

Vespertino_Ensino_Medio_V3

June 22, 2022

```
[1]: import gurobipy as gp
import pandas as pd
import numpy as np
import ezodf

def read_ods(filename, sheet_no=0, header=0):
    tab = ezodf.opendoc(filename=filename).sheets[sheet_no]
    return pd.DataFrame({col[header].value: [x.value for x in col[header+1:]]
                        for col in tab.columns()})

#leitura da tabela
df = read_ods(filename='arquivo_vespertino_ensino_medio_semCPF_semNome.ods')
```

```
[2]: class Disciplina():
    def
    ↪ __init__(self, codigo, area_ensino, quantidade_aulas_semanais, numero_turma, inicio_aula, fim_aula,
    ↪
        self.codigo=codigo
        self.area_ensino=area_ensino
        self.quantidade_aulas_semanais=quantidade_aulas_semanais
        self.numero_turma=numero_turma
        self.inicio_aula=inicio_aula
        self.fim_aula=fim_aula
        self.professor=professor
        self.numero_maximo_diario=numero_maximo_diario
        self.numero_maximo_sequencia=numero_maximo_sequencia
        self.index=index

    def nome(self):
        switcher = {
            1344: "2º PROFESSOR - 20",
            628: "ARTE",
            612: "CIÊNCIAS",
            611: "RELIGIÃO",
            536: "FILOSOFIA",
            513: "QUÍMICA",
            475: "FÍSICA",
            437: "SOCIOLOGIA",
```

```

        401: "LITERATURA",
        319: "INGLÊS",
        307: "EDUCAÇÃO FÍSICA",
        304: "HISTÓRIA",
        302: "GEOGRAFIA",
        301: "MATEMÁTICA",
        255: "BIOLOGIA",
        202: "PORTUGUES",
        1: "LETRAS - ENS. FUND.- ANOS INICIAIS"
    }

    # get() method of dictionary data type returns
    # value of passed argument if it is present
    # in dictionary otherwise second argument will
    # be assigned as default value of passed argument
    return switcher.get(self.codigo, "code not found")

def nome_area_ensino(self):
    switcher = {
        1: "ENS.FUND.ANOS INICIAIS",
        2: "ENS.FUND.ANOS FINAIS",
        3: "ENSINO MÉDIO",
    }

    # get() method of dictionary data type returns
    # value of passed argument if it is present
    # in dictionary otherwise second argument will
    # be assigned as default value of passed argument
    return switcher.get(self.area_ensino, "code not found")

def get_codigo(self):
    return self.codigo

def get_area_ensino(self):
    return self.area_ensino

def get_quantidade_aulas_semanais(self):
    return self.quantidade_aulas_semanais

def get_numero_turma(self):
    return self.numero_turma

def get_inicio_aula(self):
    return self.inicio_aula

def get_fim_aula(self):
    return self.fim_aula

```

```

def get_professor(self):
    return self.professor

def get_numero_maximo_diario(self):
    return self.numero_maximo_diario

def get_numero_maximo_sequenciar(self):
    return self.numero_maximo_sequencia

def get_index(self):
    return self.index

def get_horarios(self):
    horarios=[]
    if (self.inicio_aula == "07:45" and self.fim_aula=="11:45"):
        for x in range (2,7):
            for y in range (1,7):
                horarios.append([x,y])
    if (self.inicio_aula == "13:30" and self.fim_aula=="17:30"):
        horarios=[]
        for x in range (2,7):
            for y in range (7,13):
                horarios.append([x,y])
    return horarios

```

```

[3]: class Turma():
    def __init__(self,codigo,area_ensino,serie,index):
        self.codigo=codigo
        self.area_ensino=area_ensino
        self.serie=serie
        self.index=index

    def nome_area_ensino(self):
        switcher = {
            1: "ENS.FUND.ANOS INICIAIS",
            2: "ENS.FUND.ANOS FINAIS",
            3: "ENSINO MÉDIO",
        }

        # get() method of dictionary data type returns
        # value of passed argument if it is present
        # in dictionary otherwise second argument will
        # be assigned as default value of passed argument
        return switcher.get(self.area_ensino, "code not found")

```

```

def get_codigo(self):
    return self.codigo

def get_area_ensino(self):
    return self.area_ensino

def get_serie(self):
    return self.serie

def get_index(self):
    return self.index

```

```

[4]: class Professor():
    def __init__(self, codigo, nome, disciplinas, index):
        self.codigo=codigo
        self.nome=nome
        self.disciplinas=disciplinas
        self.index=index

    def get_codigo(self):
        return self.codigo

    def get_nome(self):
        return self.nome

    def get_carga_horaria(self):
        carga_horaria=0
        for x in range(len(self.disciplinas)):
            indice = self.disciplinas[x]
            disciplina=objs[indice]
            carga_horaria_temp = disciplina.get_quantidade_aulas_semanais()
            carga_horaria = carga_horaria + carga_horaria_temp
        return carga_horaria

    def get_disciplinas(self):
        return self.disciplinas

    def get_index(self):
        return self.index

    def get_horarios(self):
        horarios=[]

        for x in range(len(self.disciplinas)):
            indice = self.disciplinas[x]
            disciplina=objs[indice]

```

```

        if (disciplina.get_inicio_aula() == "07:45" and disciplina.
→get_fim_aula()=="11:45"):
            for x in range (2,7):
                for y in range (1,7):
                    horarios.append([x,y])
        if (disciplina.get_inicio_aula() == "13:30" and disciplina.
→get_fim_aula()=="17:30"):
            for x in range (2,7):
                for y in range (7,13):
                    horarios.append([x,y])

    temp_list = []
    for i in horarios :
        if i not in temp_list:
            temp_list.append(i)
    horarios = temp_list

    return horarios

```

```

[5]: class Grupo():
    def __init__(self,codigo,nome,maximo,disciplinas,index):
        self.codigo=codigo
        self.nome=nome
        self.maximo=maximo
        self.disciplinas=disciplinas
        self.index=index

    def get_codigo(self):
        return self.codigo

    def get_nome(self):
        return self.nome

    def get_maximo(self):
        return self.maximo

    def get_disciplinas(self):
        return self.disciplinas

    def get_index(self):
        return self.index

```

```

[6]: class Horario():
    def __init__(self,codigo,dia_semana,inicio,turno,index):
        self.codigo=codigo
        self.dia_semana=dia_semana
        self.inicio=inicio

```

```

        self.turno=turno
        self.index=index

    def get_codigo(self):
        return self.codigo

    def get_dia_semana(self):
        return self.dia_semana

    def get_inicio(self):
        return self.inicio

    def get_turno(self):
        return self.turno

    def get_index(self):
        return self.index

```

[7]: *#retorna uma lista com objetos disciplinas que sao lecionadas para a turma*

```

def disciplinas_turma(turma, disciplinas):
    retorno=[]
    for x in range(len(disciplinas)):
        disciplina=disciplinas[x]
        if (disciplina.get_numero_turma() == turma.get_codigo()):
            retorno.append(disciplina)

    return retorno

```

[8]: *#listar todas as turmas(objetos) de um professor*

```

def listar_turmas_professor(professor, objs_turmas, objs):
    retorno = []
    disciplinas_professor=professor.get_disciplinas()
    for x in range(len(disciplinas_professor)):
        indice=disciplinas_professor[x]
        disciplina=objs[indice]
        for y in range(len(objs_turmas)):
            turma=objs_turmas[y]
            if (disciplina.get_numero_turma() == turma.get_codigo()):
                retorno.append(turma)

    temp_list = []
    for i in retorno :
        if i not in temp_list:
            temp_list.append(i)

    retorno = temp_list

```

```
return retorno
```

```
[9]: #listar todos os horarios(objetos) de um professor
def listar_horarios_professor(professor,objs_horarios):#listar todas os
    ↪horarios(objetos) de um professor
    retorno = []
    dupla=professor.get_horarios()
    for x in range(len(objs_horarios)):
        horario=objs_horarios[x]
        for y in range(len(dupla)):
            dupla_codigo=dupla[y][1]
            dupla_dia=dupla[y][0]
            if(horario.get_codigo()==dupla_codigo and dupla_dia == horario.
    ↪get_dia_semana()):
                retorno.append(horario)

    temp_list = []
    for i in retorno :
        if i not in temp_list:
            temp_list.append(i)

    retorno = temp_list

    return retorno
```

```
[10]: #listar todos os professores(objetos) de uma turma
def listar_professores_turma(turma,professores,objs):
    retorno=[]
    for x in range(len(professores)):
        professor=professores[x]
        disciplinas_professor=professor.get_disciplinas()
        for y in range(len(disciplinas_professor)):
            indice=disciplinas_professor[y]
            disciplina=objs[indice]
            if (disciplina.get_numero_turma() == turma.get_codigo()):
                retorno.append(professor)

    return retorno
```

```
[11]: #listar todos os horarios de uma turma
def listar_horarios_turma(turma,professores,objs):
    retorno=[]
    for x in range(len(professores)):
        professor=professores[x]
        disciplinas_professor=professor.get_disciplinas()
        for y in range(len(disciplinas_professor)):
            horarios=[]
            indice=disciplinas_professor[y]
```

```

        disciplina=objs[indice]
        if (disciplina.get_numero_turma() == turma.get_codigo()):
            horarios_disciplina=disciplina.get_horarios()
            for z in range(len(horarios_disciplina)):
                retorno.append(horarios_disciplina[z])

temp_list = []

for i in retorno :
    if i not in temp_list:
        temp_list.append(i)

retorno = temp_list
return retorno

```

[12]: *#dado uma dupla dia e slot retorna o objeto correspondente a este horario*

```

def get_obj_horario(lista,objs_horarios):
    retorno=[]
    for y in range(len(lista)):
        for x in range(len(objs_horarios)):
            horario=objs_horarios[x]
            if(lista[y]==[horario.get_dia_semana(),horario.get_codigo()]):
                retorno.append(horario)
    return retorno

```

[13]: *#retorna lista de indices das disciplinas*

```

def listar_disciplinas(objs):
    retorno=[]
    for x in range(len(objs)):
        disciplina=objs[x]
        retorno.append(disciplina.get_index())
    return retorno

```

[14]: *#retorna lista de indices dos professores*

```

def listar_professores(objs):
    retorno=[]
    for x in range(len(objs)):
        professor=objs[x]
        retorno.append(professor.get_index())
    return retorno

```

[15]: *#retorna lista de indices dos horarios*

```

def listar_horarios(objs):
    retorno=[]
    for x in range(len(objs)):
        horario=objs[x]
        retorno.append(horario.get_index())
    return retorno

```


[16]: *#retorna lista de indices das turmas*

```
def listar_turmas(objs):  
    retorno=[]  
    for x in range(len(objs)):  
        turma=objs[x]  
        retorno.append(turma.get_index())  
    return retorno
```

[17]: *#retorna lista de indices das disciplinas, horarios e professores de um objeto*
↪ turma

```
def gerar_lista_indices_turma(turma,objs,objs_professores,objs_horarios):  
  
    disciplinas = listar_disciplinas(objs)  
    professores = listar_professores(objs_professores)  
    horarios = listar_horarios(objs_horarios)  
    listaIndiceDisciplina= []  
    listaIndiceProfessor=[]  
    listaIndiceHorario=[]  
  
    indiceTurma= turma.get_index() #turma  
    disciplinas_da_turma=disciplinas_turma(turma,objs)  
    professor_da_turma=listar_professores_turma(turma,objs_professores,objs)  
    temp_horarios_da_turma=listar_horarios_turma(turma,objs_professores,objs)  
    horarios_da_turma=get_obj_horario(temp_horarios_da_turma,objs_horarios)  
    for y in range(len(disciplinas_da_turma)):  
        disciplina=disciplinas_da_turma[y]  
        indiceDisciplina=disciplina.get_index()  
        listaIndiceDisciplina.append(indiceDisciplina)  
        for z in range(len(professor_da_turma)):  
            professor=professor_da_turma[z]  
            indiceProfessor=professor.get_index()  
            listaIndiceProfessor.append(indiceProfessor)  
            for w in range(len(horarios_da_turma)):  
                horario=horarios_da_turma[w]  
                indiceHorario=horario.get_index()  
                listaIndiceHorario.append(indiceHorario)  
  
    temp_list = []  
    for i in listaIndiceDisciplina :  
        if i not in temp_list:  
            temp_list.append(i)  
  
    listaIndiceDisciplina = temp_list  
  
    temp_list = []  
    for i in listaIndiceProfessor :  
        if i not in temp_list:
```

```

        temp_list.append(i)

    listaIndiceProfessor = temp_list

    temp_list = []
    for i in listaIndiceHorario :
        if i not in temp_list:
            temp_list.append(i)

    listaIndiceHorario = temp_list

    return (listaIndiceDisciplina,listaIndiceProfessor,listaIndiceHorario)

