3:[(2,1),(3,1)]}
```

```
horarioStartUp(colabs,lids,reuns,disps)
```

```
DIA 1
```

```
TEMPO 1
```

```
projeto 1 colaborador 1 sala 1
```

```
projeto 1 colaborador 2 sala 1
```

```
TEMPO 2
```

```
projeto 1 colaborador 2 sala 1
```

```
projeto 1 colaborador 3 sala 1
```

```
TEMPO 3
```

```
projeto 1 colaborador 1 sala 1
```

```
projeto 1 colaborador 2 sala 1
```

```
projeto 1 colaborador 3 sala 1
```

```
horario = pywraplp.Solver.CreateSolver('SCIP')
```

```
x = {}
```

```
P,C,S,T,D = 3,7,1,2,3
```

```
colabs = {1:[1,2],2:[3,4],3:[5,6,7]}
```

```
lids = {1:1,2:3,3:6}
```

```
reuns = {1:1,2:1,3:2}
```

```
disps = {1:[(1,1)],2:[(1,1)],3:[(2,2)],4:[(2,2)],5:[(1,3),(2,3)],6:[(1,3),(2,3)],7:[(2,3)]}
```

```
horarioStartUp(colabs,lids,reuns,disps)
```

```
DIA 1
```

```
TEMPO 1
```

```
projeto 1 colaborador 1 sala 1
```

```
projeto 1 colaborador 2 sala 1
```

```
TEMPO 2
```

```
DIA 2
```

```
TEMPO 1
```

```
TEMPO 2
```

```
projeto 2 colaborador 3 sala 1
```

```
projeto 2 colaborador 4 sala 1
```

```
DIA 3
```

```
TEMPO 1
```

projeto 3 colaborador 5 sala 1
projeto 3 colaborador 6 sala 1
TEMPO 2
projeto 3 colaborador 5 sala 1
projeto 3 colaborador 6 sala 1
projeto 3 colaborador 7 sala 1