```

```

[18]: #retorna lista de indices das disciplinas,horarios e turmas de um objeto
      ↪ professor
def gerar_lista_indices_professor(professor,objs,objs_turmas,objs_horarios):
    turmas = listar_turmas(objs_turmas)
    disciplinas = listar_disciplinas(objs)
    horarios = listar_horarios(objs_horarios)
    listaIndiceDisciplina= professor.get_disciplinas()
    listaIndiceTurma=[]
    listaIndiceHorario=[]

    indiceProfessor= professor.get_index() #professor

    turmas_professor=listar_turmas_professor(professor,objs_turmas,objs)
    horarios_do_professor=listar_horarios_professor(professor,objs_horarios)

    for z in range(len(turmas_professor)):
        turma=turmas_professor[z]
        indiceTurma=turma.get_index()
        listaIndiceTurma.append(indiceTurma)
    for w in range(len(horarios_do_professor)):
        horario=horarios_do_professor[w]
        indiceHorario=horario.get_index()
        listaIndiceHorario.append(indiceHorario)

    temp_list = []
    for i in listaIndiceDisciplina :
        if i not in temp_list:
            temp_list.append(i)

    listaIndiceDisciplina = temp_list

    temp_list = []

```

```

for i in listaIndiceTurma :
    if i not in temp_list:
        temp_list.append(i)

listaIndiceTurma = temp_list

temp_list = []
for i in listaIndiceHorario :
    if i not in temp_list:
        temp_list.append(i)

listaIndiceHorario = temp_list

return (listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario)

```

```

[19]: #criacao dos objetos disciplinas com os dados da tabela
lista_disciplinas = []
for i in df.index:
    lista_disciplinas.append([df['Código da disc. '][i],df['Código da área de_
↳ ensino'][i],df['Qtd de aulas na semana'][i],df['Número da_
↳ turma'][i],df['Início das aulas'][i],df['Fim das aulas'][i],df['Nome'][i]])

temp_list = []

for i in lista_disciplinas :
    if i not in temp_list:
        temp_list.append(i)

lista_disciplinas = temp_list

index=-1
objs = list()
for x in range(len(lista_disciplinas)):
    codigo = lista_disciplinas[x][0]
    area_ensino = lista_disciplinas[x][1]
    quantidade_aulas_semanais = lista_disciplinas[x][2]
    numero_turma = lista_disciplinas[x][3]
    inicio_aula = lista_disciplinas[x][4]
    fim_aula= lista_disciplinas[x][5]
    professor= lista_disciplinas[x][6]
    numero_maximo_diario= 3
    numero_maximo_sequencia= 3
    index=x
    objs.
↳ append(Disciplina(codigo,area_ensino,quantidade_aulas_semanais,numero_turma,inicio_aula,fim

```

```
[20]: #criacao dos objetos turmas com os dados da tabela
lista_turmas = []
for i in df.index:
    lista_turmas.append([df['Número da turma'][i],df['Código da área de_
→ ensino'][i],df['Etapa (séries)'][i]])

temp_list = []

for i in lista_turmas :
    if i not in temp_list:
        temp_list.append(i)

lista_turmas = temp_list

index=-1
objs_turmas = list()
for x in range(len(lista_turmas)):
    codigo = lista_turmas[x][0]
    area_ensino = lista_turmas[x][1]
    serie = lista_turmas[x][2]
    index=x
    objs_turmas.append(Turma(codigo,area_ensino,serie,index))
```

```
[21]: #criacao dos objetos professores com os dados da tabela
lista_professor = []
for i in df.index:
    lista_professor.append([df['Identificador do prof. '][i],df['Nome'][i]])

temp_list = []

for i in lista_professor :
    if i not in temp_list:
        temp_list.append(i)

lista_professor = temp_list

index=-1
objs_professores = list()
for x in range(len(lista_professor)):
    lista_disciplinas=[]
    codigo = lista_professor[x][0]
    nome = lista_professor[x][1]
    for y in range(len(objs)):
        disciplina=objs[y]
        if(nome==disciplina.get_professor()):
            lista_disciplinas.append(disciplina.get_index())
    index=x
```

```
objs_professores.append(Professor(codigo,nome,lista_disciplinas,index))
```

```
[22]: #criacao dos objetos grupos de disciplinas, nao utilizado nesta versao do
      ↪ modelo, pois nao foram adicionados restricoes relacionadas aos grupos
lista_0=["LÍNGUA PORTUGUESA E LITERATURA","LÍNGUA ESTRANGEIRA - INGLÊS","LÍNGUA
      ↪ PORTUGUESA","LETRAS - ENS. FUND.- ANOS INICIAIS","EDUCAÇÃO FÍSICA","ARTE"]
lista_1=["MATEMÁTICA"]
lista_2=["BIOLOGIA","QUÍMICA","FÍSICA","CIÊNCIAS"]
lista_3=["GEOGRAFIA","HISTÓRIA","SOCIOLOGIA","FILOSOFIA","ENSINO RELIGIOSO"]
grupo_0=[0,"Linguagens e suas Tecnologias",4,lista_0,0]
grupo_1=[1,"Matemática e suas Tecnologias",4,lista_1,1]
grupo_2=[2,"Ciências da Natureza e suas Tecnologias",4,lista_2,2]
grupo_3=[3,"Ciências Humanas e Sociais Aplicadas",4,lista_3,3]
lista_grupo = [grupo_0,grupo_1,grupo_2,grupo_3]

index=-1
objs_grupos = list()
for x in range(len(lista_grupo)):
    grupo_temp = lista_grupo[x]
    for y in range(len(grupo_temp)):
        codigo=grupo_temp[0]
        nome=grupo_temp[1]
        maximo=grupo_temp[2]
        disciplinas=grupo_temp[3]
        index=grupo_temp[4]
        objs_grupos.append(Grupo(codigo,nome,maximo,disciplinas,index))
```

```
[23]: #criacao dos objetos horarios com os dados da tabela
index=-1
objs_horarios = list()
for x in range (2,7):
    dia_semana=x
    for y in range (1,13):
        index=index+1
        codigo=y
        if(y>=1 and y<7):
            turno="Matutino"
            if(y==1):
                inicio="07:45"
            if(y==2):
                inicio="08:15"
            if(y==3):
                inicio="08:45"
            if(y==4):
                inicio="09:30"
            if(y==5):
                inicio="10:15"
```

```

        if(y==6):
            inicio="11:00"
            objs_horarios.append(Horario(codigo,dia_semana,inicio,turno,index))
    if(y>=7 and y<13):
        turno="Vespertino"
        if(y==7):
            inicio="13:30"
        if(y==8):
            inicio="14:15"
        if(y==9):
            inicio="15:00"
        if(y==10):
            inicio="15:35"
        if(y==11):
            inicio="16:15"
        if(y==12):
            inicio="16:50"
            objs_horarios.append(Horario(codigo,dia_semana,inicio,turno,index))
    if(y>=13):
        turno="Noturno"
        inicio="18:30"
        objs_horarios.append(Horario(codigo,dia_semana,inicio,turno,index))

```

```

[24]: #Inicializacao do modelo
model = gp.Model("Problema Horario")

```

Academic license - for non-commercial use only - expires 2022-07-28
 Using license file C:\Users\Asus\gurobi.lic

```

[25]: #Variaveis de decisao

#conjunto de indices1:geracao de indices para as possibilidades de alocao para
→ cada turma os indices de disciplina,professor e horario
turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)
variaveis= []

for x in range(len(turmas)):
    indiceTurma=0
    turma = objs_turmas[x]
    indiceTurma= turma.get_index() #turma
    disciplinas_da_turma=disciplinas_turma(turma,objs)
    professor_da_turma=listar_professores_turma(turma,objs_professores,objs)
    temp_horarios_da_turma=listar_horarios_turma(turma,objs_professores,objs)
    horarios_da_turma=get_obj_horario(temp_horarios_da_turma,objs_horarios)

```

```

    for y in range(len(disciplinas_da_turma)):
        disciplina=disciplinas_da_turma[y]
        indiceDisciplina=disciplina.get_index()
        for z in range(len(professor_da_turma)):
            professor=professor_da_turma[z]
            indiceProfessor=professor.get_index()
            for w in range(len(horarios_da_turma)):
                horario=horarios_da_turma[w]
                indiceHorario=horario.get_index()
                variaveis.
        ↪append((indiceTurma,indiceDisciplina,indiceProfessor,indiceHorario))

```

[26]: *#conjunto de indices2:geracao de indices para as possibilidades de alocao para*
↪cada professor os indices de disciplina,turma e horario

```

turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)

variaveis_rh2=[]
for z in range(len(professores)):
    p=0
    professor = objs_professores[z]
    ↪
    ↪listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(pro
    p=professor.get_index()
    for h in listaIndiceHorario:
        for t in listaIndiceTurma:
            for d in listaIndiceDisciplina:
                variaveis_rh2.append((t,d,p,h))
for w in variaveis_rh2:
    variaveis.append(w)

#realizando interseccao de conjunto de indices1 e conjunto de indices2
set_variaveis=set(variaveis)
variaveis=list(set_variaveis)

```

[27]: *#instanciando as variaveis relacionadas a alocao de horarios ao modelo*

```

x = model.addVars(variaveis,vtype=gp.GRB.BINARY)

#conjunto de indices3:relacionados a janelas e dias de trabalho

#janelas
segunda_tarde=[]
terca_tarde=[]
quarta_tarde=[]
quinta_tarde=[]

```

```

sexta_tarde=[]

#dias
segunda=[]
terca=[]
quarta=[]
quinta=[]
sexta=[]

for i in range(len(professores)):
    #dias
    segunda.append(i)
    terca.append(i)
    quarta.append(i)
    quinta.append(i)
    sexta.append(i)

    #janelas
    for h in range(6,10):
        segunda_tarde.append((i,h))
    for h in range(18,22):
        terca_tarde.append((i,h))
    for h in range(30,34):
        quarta_tarde.append((i,h))
    for h in range(42,46):
        quinta_tarde.append((i,h))
    for h in range(54,58):
        sexta_tarde.append((i,h))

#instanciando as variaveis relacionadas a janelas e dias da semana em que o
    ↪ professor leciona
#janela
janela_segunda_tarde = model.addVars(segunda_tarde,vtype=gp.GRB.BINARY)
janela_terca_tarde = model.addVars(terca_tarde,vtype=gp.GRB.BINARY)
janela_quarta_tarde = model.addVars(quarta_tarde,vtype=gp.GRB.BINARY)
janela_quinta_tarde = model.addVars(quinta_tarde,vtype=gp.GRB.BINARY)
janela_sexta_tarde = model.addVars(sexta_tarde,vtype=gp.GRB.BINARY)

#dia
dia_segunda=model.addVars(segunda,vtype=gp.GRB.BINARY)
dia_terca=model.addVars(terca,vtype=gp.GRB.BINARY)
dia_quarta=model.addVars(quarta,vtype=gp.GRB.BINARY)
dia_quinta=model.addVars(quinta,vtype=gp.GRB.BINARY)

```



```
dia_sexta=model.addVars(sexta,vtype=gp.GRB.BINARY)
```

```
[28]: #conjunto de restricoes necessarias para identificar se um professor leciona em
      ↪um dia na semana
for z in range(len(professores)):
    p=0
    professor = objs_professores[z]
    ↪
    ↪listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(pro
    p=professor.get_index()

    # Constants
    eps = 0.0001
    M = 10 + eps # smallest possible given bounds on x and y
    #seg
    # Model if x>y then b = 1, otherwise b = 0
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in ↪
    ↪listaIndiceDisciplina for h in range (6,12)) >= 0 + eps - M * (1 - ↪
    ↪dia_segunda[p]), name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_seg1")
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in ↪
    ↪listaIndiceDisciplina for h in range (6,12)) <= 0 + M * dia_segunda[p], ↪
    ↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_seg2")

    #terca
    # Model if x>y then b = 1, otherwise b = 0
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in ↪
    ↪listaIndiceDisciplina for h in range (18,24)) >= 0 + eps - M * (1 - ↪
    ↪dia_terca[p]), name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_ter1")
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in ↪
    ↪listaIndiceDisciplina for h in range (18,24)) <= 0 + M * dia_terca[p], ↪
    ↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_ter2")

    #quarta
    # Model if x>y then b = 1, otherwise b = 0
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in ↪
    ↪listaIndiceDisciplina for h in range (30,36)) >= 0 + eps - M * (1 - ↪
    ↪dia_quarta[p]), name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_qua1")
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in ↪
    ↪listaIndiceDisciplina for h in range (30,36)) <= 0 + M * dia_quarta[p], ↪
    ↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_qua")

    #quinta
    # Model if x>y then b = 1, otherwise b = 0
```

```

    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina for h in range (42,48)) >= 0 + eps - M * (1 -
↪dia_quinta[p]), name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_qui1")
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina for h in range (42,48)) <= 0 + M * dia_quinta[p],
↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_qui2")

    #sexta
    # Model if x>y then b = 1, otherwise b = 0
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina for h in range (54,60)) >= 0 + eps - M * (1 -
↪dia_sexta[p]), name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_sex1")
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina for h in range (54,60)) <= 0 + M * dia_sexta[p],
↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"bigM_sex2")

```

[29]:

```

#conjunto de restricoes necessarias para janelas de um professor
for z in range(len(professores)):
    p=0
    professor = objs_professores[z]

    ↪
    ↪listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(pro
    p=professor.get_index()
    # Constants
    eps = 0.0001
    M = 10 + eps # smallest possible given bounds on x and y

    #segunda
    for h in range(6,10):
        # Model if x>y then b = 1, otherwise b = 0
        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↪listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↪for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↪quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina)))) >= 0 + eps - M * (1 - janela_segunda_tarde[p,h]),
↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_seg1")
        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↪listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↪for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↪quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina)))) <= 0 + M * janela_segunda_tarde[p,h],
↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_seg2")

    #terca
    for h in range(18,22):

```

```

        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↳ listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↳ for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↳ quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↳ listaIndiceDisciplina)))) >= 0 + eps - M * (1 - janela_terca_tarde[p,h]),
↳ name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_ter1")

        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↳ listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↳ for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↳ quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↳ listaIndiceDisciplina)))) <= 0 + M * janela_terca_tarde[p,h],
↳ name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_ter2")

    #quarta
    for h in range(30,34):
        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↳ listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↳ for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↳ quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↳ listaIndiceDisciplina)))) >= 0 + eps - M * (1 - janela_quarta_tarde[p,h]),
↳ name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_qua1")

        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↳ listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↳ for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↳ quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↳ listaIndiceDisciplina)))) <= 0 + M * janela_quarta_tarde[p,h],
↳ name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_qua2")

    #quinta
    for h in range(42,46):
        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↳ listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↳ for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↳ quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↳ listaIndiceDisciplina)))) >= 0 + eps - M * (1 - janela_quinta_tarde[p,h]),
↳ name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_qui1")

        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↳ listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↳ for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↳ quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↳ listaIndiceDisciplina)))) <= 0 + M * janela_quinta_tarde[p,h],
↳ name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_qui2")

    #sexta
    for h in range(54,58):

```

```

        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↪listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↪for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↪quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina)))) >= 0 + eps - M * (1 - janela_sexta_tarde[p,h]),
↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_sex1")
        model.addConstr((1 - ((1 - gp.quicksum(x[t,d,p,h] for t in
↪listaIndiceTurma for d in listaIndiceDisciplina)) + gp.quicksum(x[t,d,p,h+1]
↪for t in listaIndiceTurma for d in listaIndiceDisciplina) + (1 - gp.
↪quicksum(x[t,d,p,h+2] for t in listaIndiceTurma for d in
↪listaIndiceDisciplina)))) <= 0 + M * janela_sexta_tarde[p,h]),
↪name="RestricaoObjetivo_ProfessorIndex_"+str(p)+"HorarioIndex_"+str(h)+"bigM_sex2")

```

```

[30]: #RH1:Para cada turma,cada horário em uma dia da semana é ocupado por no máximo
↪uma disciplina nesta turma
turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)

for z in range(len(turmas)):
    t=0
    turma = objs_turmas[z]

    ↪
    ↪listaIndiceDisciplina,listaIndiceProfessor,listaIndiceHorario=gerar_lista_indices_turma(turmas[z])
    t=turma.get_index()
    for h in listaIndiceHorario:
        model.addConstr(gp.quicksum(x[t,d,p,h] for d in listaIndiceDisciplina
↪for p in listaIndiceProfessor)<=
↪1,"RH1_TurmaIndex_"+str(t)+"__HorarioIndex_"+str(h))

```

```

[31]: #RH2:Cada professor pode lecionar no máximo uma disciplina em um mesmo horário
↪que ele está disponível.
turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)

for z in range(len(professores)):
    p=0
    professor = objs_professores[z]

    ↪
    ↪listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(professores[z])
    p=professor.get_index()
    for h in listaIndiceHorario:

```

```

        model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in
↪ listaIndiceDisciplina)<=
↪ 1,"RH2_ProfessorIndex_"+str(p)+"__HorarioIndex_"+str(h))

```

[32]: *#RH3: Um professor não pode ultrapassar uma determinada quantidade máxima de*
↪ aulas semanais.

```

turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)

for z in range(len(professores)):
    p=0
    professor = objs_professores[z]
    ↪
    ↪ listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(prof
    p=professor.get_index()
    aulas=professor.get_carga_horaria()
    model.addConstr(gp.quicksum(x[t,d,p,h] for t in listaIndiceTurma for d in
↪ listaIndiceDisciplina for h in
↪ listaIndiceHorario)==aulas,"RH3_ProfessorIndex_"+str(p))

```

[33]: *#RH4: Uma turma t deve ter alocado exatamente o número de aulas previsto para a*
↪ disciplina d lecionado pelo professor p,
#conforme sua grade curricular.

```

turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)

for z in range(len(turmas)):
    t=0
    turma = objs_turmas[z]
    ↪
    ↪ listaIndiceDisciplina,listaIndiceProfessor,listaIndiceHorario=gerar_lista_indices_turma(tur
    t=turma.get_index()
    for d in listaIndiceDisciplina:
        p=0
        disciplina=objs[d]
        professor_nome=disciplina.get_professor()
        for prof in listaIndiceProfessor:
            temp_professor=objs_professores[prof]
            temp_nome=temp_professor.get_nome()
            if(professor_nome==temp_nome):
                p=prof
                break

```

```

aulas=disciplina.get_quantidade_aulas_semanais()
model.addConstr(gp.quicksum(x[t,d,p,h] for h in listaIndiceHorario)==
↪aulas,"RH4_TurmaIndex_"+str(t)+"__DisciplinaIndex_"+str(d)+"__ProfessorIndex_"+str(p))

```

```

[34]: #RH5:Cada disciplina de uma turma é dada no maximo 1 vez por dia da semana
turmas = listar_turmas(objs_turmas)
disciplinas = listar_disciplinas(objs)
professores = listar_professores(objs_professores)
horarios = listar_horarios(objs_horarios)
diadasemana=[(0,1,2,3,4,5,6,7,8,9,10,11),(12,13,14,15,16,17,18,19,20,21,22,23),(24,25,26,27,28,29,30,31)]

for z in range(len(turmas)):
    t=0
    turma = objs_turmas[z]
    ↪
    ↪listaIndiceDisciplina,listaIndiceProfessor,listaIndiceHorario=gerar_lista_indices_turma(turma)
    t=turma.get_index()
    for dia in range(len(diadasemana)):
        indices_do_dia=diadasemana[dia]
        myList=indices_do_dia
        for indice in indices_do_dia:
            if(indice not in listaIndiceHorario):
                valueToBeRemoved = indice
                myList = [value for value in myList if value !=
↪valueToBeRemoved]
            for d in listaIndiceDisciplina:
                model.addConstr(gp.quicksum(x[t,d,p,h] for p in
↪listaIndiceProfessor for h in myList)<=
↪1,"RH6_TurmaIndex_"+str(t)+"__DisciplinaIndex_"+str(d)+"__DiaDaSemanaIndex_"+str(dia))

```

```

[35]: #funcao objetivo minimizar janelas para todos os professores
#model.setObjective(gp.quicksum(janela_segunda_tarde[p,h] for p in
↪range(len(professores)) for h in range(6,10))+gp.
↪quicksum(janela_terca_tarde[p,h] for p in range(len(professores)) for h in
↪range(18,22))+gp.quicksum(janela_quarta_tarde[p,h] for p in
↪range(len(professores)) for h in range(30,34))+gp.
↪quicksum(janela_quinta_tarde[p,h] for p in range(len(professores)) for h in
↪range(42,46))+gp.quicksum(janela_sexta_tarde[p,h] for p in
↪range(len(professores)) for h in range(54,58)),sense = gp.GRB.MINIMIZE)

#funcao objetivo minimizar dias de aulas para todos os professores
#model.setObjective(gp.quicksum(dia_segunda[p] + dia_terca[p] + dia_quarta[p] +
↪dia_quinta[p] + dia_sexta[p] for p in range(len(professores))),sense = gp.
↪GRB.MINIMIZE)

#funcao objetivo combinando

```

```

model.setObjective(gp.quicksum(janela_segunda_tarde[p,h] for p in
    ↳range(len(professores)) for h in range(6,10))+gp.
    ↳quicksum(janela_terca_tarde[p,h] for p in range(len(professores)) for h in
    ↳range(18,22))+gp.quicksum(janela_quarta_tarde[p,h] for p in
    ↳range(len(professores)) for h in range(30,34))+gp.
    ↳quicksum(janela_quinta_tarde[p,h] for p in range(len(professores)) for h in
    ↳range(42,46))+gp.quicksum(janela_sexta_tarde[p,h] for p in
    ↳range(len(professores)) for h in range(54,58))+4*gp.quicksum(dia_segunda[p]
    ↳+ dia_terca[p] + dia_quarta[p] + dia_quinta[p] + dia_sexta[p] for p in
    ↳range(len(professores))),sense = gp.GRB.MINIMIZE)

```

```
[36]: model.optimize()
```

Gurobi Optimizer version 9.1.2 build v9.1.2rc0 (win64)
 Thread count: 2 physical cores, 4 logical processors, using up to 4 threads
 Optimize a model with 1968 rows, 27450 columns and 103500 nonzeros
 Model fingerprint: 0xe40c94bc
 Variable types: 0 continuous, 27450 integer (27450 binary)
 Coefficient statistics:
 Matrix range [1e+00, 1e+01]
 Objective range [1e+00, 4e+00]
 Bounds range [1e+00, 1e+00]
 RHS range [1e+00, 2e+01]
 Presolve removed 338 rows and 25275 columns
 Presolve time: 0.18s
 Presolved: 1630 rows, 2175 columns, 16945 nonzeros
 Variable types: 0 continuous, 2175 integer (2175 binary)

 Root relaxation: objective 1.960000e+02, 2323 iterations, 0.24 seconds

Nodes		Current Node			Objective Bounds			Work	
Expl	Unexpl	Obj	Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
	0	0	196.00000	0	27	-	196.00000	-	1s
H	0	0			212.0000000	196.00000	7.55%	-	1s
H	0	0			210.0000000	196.00000	6.67%	-	1s
	0	0	196.00000	0	57	210.00000	196.00000	6.67%	1s
H	0	0			199.0000000	196.00000	1.51%	-	1s
H	0	0			197.0000000	196.00000	0.51%	-	2s
	0	0	196.00000	0	154	197.00000	196.00000	0.51%	2s
	0	0	196.00000	0	10	197.00000	196.00000	0.51%	2s
	0	0	196.00000	0	10	197.00000	196.00000	0.51%	2s
	0	0	196.00000	0	16	197.00000	196.00000	0.51%	2s
	0	0	196.00000	0	17	197.00000	196.00000	0.51%	4s
	0	0	196.00000	0	32	197.00000	196.00000	0.51%	4s
	0	0	196.00000	0	32	197.00000	196.00000	0.51%	4s
H	0	0			196.0000000	196.00000	0.00%	-	4s
	0	0	196.00000	0	32	196.00000	196.00000	0.00%	4s

Cutting planes:

MIR: 1
Zero half: 8
RLT: 14

Explored 1 nodes (26917 simplex iterations) in 4.76 seconds
Thread count was 4 (of 4 available processors)

Solution count 5: 196 197 199 ... 212

Optimal solution found (tolerance 1.00e-04)

Best objective 1.960000000000e+02, best bound 1.960000000000e+02, gap 0.0000%

```
[37]: #printar grade horaria professor
for z in range(len(professores)):
    p=0
    atual=0
    anterior=0
    professor = objs_professores[z]
    ↳
    ↳ listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(pro
    p=professor.get_index()
    contador=0
    print(" ")
    ↳
    ↳ print("-----")
    print("| Prof: ",professor.get_nome(),"|")
    for h in listaIndiceHorario:
        horario=objs_horarios[h]
        atual=horario.get_dia_semana()
        for t in listaIndiceTurma:
            turma=objs_turmas[t]
            for d in listaIndiceDisciplina:
                disciplina=objs[d]
                if (round(x[t,d,p,h].X)==1):
                    contador=contador+1
                    if(atual!=anterior):
                        ↳
    ↳ print("-----")
                                anterior=horario.get_dia_semana()
                                print("%15s %1d | %6s %2d | %12s %16s | %6s %4s |" %("Dia_
    ↳ da semana: ",horario.get_dia_semana(),"Slot: ",horario.
    ↳ get_codigo(),"Disciplina: ",disciplina.nome(),"Turma: ",round(turma.
    ↳ get_codigo()))
```



```

                                #print("/ Dia da Semana: ",horario.
→get_dia_semana(),"\N{DEGREE SIGN}","/ Slot: ",horario.get_codigo(),"/
→Disciplina: ",disciplina.nome(),"/ Turma: ",round(turma.get_codigo()),"/
→Aula: ",round(x[t,d,p,h].X),"/")
    print(" ")
    print("| Carga horária: ",contador,"|")
    print(" ")

```

| Prof: P1 |

Dia da semana:	2	Slot:	7	Disciplina:	SOCIOLOGIA	Turma:	202	
Dia da semana:	2	Slot:	8	Disciplina:	SOCIOLOGIA	Turma:	103	
Dia da semana:	2	Slot:	9	Disciplina:	SOCIOLOGIA	Turma:	302	
Dia da semana:	2	Slot:	10	Disciplina:	SOCIOLOGIA	Turma:	102	
Dia da semana:	2	Slot:	11	Disciplina:	SOCIOLOGIA	Turma:	203	

Dia da semana:	4	Slot:	7	Disciplina:	SOCIOLOGIA	Turma:	203	
Dia da semana:	4	Slot:	8	Disciplina:	SOCIOLOGIA	Turma:	202	
Dia da semana:	4	Slot:	11	Disciplina:	SOCIOLOGIA	Turma:	103	
Dia da semana:	4	Slot:	12	Disciplina:	SOCIOLOGIA	Turma:	102	

Dia da semana:	6	Slot:	8	Disciplina:	SOCIOLOGIA	Turma:	102	
Dia da semana:	6	Slot:	9	Disciplina:	SOCIOLOGIA	Turma:	302	
Dia da semana:	6	Slot:	10	Disciplina:	SOCIOLOGIA	Turma:	202	
Dia da semana:	6	Slot:	11	Disciplina:	SOCIOLOGIA	Turma:	103	
Dia da semana:	6	Slot:	12	Disciplina:	SOCIOLOGIA	Turma:	203	

| Carga horária: 14 |

| Prof: P2 |

Dia da semana:	2	Slot:	7	Disciplina:	LITERATURA	Turma:	103	
Dia da semana:	2	Slot:	8	Disciplina:	LITERATURA	Turma:	302	
Dia da semana:	2	Slot:	9	Disciplina:	LITERATURA	Turma:	203	
Dia da semana:	2	Slot:	12	Disciplina:	LITERATURA	Turma:	102	

Dia da semana:	3	Slot:	7	Disciplina:	LITERATURA	Turma:	102	
----------------	---	-------	---	-------------	------------	--------	-----	--

Dia da semana:	3	Slot:	10	Disciplina:	LITERATURA	Turma:	302	
Dia da semana:	3	Slot:	11	Disciplina:	LITERATURA	Turma:	203	
Dia da semana:	3	Slot:	12	Disciplina:	LITERATURA	Turma:	103	

Dia da semana:	4	Slot:	7	Disciplina:	LITERATURA	Turma:	102	
Dia da semana:	4	Slot:	8	Disciplina:	LITERATURA	Turma:	103	
Dia da semana:	4	Slot:	11	Disciplina:	LITERATURA	Turma:	203	
Dia da semana:	4	Slot:	12	Disciplina:	LITERATURA	Turma:	302	

Dia da semana:	6	Slot:	7	Disciplina:	LITERATURA	Turma:	302	
Dia da semana:	6	Slot:	8	Disciplina:	LITERATURA	Turma:	103	
Dia da semana:	6	Slot:	11	Disciplina:	LITERATURA	Turma:	203	
Dia da semana:	6	Slot:	12	Disciplina:	LITERATURA	Turma:	102	

| Carga horária: 16 |

| Prof: P3 |

Dia da semana:	2	Slot:	10	Disciplina:	MATEMÁTICA	Turma:	302	
----------------	---	-------	----	-------------	------------	--------	-----	--

Dia da semana:	3	Slot:	7	Disciplina:	MATEMÁTICA	Turma:	302	
----------------	---	-------	---	-------------	------------	--------	-----	--

Dia da semana:	4	Slot:	10	Disciplina:	MATEMÁTICA	Turma:	302	
----------------	---	-------	----	-------------	------------	--------	-----	--

Dia da semana:	6	Slot:	10	Disciplina:	MATEMÁTICA	Turma:	302	
----------------	---	-------	----	-------------	------------	--------	-----	--

| Carga horária: 4 |

| Prof: P4 |

Dia da semana:	2	Slot:	7	Disciplina:	ARTE	Turma:	302	
Dia da semana:	2	Slot:	8	Disciplina:	ARTE	Turma:	202	
Dia da semana:	2	Slot:	9	Disciplina:	ARTE	Turma:	102	
Dia da semana:	2	Slot:	10	Disciplina:	ARTE	Turma:	203	

Dia da semana:	3	Slot:	7	Disciplina:	ARTE	Turma:	203	
Dia da semana:	3	Slot:	11	Disciplina:	ARTE	Turma:	302	
Dia da semana:	3	Slot:	12	Disciplina:	ARTE	Turma:	102	

Dia da semana:	6	Slot:	7	Disciplina:	ARTE	Turma:	102	
Dia da semana:	6	Slot:	8	Disciplina:	ARTE	Turma:	302	
Dia da semana:	6	Slot:	12	Disciplina:	ARTE	Turma:	202	

| Carga horária: 10 |

| Prof: P5 |

Dia da semana:	4	Slot:	9	Disciplina:	BIOLOGIA	Turma:	302	
Dia da semana:	4	Slot:	10	Disciplina:	BIOLOGIA	Turma:	103	
Dia da semana:	4	Slot:	11	Disciplina:	BIOLOGIA	Turma:	102	

Dia da semana:	5	Slot:	10	Disciplina:	BIOLOGIA	Turma:	302	
Dia da semana:	5	Slot:	11	Disciplina:	BIOLOGIA	Turma:	102	
Dia da semana:	5	Slot:	12	Disciplina:	BIOLOGIA	Turma:	103	

Dia da semana:	6	Slot:	7	Disciplina:	BIOLOGIA	Turma:	103	
Dia da semana:	6	Slot:	11	Disciplina:	BIOLOGIA	Turma:	102	
Dia da semana:	6	Slot:	12	Disciplina:	BIOLOGIA	Turma:	302	

| Carga horária: 9 |

| Prof: P6 |

Dia da semana:	2	Slot:	8	Disciplina:	HISTÓRIA	Turma:	203	
Dia da semana:	2	Slot:	9	Disciplina:	HISTÓRIA	Turma:	103	
Dia da semana:	2	Slot:	10	Disciplina:	HISTÓRIA	Turma:	202	
Dia da semana:	2	Slot:	11	Disciplina:	HISTÓRIA	Turma:	102	
Dia da semana:	2	Slot:	12	Disciplina:	HISTÓRIA	Turma:	302	

Dia da semana:	3	Slot:	7	Disciplina:	HISTÓRIA	Turma:	103	
----------------	---	-------	---	-------------	----------	--------	-----	--

Dia da semana:	3	Slot:	8	Disciplina:	HISTÓRIA	Turma:	102	
Dia da semana:	3	Slot:	9	Disciplina:	HISTÓRIA	Turma:	203	
Dia da semana:	3	Slot:	10	Disciplina:	HISTÓRIA	Turma:	202	

Dia da semana:	5	Slot:	7	Disciplina:	HISTÓRIA	Turma:	103	
Dia da semana:	5	Slot:	8	Disciplina:	HISTÓRIA	Turma:	102	
Dia da semana:	5	Slot:	9	Disciplina:	HISTÓRIA	Turma:	302	
Dia da semana:	5	Slot:	10	Disciplina:	HISTÓRIA	Turma:	202	
Dia da semana:	5	Slot:	11	Disciplina:	HISTÓRIA	Turma:	203	

| Carga horária: 14 |

| Prof: P7 |

Dia da semana:	3	Slot:	8	Disciplina:	FÍSICA	Turma:	103	
Dia da semana:	3	Slot:	9	Disciplina:	FÍSICA	Turma:	202	
Dia da semana:	3	Slot:	10	Disciplina:	FÍSICA	Turma:	203	
Dia da semana:	3	Slot:	11	Disciplina:	FÍSICA	Turma:	102	
Dia da semana:	3	Slot:	12	Disciplina:	FÍSICA	Turma:	302	

Dia da semana:	5	Slot:	7	Disciplina:	FÍSICA	Turma:	202	
Dia da semana:	5	Slot:	8	Disciplina:	FÍSICA	Turma:	302	
Dia da semana:	5	Slot:	9	Disciplina:	FÍSICA	Turma:	102	
Dia da semana:	5	Slot:	10	Disciplina:	FÍSICA	Turma:	203	
Dia da semana:	5	Slot:	11	Disciplina:	FÍSICA	Turma:	103	

| Carga horária: 10 |

| Prof: P8 |

Dia da semana:	4	Slot:	7	Disciplina:	GEOGRAFIA	Turma:	302	
Dia da semana:	4	Slot:	8	Disciplina:	GEOGRAFIA	Turma:	102	
Dia da semana:	4	Slot:	9	Disciplina:	GEOGRAFIA	Turma:	103	
Dia da semana:	4	Slot:	10	Disciplina:	GEOGRAFIA	Turma:	203	
Dia da semana:	4	Slot:	11	Disciplina:	GEOGRAFIA	Turma:	202	

Dia da semana:	5	Slot:	7	Disciplina:	GEOGRAFIA	Turma:	102	
----------------	---	-------	---	-------------	-----------	--------	-----	--

Dia da semana:	5	Slot:	8	Disciplina:	GEOGRAFIA	Turma:	202	
Dia da semana:	5	Slot:	9	Disciplina:	GEOGRAFIA	Turma:	203	
Dia da semana:	5	Slot:	10	Disciplina:	GEOGRAFIA	Turma:	103	
Dia da semana:	5	Slot:	11	Disciplina:	GEOGRAFIA	Turma:	302	

| Carga horária: 10 |

| Prof: P9 |

Dia da semana:	3	Slot:	8	Disciplina:	FILOSOFIA	Turma:	302	
Dia da semana:	3	Slot:	9	Disciplina:	FILOSOFIA	Turma:	102	
Dia da semana:	3	Slot:	10	Disciplina:	FILOSOFIA	Turma:	103	
Dia da semana:	3	Slot:	11	Disciplina:	FILOSOFIA	Turma:	202	
Dia da semana:	3	Slot:	12	Disciplina:	FILOSOFIA	Turma:	203	

Dia da semana:	4	Slot:	8	Disciplina:	FILOSOFIA	Turma:	302	
Dia da semana:	4	Slot:	9	Disciplina:	FILOSOFIA	Turma:	203	
Dia da semana:	4	Slot:	10	Disciplina:	FILOSOFIA	Turma:	202	

| Carga horária: 8 |

| Prof: P10 |

Dia da semana:	3	Slot:	8	Disciplina:	QUÍMICA	Turma:	203	
Dia da semana:	3	Slot:	9	Disciplina:	QUÍMICA	Turma:	302	
Dia da semana:	3	Slot:	10	Disciplina:	QUÍMICA	Turma:	102	
Dia da semana:	3	Slot:	11	Disciplina:	QUÍMICA	Turma:	103	
Dia da semana:	3	Slot:	12	Disciplina:	QUÍMICA	Turma:	202	

Dia da semana:	5	Slot:	8	Disciplina:	QUÍMICA	Turma:	203	
Dia da semana:	5	Slot:	9	Disciplina:	QUÍMICA	Turma:	103	
Dia da semana:	5	Slot:	10	Disciplina:	QUÍMICA	Turma:	102	
Dia da semana:	5	Slot:	11	Disciplina:	QUÍMICA	Turma:	202	
Dia da semana:	5	Slot:	12	Disciplina:	QUÍMICA	Turma:	302	

| Carga horária: 10 |

Prof: P11

Dia da semana: 4	Slot: 10	Disciplina: INGLÊS	Turma: 102
Dia da semana: 4	Slot: 11	Disciplina: INGLÊS	Turma: 302
Dia da semana: 4	Slot: 12	Disciplina: INGLÊS	Turma: 103

Dia da semana: 5	Slot: 7	Disciplina: INGLÊS	Turma: 302
Dia da semana: 5	Slot: 8	Disciplina: INGLÊS	Turma: 103
Dia da semana: 5	Slot: 12	Disciplina: INGLÊS	Turma: 102

Carga horária: 6

Prof: P12

Dia da semana: 2	Slot: 7	Disciplina: EDUCAÇÃO FÍSICA	Turma: 102
Dia da semana: 2	Slot: 11	Disciplina: EDUCAÇÃO FÍSICA	Turma: 302
Dia da semana: 2	Slot: 12	Disciplina: EDUCAÇÃO FÍSICA	Turma: 203

Dia da semana: 6	Slot: 9	Disciplina: EDUCAÇÃO FÍSICA	Turma: 102
Dia da semana: 6	Slot: 10	Disciplina: EDUCAÇÃO FÍSICA	Turma: 203
Dia da semana: 6	Slot: 11	Disciplina: EDUCAÇÃO FÍSICA	Turma: 302

Carga horária: 6

Prof: P13

|Dia da semana: 4 | Slot: 7 | Disciplina: INGLÊS | Turma: 202 |
|Dia da semana: 4 | Slot: 12 | Disciplina: INGLÊS | Turma: 203 |

|Dia da semana: 5 | Slot: 7 | Disciplina: INGLÊS | Turma: 203 |
|Dia da semana: 5 | Slot: 12 | Disciplina: INGLÊS | Turma: 202 |

|Dia da semana: 6 | Slot: 7 | Disciplina: INGLÊS | Turma: 203 |

|Dia da semana: 6 | Slot: 8 | Disciplina: INGLÊS | Turma: 202 |
| Carga horária: 6 |

| Prof: P14 |

Dia da semana: 2	Slot: 7	Disciplina: MATEMÁTICA	Turma: 203
Dia da semana: 2	Slot: 8	Disciplina: MATEMÁTICA	Turma: 102
Dia da semana: 2	Slot: 9	Disciplina: MATEMÁTICA	Turma: 202
Dia da semana: 2	Slot: 10	Disciplina: MATEMÁTICA	Turma: 103

Dia da semana: 4	Slot: 7	Disciplina: MATEMÁTICA	Turma: 103
Dia da semana: 4	Slot: 8	Disciplina: MATEMÁTICA	Turma: 203
Dia da semana: 4	Slot: 9	Disciplina: MATEMÁTICA	Turma: 102
Dia da semana: 4	Slot: 12	Disciplina: MATEMÁTICA	Turma: 202

Dia da semana: 6	Slot: 9	Disciplina: MATEMÁTICA	Turma: 203
Dia da semana: 6	Slot: 10	Disciplina: MATEMÁTICA	Turma: 102
Dia da semana: 6	Slot: 11	Disciplina: MATEMÁTICA	Turma: 202
Dia da semana: 6	Slot: 12	Disciplina: MATEMÁTICA	Turma: 103

| Carga horária: 12 |

| Prof: P15 |

|Dia da semana: 5 | Slot: 9 | Disciplina: BIOLOGIA | Turma: 202 |
|Dia da semana: 5 | Slot: 12 | Disciplina: BIOLOGIA | Turma: 203 |

|Dia da semana: 6 | Slot: 7 | Disciplina: BIOLOGIA | Turma: 202 |
|Dia da semana: 6 | Slot: 8 | Disciplina: BIOLOGIA | Turma: 203 |

| Carga horária: 4 |

| Prof: P16 |

```

-----
---
|Dia da semana:  2 | Slot:  11 | Disciplina:  EDUCAÇÃO FÍSICA | Turma:  202 |
|Dia da semana:  2 | Slot:  12 | Disciplina:  EDUCAÇÃO FÍSICA | Turma:  103 |
-----
---
|Dia da semana:  3 | Slot:   7 | Disciplina:  EDUCAÇÃO FÍSICA | Turma:  202 |
-----
---
|Dia da semana:  6 | Slot:   9 | Disciplina:  EDUCAÇÃO FÍSICA | Turma:  202 |
|Dia da semana:  6 | Slot:  10 | Disciplina:  EDUCAÇÃO FÍSICA | Turma:  103 |

| Carga horária:  5 |

```

```

-----
---
| Prof:  P17 |
-----
---
|Dia da semana:  2 | Slot:  12 | Disciplina:          LITERATURA | Turma:  202 |
-----
---
|Dia da semana:  3 | Slot:   8 | Disciplina:          LITERATURA | Turma:  202 |
-----
---
|Dia da semana:  4 | Slot:   9 | Disciplina:          LITERATURA | Turma:  202 |

| Carga horária:  3 |

```

```

-----
---
| Prof:  P18 |
-----
---
|Dia da semana:  2 | Slot:  11 | Disciplina:          ARTE | Turma:  103 |
-----
---
|Dia da semana:  3 | Slot:   9 | Disciplina:          ARTE | Turma:  103 |
-----
---
|Dia da semana:  6 | Slot:   9 | Disciplina:          ARTE | Turma:  103 |

| Carga horária:  3 |

```



```

[38]: #printar grade horaria turma
for z in range(len(turmas)):
    atual=0
    anterior=0
    t=0
    turma = objs_turmas[z]

    ↳
    ↳ listaIndiceDisciplina,listaIndiceProfessor,listaIndiceHorario=gerar_lista_indices_turma(turma)
    t=turma.get_index()
    contador=0
    print(" ")

    ↳
    ↳ print("-----")
    print("| Turma: ",round(turma.get_codigo()),"|")
    for h in listaIndiceHorario:
        horario=objs_horarios[h]
        atual=horario.get_dia_semana()
        if(atual!=anterior):

            ↳
            ↳ print("-----")
            for d in listaIndiceDisciplina:
                disciplina=objs[d]
                for p in listaIndiceProfessor:
                    professor = objs_professores[p]
                    if (round(x[t,d,p,h].X)==1):
                        contador=contador+1
                        anterior=horario.get_dia_semana()
                        print("|%15s %1d | %6s %2d | %12s %16s | %12s %3s |" %("Dia",
↳ da semana: ",horario.get_dia_semana(),"Slot: ",horario.
↳ get_codigo(),"Disciplina: ",disciplina.nome(),"Professor: ",professor.
↳ get_nome()))

                        #print("/ Dia da Semana: ",horario.
↳ get_dia_semana(),"\N{DEGREE SIGN}","/ Slot: ",horario.get_codigo(),"|
↳ Disciplina: ",disciplina.nome(),"/ Professor: ",professor.get_nome(),"/ Aula:
↳ ",round(x[t,d,p,h].X),"/")

            ↳
            ↳ print("-----")
            print(" ")
            print("|Carga horária: ",contador,"|")
            print(" ")

```

```

-----
---
| Turma: 302 |
-----
---
```

Dia da semana:	2	Slot:	7	Disciplina:	ARTE	Professor:
P4						
Dia da semana:	2	Slot:	8	Disciplina:	LITERATURA	Professor:
P2						
Dia da semana:	2	Slot:	9	Disciplina:	SOCIOLOGIA	Professor:
P1						
Dia da semana:	2	Slot:	10	Disciplina:	MATEMÁTICA	Professor:
P3						
Dia da semana:	2	Slot:	11	Disciplina:	EDUCAÇÃO FÍSICA	Professor:
P12						
Dia da semana:	2	Slot:	12	Disciplina:	HISTÓRIA	Professor:
P6						

Dia da semana:	3	Slot:	7	Disciplina:	MATEMÁTICA	Professor:
P3						
Dia da semana:	3	Slot:	8	Disciplina:	FILOSOFIA	Professor:
P9						
Dia da semana:	3	Slot:	9	Disciplina:	QUÍMICA	Professor:
P10						
Dia da semana:	3	Slot:	10	Disciplina:	LITERATURA	Professor:
P2						
Dia da semana:	3	Slot:	11	Disciplina:	ARTE	Professor:
P4						
Dia da semana:	3	Slot:	12	Disciplina:	FÍSICA	Professor:
P7						

Dia da semana:	4	Slot:	7	Disciplina:	GEOGRAFIA	Professor:
P8						
Dia da semana:	4	Slot:	8	Disciplina:	FILOSOFIA	Professor:
P9						
Dia da semana:	4	Slot:	9	Disciplina:	BIOLOGIA	Professor:
P5						
Dia da semana:	4	Slot:	10	Disciplina:	MATEMÁTICA	Professor:
P3						
Dia da semana:	4	Slot:	11	Disciplina:	INGLÊS	Professor:
P11						
Dia da semana:	4	Slot:	12	Disciplina:	LITERATURA	Professor:
P2						

Dia da semana:	5	Slot:	7	Disciplina:	INGLÊS	Professor:
P11						
Dia da semana:	5	Slot:	8	Disciplina:	FÍSICA	Professor:
P7						
Dia da semana:	5	Slot:	9	Disciplina:	HISTÓRIA	Professor:
P6						

Dia da semana:	5	Slot:	10	Disciplina:	BIOLOGIA	Professor:
P5						
Dia da semana:	5	Slot:	11	Disciplina:	GEOGRAFIA	Professor:
P8						
Dia da semana:	5	Slot:	12	Disciplina:	QUÍMICA	Professor:
P10						

Dia da semana:	6	Slot:	7	Disciplina:	LITERATURA	Professor:
P2						
Dia da semana:	6	Slot:	8	Disciplina:	ARTE	Professor:
P4						
Dia da semana:	6	Slot:	9	Disciplina:	SOCIOLOGIA	Professor:
P1						
Dia da semana:	6	Slot:	10	Disciplina:	MATEMÁTICA	Professor:
P3						
Dia da semana:	6	Slot:	11	Disciplina:	EDUCAÇÃO FÍSICA	Professor:
P12						
Dia da semana:	6	Slot:	12	Disciplina:	BIOLOGIA	Professor:
P5						

|Carga horária: 30 |

| Turma: 203 |

Dia da semana:	2	Slot:	7	Disciplina:	MATEMÁTICA	Professor:
P14						
Dia da semana:	2	Slot:	8	Disciplina:	HISTÓRIA	Professor:
P6						
Dia da semana:	2	Slot:	9	Disciplina:	LITERATURA	Professor:
P2						
Dia da semana:	2	Slot:	10	Disciplina:	ARTE	Professor:
P4						
Dia da semana:	2	Slot:	11	Disciplina:	SOCIOLOGIA	Professor:
P1						
Dia da semana:	2	Slot:	12	Disciplina:	EDUCAÇÃO FÍSICA	Professor:
P12						

Dia da semana:	3	Slot:	7	Disciplina:	ARTE	Professor:
P4						
Dia da semana:	3	Slot:	8	Disciplina:	QUÍMICA	Professor:

P10 |
 |Dia da semana: 3 | Slot: 9 | Disciplina: HISTÓRIA | Professor:
 P6 |
 |Dia da semana: 3 | Slot: 10 | Disciplina: FÍSICA | Professor:
 P7 |
 |Dia da semana: 3 | Slot: 11 | Disciplina: LITERATURA | Professor:
 P2 |
 |Dia da semana: 3 | Slot: 12 | Disciplina: FILOSOFIA | Professor:
 P9 |

|Dia da semana: 4 | Slot: 7 | Disciplina: SOCIOLOGIA | Professor:
 P1 |
 |Dia da semana: 4 | Slot: 8 | Disciplina: MATEMÁTICA | Professor:
 P14 |
 |Dia da semana: 4 | Slot: 9 | Disciplina: FILOSOFIA | Professor:
 P9 |
 |Dia da semana: 4 | Slot: 10 | Disciplina: GEOGRAFIA | Professor:
 P8 |
 |Dia da semana: 4 | Slot: 11 | Disciplina: LITERATURA | Professor:
 P2 |
 |Dia da semana: 4 | Slot: 12 | Disciplina: INGLÊS | Professor:
 P13 |

|Dia da semana: 5 | Slot: 7 | Disciplina: INGLÊS | Professor:
 P13 |
 |Dia da semana: 5 | Slot: 8 | Disciplina: QUÍMICA | Professor:
 P10 |
 |Dia da semana: 5 | Slot: 9 | Disciplina: GEOGRAFIA | Professor:
 P8 |
 |Dia da semana: 5 | Slot: 10 | Disciplina: FÍSICA | Professor:
 P7 |
 |Dia da semana: 5 | Slot: 11 | Disciplina: HISTÓRIA | Professor:
 P6 |
 |Dia da semana: 5 | Slot: 12 | Disciplina: BIOLOGIA | Professor:
 P15 |

|Dia da semana: 6 | Slot: 7 | Disciplina: INGLÊS | Professor:
 P13 |
 |Dia da semana: 6 | Slot: 8 | Disciplina: BIOLOGIA | Professor:
 P15 |
 |Dia da semana: 6 | Slot: 9 | Disciplina: MATEMÁTICA | Professor:
 P14 |
 |Dia da semana: 6 | Slot: 10 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
 P12 |
 |Dia da semana: 6 | Slot: 11 | Disciplina: LITERATURA | Professor:

P2 |
|Dia da semana: 6 | Slot: 12 | Disciplina: SOCIOLOGIA | Professor:
P1

|Carga horária: 30 |

Turma: 202

|Dia da semana: 2 | Slot: 7 | Disciplina: SOCIOLOGIA | Professor:
P1 |
|Dia da semana: 2 | Slot: 8 | Disciplina: ARTE | Professor:
P4 |
|Dia da semana: 2 | Slot: 9 | Disciplina: MATEMÁTICA | Professor:
P14 |
|Dia da semana: 2 | Slot: 10 | Disciplina: HISTÓRIA | Professor:
P6 |
|Dia da semana: 2 | Slot: 11 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
P16 |
|Dia da semana: 2 | Slot: 12 | Disciplina: LITERATURA | Professor:
P17

|Dia da semana: 3 | Slot: 7 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
P16 |
|Dia da semana: 3 | Slot: 8 | Disciplina: LITERATURA | Professor:
P17 |
|Dia da semana: 3 | Slot: 9 | Disciplina: FÍSICA | Professor:
P7 |
|Dia da semana: 3 | Slot: 10 | Disciplina: HISTÓRIA | Professor:
P6 |
|Dia da semana: 3 | Slot: 11 | Disciplina: FILOSOFIA | Professor:
P9 |
|Dia da semana: 3 | Slot: 12 | Disciplina: QUÍMICA | Professor:
P10

|Dia da semana: 4 | Slot: 7 | Disciplina: INGLÊS | Professor:
P13 |
|Dia da semana: 4 | Slot: 8 | Disciplina: SOCIOLOGIA | Professor:
P1 |
|Dia da semana: 4 | Slot: 9 | Disciplina: LITERATURA | Professor:
P17 |

|Dia da semana: 4 | Slot: 10 | Disciplina: FILOSOFIA | Professor:
P9 |
|Dia da semana: 4 | Slot: 11 | Disciplina: GEOGRAFIA | Professor:
P8 |
|Dia da semana: 4 | Slot: 12 | Disciplina: MATEMÁTICA | Professor:
P14 |

|Dia da semana: 5 | Slot: 7 | Disciplina: FÍSICA | Professor:
P7 |
|Dia da semana: 5 | Slot: 8 | Disciplina: GEOGRAFIA | Professor:
P8 |
|Dia da semana: 5 | Slot: 9 | Disciplina: BIOLOGIA | Professor:
P15 |
|Dia da semana: 5 | Slot: 10 | Disciplina: HISTÓRIA | Professor:
P6 |
|Dia da semana: 5 | Slot: 11 | Disciplina: QUÍMICA | Professor:
P10 |
|Dia da semana: 5 | Slot: 12 | Disciplina: INGLÊS | Professor:
P13 |

|Dia da semana: 6 | Slot: 7 | Disciplina: BIOLOGIA | Professor:
P15 |
|Dia da semana: 6 | Slot: 8 | Disciplina: INGLÊS | Professor:
P13 |
|Dia da semana: 6 | Slot: 9 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
P16 |
|Dia da semana: 6 | Slot: 10 | Disciplina: SOCIOLOGIA | Professor:
P1 |
|Dia da semana: 6 | Slot: 11 | Disciplina: MATEMÁTICA | Professor:
P14 |
|Dia da semana: 6 | Slot: 12 | Disciplina: ARTE | Professor:
P4 |

|Carga horária: 30 |

| Turma: 103 |

|Dia da semana: 2 | Slot: 7 | Disciplina: LITERATURA | Professor:
P2 |
|Dia da semana: 2 | Slot: 8 | Disciplina: SOCIOLOGIA | Professor:

P1 |
 |Dia da semana: 2 | Slot: 9 | Disciplina: HISTÓRIA | Professor:
 P6 |
 |Dia da semana: 2 | Slot: 10 | Disciplina: MATEMÁTICA | Professor:
 P14 |
 |Dia da semana: 2 | Slot: 11 | Disciplina: ARTE | Professor:
 P18 |
 |Dia da semana: 2 | Slot: 12 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
 P16 |

|Dia da semana: 3 | Slot: 7 | Disciplina: HISTÓRIA | Professor:
 P6 |
 |Dia da semana: 3 | Slot: 8 | Disciplina: FÍSICA | Professor:
 P7 |
 |Dia da semana: 3 | Slot: 9 | Disciplina: ARTE | Professor:
 P18 |
 |Dia da semana: 3 | Slot: 10 | Disciplina: FILOSOFIA | Professor:
 P9 |
 |Dia da semana: 3 | Slot: 11 | Disciplina: QUÍMICA | Professor:
 P10 |
 |Dia da semana: 3 | Slot: 12 | Disciplina: LITERATURA | Professor:
 P2 |

|Dia da semana: 4 | Slot: 7 | Disciplina: MATEMÁTICA | Professor:
 P14 |
 |Dia da semana: 4 | Slot: 8 | Disciplina: LITERATURA | Professor:
 P2 |
 |Dia da semana: 4 | Slot: 9 | Disciplina: GEOGRAFIA | Professor:
 P8 |
 |Dia da semana: 4 | Slot: 10 | Disciplina: BIOLOGIA | Professor:
 P5 |
 |Dia da semana: 4 | Slot: 11 | Disciplina: SOCIOLOGIA | Professor:
 P1 |
 |Dia da semana: 4 | Slot: 12 | Disciplina: INGLÊS | Professor:
 P11 |

|Dia da semana: 5 | Slot: 7 | Disciplina: HISTÓRIA | Professor:
 P6 |
 |Dia da semana: 5 | Slot: 8 | Disciplina: INGLÊS | Professor:
 P11 |
 |Dia da semana: 5 | Slot: 9 | Disciplina: QUÍMICA | Professor:
 P10 |
 |Dia da semana: 5 | Slot: 10 | Disciplina: GEOGRAFIA | Professor:
 P8 |
 |Dia da semana: 5 | Slot: 11 | Disciplina: FÍSICA | Professor:

P7 |
|Dia da semana: 5 | Slot: 12 | Disciplina: BIOLOGIA | Professor:
P5

|Dia da semana: 6 | Slot: 7 | Disciplina: BIOLOGIA | Professor:
P5 |
|Dia da semana: 6 | Slot: 8 | Disciplina: LITERATURA | Professor:
P2 |
|Dia da semana: 6 | Slot: 9 | Disciplina: ARTE | Professor:
P18 |
|Dia da semana: 6 | Slot: 10 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
P16 |
|Dia da semana: 6 | Slot: 11 | Disciplina: SOCIOLOGIA | Professor:
P1 |
|Dia da semana: 6 | Slot: 12 | Disciplina: MATEMÁTICA | Professor:
P14

|Carga horária: 30 |

Turma: 102

|Dia da semana: 2 | Slot: 7 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
P12 |
|Dia da semana: 2 | Slot: 8 | Disciplina: MATEMÁTICA | Professor:
P14 |
|Dia da semana: 2 | Slot: 9 | Disciplina: ARTE | Professor:
P4 |
|Dia da semana: 2 | Slot: 10 | Disciplina: SOCIOLOGIA | Professor:
P1 |
|Dia da semana: 2 | Slot: 11 | Disciplina: HISTÓRIA | Professor:
P6 |
|Dia da semana: 2 | Slot: 12 | Disciplina: LITERATURA | Professor:
P2

|Dia da semana: 3 | Slot: 7 | Disciplina: LITERATURA | Professor:
P2 |
|Dia da semana: 3 | Slot: 8 | Disciplina: HISTÓRIA | Professor:
P6 |
|Dia da semana: 3 | Slot: 9 | Disciplina: FILOSOFIA | Professor:
P9 |

|Dia da semana: 3 | Slot: 10 | Disciplina: QUÍMICA | Professor:
P10 |

|Dia da semana: 3 | Slot: 11 | Disciplina: FÍSICA | Professor:
P7 |

|Dia da semana: 3 | Slot: 12 | Disciplina: ARTE | Professor:
P4 |

|Dia da semana: 4 | Slot: 7 | Disciplina: LITERATURA | Professor:
P2 |

|Dia da semana: 4 | Slot: 8 | Disciplina: GEOGRAFIA | Professor:
P8 |

|Dia da semana: 4 | Slot: 9 | Disciplina: MATEMÁTICA | Professor:
P14 |

|Dia da semana: 4 | Slot: 10 | Disciplina: INGLÊS | Professor:
P11 |

|Dia da semana: 4 | Slot: 11 | Disciplina: BIOLOGIA | Professor:
P5 |

|Dia da semana: 4 | Slot: 12 | Disciplina: SOCIOLOGIA | Professor:
P1 |

|Dia da semana: 5 | Slot: 7 | Disciplina: GEOGRAFIA | Professor:
P8 |

|Dia da semana: 5 | Slot: 8 | Disciplina: HISTÓRIA | Professor:
P6 |

|Dia da semana: 5 | Slot: 9 | Disciplina: FÍSICA | Professor:
P7 |

|Dia da semana: 5 | Slot: 10 | Disciplina: QUÍMICA | Professor:
P10 |

|Dia da semana: 5 | Slot: 11 | Disciplina: BIOLOGIA | Professor:
P5 |

|Dia da semana: 5 | Slot: 12 | Disciplina: INGLÊS | Professor:
P11 |

|Dia da semana: 6 | Slot: 7 | Disciplina: ARTE | Professor:
P4 |

|Dia da semana: 6 | Slot: 8 | Disciplina: SOCIOLOGIA | Professor:
P1 |

|Dia da semana: 6 | Slot: 9 | Disciplina: EDUCAÇÃO FÍSICA | Professor:
P12 |

|Dia da semana: 6 | Slot: 10 | Disciplina: MATEMÁTICA | Professor:
P14 |

|Dia da semana: 6 | Slot: 11 | Disciplina: BIOLOGIA | Professor:
P5 |

|Dia da semana: 6 | Slot: 12 | Disciplina: LITERATURA | Professor:
P2 |

|Carga horária: 30 |

```
[39]: print("-----Segunda-----")
      for h in range (6,12):
          horario=objs_horarios[h]
          print("| SLOT:",horario.get_codigo(),"| Inicio: ",horario.
            ↳get_inicio(),"|")
      print("-----Terca-----")
      for h in range (18,24):
          horario=objs_horarios[h]
          print("| SLOT:",horario.get_codigo(),"| Inicio: ",horario.
            ↳get_inicio(),"|")
      print("-----Quarta-----")
      for h in range (30,36):
          horario=objs_horarios[h]
          print("| SLOT:",horario.get_codigo(),"| Inicio: ",horario.
            ↳get_inicio(),"|")
      print("-----Quinta-----")
      for h in range (42,48):
          horario=objs_horarios[h]
          print("| SLOT:",horario.get_codigo(),"| Inicio: ",horario.
            ↳get_inicio(),"|")
      print("-----Sexta-----")
      for h in range (54,60):
          horario=objs_horarios[h]
          print("| SLOT:",horario.get_codigo(),"| Inicio: ",horario.
            ↳get_inicio(),"|")
```

```
-----Segunda-----
| SLOT: 7 | Inicio: 13:30 |
| SLOT: 8 | Inicio: 14:15 |
| SLOT: 9 | Inicio: 15:00 |
| SLOT: 10 | Inicio: 15:35 |
| SLOT: 11 | Inicio: 16:15 |
| SLOT: 12 | Inicio: 16:50 |
-----Terca-----
| SLOT: 7 | Inicio: 13:30 |
| SLOT: 8 | Inicio: 14:15 |
| SLOT: 9 | Inicio: 15:00 |
| SLOT: 10 | Inicio: 15:35 |
| SLOT: 11 | Inicio: 16:15 |
| SLOT: 12 | Inicio: 16:50 |
-----Quarta-----
| SLOT: 7 | Inicio: 13:30 |
```

```

| SLOT: 8 | Inicio: 14:15 |
| SLOT: 9 | Inicio: 15:00 |
| SLOT: 10 | Inicio: 15:35 |
| SLOT: 11 | Inicio: 16:15 |
| SLOT: 12 | Inicio: 16:50 |
-----Quinta-----
| SLOT: 7 | Inicio: 13:30 |
| SLOT: 8 | Inicio: 14:15 |
| SLOT: 9 | Inicio: 15:00 |
| SLOT: 10 | Inicio: 15:35 |
| SLOT: 11 | Inicio: 16:15 |
| SLOT: 12 | Inicio: 16:50 |
-----Sexta-----
| SLOT: 7 | Inicio: 13:30 |
| SLOT: 8 | Inicio: 14:15 |
| SLOT: 9 | Inicio: 15:00 |
| SLOT: 10 | Inicio: 15:35 |
| SLOT: 11 | Inicio: 16:15 |
| SLOT: 12 | Inicio: 16:50 |

```

```

[40]: print("-----Análise de resultados-----")
print("")
total=0
janela_semanal_total=0
print("|-----Siglas-----|")
print("|h1 corresponde a um slot de horario |")
print("|h2 corresponde ao slot sequente de h1 |")
print("|h3 corresponde ao slot sequente de h2 |")
print("|j corresponde se existe janela |")
print("|p corresponde a penalidade atribuido a sequencia de h1,h2,h3 |")
print("|-----|")
print("|penalidade aplicada nesta execucao,se foi adicionada na funcao_
↪objetivo|")
print("")
for z in range(len(professores)):
    p=0
    professor = objs_professores[z]
    ↪
    ↪listaIndiceDisciplina,listaIndiceTurma,listaIndiceHorario=gerar_lista_indices_professor(pro
    p=professor.get_index()
    janela_segunda=0
    janela_terca=0
    janela_quarta=0
    janela_quinta=0
    janela_sexta=0
    janela_semanal=0
    janela_semanal_real=0

```

```

print("-----")
print("Professor:", professor.get_nome())

print("-----Segunda-----")
print("h1", "h2", "h3", "j", " p")
for h in range (6,10):
    a=0
    b=0
    c=0
    j=0
    pen=0
    for t in listaIndiceTurma:
        for d in listaIndiceDisciplina:
            a=a+round(x[t,d,p,h].X)
            b=b+round(x[t,d,p,h+1].X)
            c=c+round(x[t,d,p,h+2].X)
        janela_segunda=janela_segunda+(a*(1-b)*c)
        j=a*(1-b)*c
        pen= a*(1-b)*c
    print(a," ",b," ",c," ",j," ",pen)

slot1=0
slot2=0
slot3=0
slot4=0
slot5=0
slot6=0
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
for t in listaIndiceTurma:
    for d in listaIndiceDisciplina:
        slot1=slot1+round(x[t,d,p,6].X)
        slot2=slot2+round(x[t,d,p,7].X)
        slot3=slot3+round(x[t,d,p,8].X)
        slot4=slot4+round(x[t,d,p,9].X)
        slot5=slot5+round(x[t,d,p,10].X)
        slot6=slot6+round(x[t,d,p,11].X)
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
janelas_existentes_segunda=0
contador_janelas_segunda=0
pular=0
for b in range(0,5):
    if(pular>0):
        pular=pular-1
    if(pular==0):
        if(lista[b]==1):
            contador_janelas_segunda=0

```

```

        for e in range(b+1,6):
            if(e==5 and lista[e]==0):
                contador_janelas_segunda=0
                break
            if(lista[e]==0):
                contador_janelas_segunda=contador_janelas_segunda+1
            if(e!=b and lista[e]==1):
                break
        if(contador_janelas_segunda>0):
            pular=contador_janelas_segunda+1
        □
    →janelas_existentes_segunda=janelas_existentes_segunda+contador_janelas_segunda
    print(" ")
    print("janelas analisando h1,h2,h3 na segunda =",janela_segunda)
    print(" ")
    print("janelas existentes na segunda =",janelas_existentes_segunda)

    print("-----Terca-----")
    print("h1","h2","h3","j"," p")
    for h in range (18,22):
        a=0
        b=0
        c=0
        j=0
        pen=0
        for t in listaIndiceTurma:
            for d in listaIndiceDisciplina:
                a=a+round(x[t,d,p,h].X)
                b=b+round(x[t,d,p,h+1].X)
                c=c+round(x[t,d,p,h+2].X)
            janela_terca=janela_terca+(a*(1-b)*c)
            j=a*(1-b)*c
            pen=a*(1-b)*c
        print(a,"",b,"",c,"",j," ",pen)

    slot1=0
    slot2=0
    slot3=0
    slot4=0
    slot5=0
    slot6=0
    lista=[slot1,slot2,slot3,slot4,slot5,slot6]
    for t in listaIndiceTurma:
        for d in listaIndiceDisciplina:
            slot1=slot1+round(x[t,d,p,18].X)

```

```

        slot2=slot2+round(x[t,d,p,19].X)
        slot3=slot3+round(x[t,d,p,20].X)
        slot4=slot4+round(x[t,d,p,21].X)
        slot5=slot5+round(x[t,d,p,22].X)
        slot6=slot6+round(x[t,d,p,23].X)
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
janelas_existentes_terca=0
contador_janelas_terca=0
pular=0
for b in range(0,5):
    if(pular>0):
        pular=pular-1
    if(pular==0):
        if(lista[b]==1):
            contador_janelas_terca=0
            for e in range(b+1,6):
                if(e==5 and lista[e]==0):
                    contador_janelas_terca=0
                    break
                if(lista[e]==0):
                    contador_janelas_terca=contador_janelas_terca+1
                if(e!=b and lista[e]==1):
                    break
            if(contador_janelas_terca>0):
                pular=contador_janelas_terca+1
        □
→janelas_existentes_terca=janelas_existentes_terca+contador_janelas_terca
print(" ")
print("janelas analisando h1,h2,h3 na terca =",janela_terca)
print(" ")
print("janelas existentes na terca=",janelas_existentes_terca)

print("-----Quarta-----")
print("h1","h2","h3","j"," p")
for h in range (30,34):
    a=0
    b=0
    c=0
    j=0
    pen=0
    for t in listaIndiceTurma:
        for d in listaIndiceDisciplina:
            a=a+round(x[t,d,p,h].X)
            b=b+round(x[t,d,p,h+1].X)
            c=c+round(x[t,d,p,h+2].X)
    janela_quarta=janela_quarta+(a*(1-b)*c)
    j=a*(1-b)*c

```

```

pen=a*(1-b)*c
print(a," ",b," ",c," ",j," ",pen)

slot1=0
slot2=0
slot3=0
slot4=0
slot5=0
slot6=0
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
for t in listaIndiceTurma:
    for d in listaIndiceDisciplina:
        slot1=slot1+round(x[t,d,p,30].X)
        slot2=slot2+round(x[t,d,p,31].X)
        slot3=slot3+round(x[t,d,p,32].X)
        slot4=slot4+round(x[t,d,p,33].X)
        slot5=slot5+round(x[t,d,p,34].X)
        slot6=slot6+round(x[t,d,p,35].X)
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
janelas_existentes_quarta=0
contador_janelas_quarta=0
pular=0
for b in range(0,5):
    if(pular>0):
        pular=pular-1
    if(pular==0):
        if(lista[b]==1):
            contador_janelas_quarta=0
            for e in range(b+1,6):
                if(e==5 and lista[e]==0):
                    contador_janelas_quarta=0
                    break
                if(lista[e]==0):
                    contador_janelas_quarta=contador_janelas_quarta+1
                if(e!=b and lista[e]==1):
                    break
            if(contador_janelas_quarta>0):
                pular=contador_janelas_quarta+1
            □
        ↪ janelas_existentes_quarta=janelas_existentes_quarta+contador_janelas_quarta
print(" ")
print("janelas analisando h1,h2,h3 na quarta =",janela_quarta)
print(" ")
print("janelas existentes na quarta=",janelas_existentes_quarta)

```

```

print("-----Quinta-----")
print("h1","h2","h3","j"," p")
for h in range (42,46):
    a=0
    b=0
    c=0
    j=0
    pen=0
    for t in listaIndiceTurma:
        for d in listaIndiceDisciplina:
            a=a+round(x[t,d,p,h].X)
            b=b+round(x[t,d,p,h+1].X)
            c=c+round(x[t,d,p,h+2].X)
        janela_quinta=janela_quinta+(a*(1-b)*c)
        j=a*(1-b)*c
        pen=a*(1-b)*c
        print(a,"",b,"",c,"",j," ",pen)

slot1=0
slot2=0
slot3=0
slot4=0
slot5=0
slot6=0
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
for t in listaIndiceTurma:
    for d in listaIndiceDisciplina:
        slot1=slot1+round(x[t,d,p,42].X)
        slot2=slot2+round(x[t,d,p,43].X)
        slot3=slot3+round(x[t,d,p,44].X)
        slot4=slot4+round(x[t,d,p,45].X)
        slot5=slot5+round(x[t,d,p,46].X)
        slot6=slot6+round(x[t,d,p,47].X)
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
janelas_existentes_quinta=0
contador_janelas_quinta=0
pular=0
for b in range(0,5):
    if(pular>0):
        pular=pular-1
    if(pular==0):
        if(lista[b]==1):
            contador_janelas_quinta=0
            for e in range(b+1,6):
                if(e==5 and lista[e]==0):
                    contador_janelas_quinta=0

```



```

        break
    if(lista[e]==0):
        contador_janelas_quinta=contador_janelas_quinta+1
    if(e!=b and lista[e]==1):
        break
    if(contador_janelas_quinta>0):
        pular=contador_janelas_quinta+1
    ↵
↪janelas_existentes_quinta=janelas_existentes_quinta+contador_janelas_quinta
print(" ")
print("janelas analisando h1,h2,h3 na quinta =",janela_quinta)
print(" ")
print("janelas existentes na quinta=",janelas_existentes_quinta)

print("-----Sexta-----")
print("h1","h2","h3","j"," p")
for h in range (54,58):
    a=0
    b=0
    c=0
    j=0
    pen=0
    for t in listaIndiceTurma:
        for d in listaIndiceDisciplina:
            a=a+round(x[t,d,p,h].X)
            b=b+round(x[t,d,p,h+1].X)
            c=c+round(x[t,d,p,h+2].X)
        janela_sexta=janela_sexta+(a*(1-b)*c)
        j=a*(1-b)*c
        pen=a*(1-b)*c
    print(a,"",b,"",c,"",j," ",pen)

slot1=0
slot2=0
slot3=0
slot4=0
slot5=0
slot6=0
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
for t in listaIndiceTurma:
    for d in listaIndiceDisciplina:
        slot1=slot1+round(x[t,d,p,54].X)
        slot2=slot2+round(x[t,d,p,55].X)
        slot3=slot3+round(x[t,d,p,56].X)
        slot4=slot4+round(x[t,d,p,57].X)
        slot5=slot5+round(x[t,d,p,58].X)

```

```

        slot6=slot6+round(x[t,d,p,59].X)
lista=[slot1,slot2,slot3,slot4,slot5,slot6]
janelas_existentes_sexta=0
contador_janelas_sexta=0
pular=0
for b in range(0,5):
    if(pular>0):
        pular=pular-1
    if(pular==0):
        if(lista[b]==1):
            contador_janelas_sexta=0
            for e in range(b+1,6):
                if(e==5 and lista[e]==0):
                    contador_janelas_sexta=0
                    break
                if(lista[e]==0):
                    contador_janelas_sexta=contador_janelas_sexta+1
                if(e!=b and lista[e]==1):
                    break
            if(contador_janelas_sexta>0):
                pular=contador_janelas_sexta+1
        □
    ↪janelas_existentes_sexta=janelas_existentes_sexta+contador_janelas_sexta
    print(" ")
    print("janelas analisando h1,h2,h3 na sexta =",janela_sexta)
    print(" ")
    print("janelas existentes na sexta=",janelas_existentes_sexta)

    □
    ↪janela_semanal=janela_segunda+janela_terca+janela_quarta+janela_quinta+janela_sexta
    □
    ↪janela_semanal_real=janelas_existentes_segunda+janelas_existentes_terca+janelas_existentes_

    print("-----Semana-----")
    print("|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA_|")
    ↪="",janela_semanal,"|")
    print("|TOTAL DE JANELAS EXISTENTES NA SEMANA =",janela_semanal_real,"|")
    total=janela_semanal+total
    janela_semanal_total=janela_semanal_total+janela_semanal_real
print(" ")
print("-----FIM-----")
print("|TOTAL DE JANELAS ANALISANDO H1,H2,H3 =",total,"|")
print("|TOTAL DE JANELAS EXISTENTES NESSA SOLUCAO =",janela_semanal_total,"|")
diasAlocados=gp.quicksum(dia_segunda[p].X + dia_terca[p].X + dia_quarta[p].X +
    ↪dia_quinta[p].X + dia_sexta[p].X for p in range(len(professores)))
print("|Dias alocados: ",diasAlocados,"|")

```

-----Analise de resultados-----

```
|-----Siglas-----|
|h1 corresponde a um slot de horario |
|h2 corresponde ao slot sequente de h1 |
|h3 corresponde ao slot sequente de h2 |
|j corresponde se existe janela |
|p corresponde a penalidade atribuido a sequencia de h1,h2,h3 |
|-----|
|penalidade aplicada nesta execucao,se foi adicionada na funcao objetivo|
```

Professor: P1

-----Segunda-----

h1	h2	h3	j	p
1	1	1	0	0
1	1	1	0	0
1	1	1	0	0
1	1	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
1	1	0	0	0
1	0	0	0	0
0	0	1	0	0
0	1	1	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 2

-----Quinta-----

h1	h2	h3	j	p
0	0	0	0	0

```

0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

```

h1 h2 h3 j  p
0  1  1  0  0
1  1  1  0  0
1  1  1  0  0
1  1  1  0  0

```

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

```

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 2 |

```

Professor: P2

-----Segunda-----

```

h1 h2 h3 j  p
1  1  1  0  0
1  1  0  0  0
1  0  0  0  0
0  0  1  0  0

```

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 2

-----Terca-----

```

h1 h2 h3 j  p
1  0  0  0  0
0  0  1  0  0
0  1  1  0  0
1  1  1  0  0

```

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 2

-----Quarta-----

```

h1 h2 h3 j  p
1  1  0  0  0
1  0  0  0  0
0  0  1  0  0
0  1  1  0  0

```

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 2

-----Quinta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
1	1	0	0	0
1	0	0	0	0
0	0	1	0	0
0	1	1	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 2

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 8 |

Professor: P3

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
1	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P4

-----Segunda-----

h1	h2	h3	j	p
1	1	1	0	0
1	1	1	0	0
1	1	0	0	0
1	0	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
----	----	----	---	---

```

1 0 0 0 0
0 0 0 0 0
0 0 1 0 0
0 1 1 0 0

```

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 3

-----Quarta-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

```

h1 h2 h3 j  p
1 1 0 0 0
1 0 0 0 0
0 0 0 0 0
0 0 1 0 0

```

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 3

-----Semana-----

```

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 6 |

```

Professor: P5

-----Segunda-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

0 0 0 0 0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	1	0	0
---	---	---	---	---

0	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	1	0	0
---	---	---	---	---

0	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
----	----	----	---	---

1	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	1	0	0
---	---	---	---	---

0	1	1	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 3

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 3 |

Professor: P6

-----Segunda-----

h1	h2	h3	j	p
0	1	1	0	0
1	1	1	0	0
1	1	1	0	0
1	1	1	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
1	1	1	0	0
1	1	1	0	0
1	1	0	0	0
1	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
1	1	1	0	0
1	1	1	0	0
1	1	1	0	0
1	1	0	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P7

-----Segunda-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
----	----	----	---	---

0	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
----	----	----	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P8

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
1	1	1	0	0
1	1	1	0	0
1	1	1	0	0
1	1	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
----	----	----	---	---

```

1 1 1 0 0
1 1 1 0 0
1 1 1 0 0
1 1 0 0 0

```

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P9

-----Segunda-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

```

h1 h2 h3 j  p
0 1 1 0 0
1 1 1 0 0
1 1 1 0 0
1 1 1 0 0

```

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

```

h1 h2 h3 j  p
0 1 1 0 0
1 1 1 0 0
1 1 0 0 0

```

1 0 0 0 0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P10

-----Segunda-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
----	----	----	---	---

0	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
0	1	1	0	0
1	1	1	0	0
1	1	1	0	0
1	1	1	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P11

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	1	0	0
0	1	1	0	0
1	1	1	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
1	1	0	0	0
1	0	0	0	0
0	0	0	0	0
0	0	1	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 3

-----Sexta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
 |TOTAL DE JANELAS EXISTENTES NA SEMANA = 3 |

Professor: P12

-----Segunda-----

h1	h2	h3	j	p
1	0	0	0	0
0	0	0	0	0

```

0 0 1 0 0
0 1 1 0 0

```

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 3

-----Terca-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

```

h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

```

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

```

h1 h2 h3 j  p
0 0 1 0 0
0 1 1 0 0
1 1 1 0 0
1 1 0 0 0

```

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 3 |

Professor: P13

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
1	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 4

-----Quinta-----

h1	h2	h3	j	p
1	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 4

-----Sexta-----

h1	h2	h3	j	p
1	1	0	0	0
1	0	0	0	0
0	0	0	0	0

0 0 0 0 0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 8 |

Professor: P14

-----Segunda-----

h1	h2	h3	j	p
----	----	----	---	---

1	1	1	0	0
---	---	---	---	---

1	1	1	0	0
---	---	---	---	---

1	1	0	0	0
---	---	---	---	---

1	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
----	----	----	---	---

1	1	1	0	0
---	---	---	---	---

1	1	0	0	0
---	---	---	---	---

1	0	0	0	0
---	---	---	---	---

0	0	1	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 2

-----Quinta-----

h1	h2	h3	j	p
----	----	----	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

0	0	0	0	0
---	---	---	---	---

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	1	0	0
0	1	1	0	0
1	1	1	0	0
1	1	1	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
TOTAL DE JANELAS EXISTENTES NA SEMANA = 2

Professor: P15

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0
0	0	1	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 2

-----Sexta-----

h1	h2	h3	j	p
1	1	0	0	0
1	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
 |TOTAL DE JANELAS EXISTENTES NA SEMANA = 2 |

Professor: P16

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	1	1	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
1	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0

```
0 0 0 0 0
0 0 0 0 0
```

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

```
h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

```
h1 h2 h3 j  p
0 0 1 0 0
0 1 1 0 0
1 1 0 0 0
1 0 0 0 0
```

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

```
|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |
```

Professor: P17

-----Segunda-----

```
h1 h2 h3 j  p
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 1 0 0
```

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

```
h1 h2 h3 j  p
0 1 0 0 0
1 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |

|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

Professor: P18

-----Segunda-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0

janelas analisando h1,h2,h3 na segunda = 0

janelas existentes na segunda = 0

-----Terca-----

h1	h2	h3	j	p
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na terca = 0

janelas existentes na terca= 0

-----Quarta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quarta = 0

janelas existentes na quarta= 0

-----Quinta-----

h1	h2	h3	j	p
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na quinta = 0

janelas existentes na quinta= 0

-----Sexta-----

h1	h2	h3	j	p
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0
0	0	0	0	0

janelas analisando h1,h2,h3 na sexta = 0

janelas existentes na sexta= 0

-----Semana-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 NA SEMANA = 0 |
|TOTAL DE JANELAS EXISTENTES NA SEMANA = 0 |

-----FIM-----

|TOTAL DE JANELAS ANALISANDO H1,H2,H3 = 0 |
|TOTAL DE JANELAS EXISTENTES NESSA SOLUCAO = 37 |
|Dias alocados: <gurobi.LinExpr: 49.0> |

```
[41]: model.write("teste_vespertino_V3.lp")  
      model.dispose()
```