PERT

In [1]:

```
import numpy as np
import pandas as pd
import math
from gamspy import (
    Container,Set,Alias,Parameter,Variable,Equation,Model,Problem,Sense,Opti
    Domain,Number,Sum,Product,Smax,Smin,Ord,Card,SpecialValues,
)
import gamspy.math as gpm

options = Options(variable_listing_limit=0, equation_listing_limit=20)

m = Container(options=options)
```

In [2]:

```
time = Set(m,'time',records=['A','B','C','D','E','F','G','H'])
i = Alias(m,'i',time)
j = Alias(m,'j',time)
hdr = Set(m,'hdr',records=['w','dlo','dup'])

def data_records():
    cols = ["w", "dlo", "dup"]
    idxs = [("A", "B"), ("A", "D"), ("B", "C"), ("B", "D"),
        ("B", "E"), ("C", "E"), ("C", "G"), ("D", "E"),
        ("D", "F"), ("E", "F"), ("E", "G"), ("E", "H"),
        ("F", "H"), ("G", "H")]

    data = np.array([
        [3,   3,  4],
        [4,   2,  3],
        [2,   2,  5],
        [2,   4,  5],
        [3,   6,  8],
        [2,   5,  6],
        [1,   2,  4],
        [3,   5,  7],
        [1,   8,  9],
        [1,   6,  10],
        [1,   4,  7],
        [1,   2,  3],
        [1,   2,  3],
        [1,   4,  5]]
        )

    idxs = pd.MultiIndex.from_tuples(idxs, names=["i", "j"])
    data = pd.DataFrame(data, columns=cols, index=idxs)
    data.reset_index(inplace=True)
    melted_data = data.melt(
        id_vars=["i", "j"], value_vars=["w", "dlo", "dup"]
    )
    return melted_data
```

```
data = Parameter(
    m, name="data", domain=[i, j, hdr], records=data_records()
)
display(data.pivot())
```

		w	dlo	dup
A	B	3.0	3.0	4.0
	D	4.0	2.0	3.0
B	C	2.0	2.0	5.0
	D	2.0	4.0	5.0
	E	3.0	6.0	8.0
C	E	2.0	5.0	6.0
	G	1.0	2.0	4.0
D	E	3.0	5.0	7.0
	F	1.0	8.0	9.0
E	F	1.0	6.0	10.0
	G	1.0	4.0	7.0
	H	1.0	2.0	3.0
F	H	1.0	2.0	3.0
G	H	1.0	4.0	5.0

In [3]:

```
# PUT YOUR CODE HERE - I USED projDur as a variable for the project duratior
t = Variable(m, 't', 'positive', domain=i)
projDur = Variable(m, 'projDur', 'free')

arcs = Set(m, 'arcs', domain=[i, j], records=[('A', 'B'), ('A', 'D'), ('B',

incid = Equation(m, 'incidence', domain=[i, j])
incid[arcs[i, j]] = t[j] >= t[i] + data[i, j, 'dup']

end= Equation(m, 'endTime', domain=i)
end[i] = projDur >= t[i]

cpm = m.addModel('cpm',
    equations=m.getEquations(),
    problem=Problem.LP,
    sense=Sense.MIN,
    objective=projDur,
)

cpm.solve()
```

Out[3]:

	Solver Status	Model Status	Objective	Num of Equations	Num of Variables	Model Type	Solver	Solve Tim
0	Normal	OptimalGlobal	29	21	9	LP	CPLEX	0.00

Critical path is identifed by binding constraints (those with positive multipliers)

In [ ]:
```
critical = Set(m,'critical',domain=i,description="critical activities")
# CODE HERE TO COMPUTE CRITICAL PATH SET
critical[i] = Number(1).where[(Smax(j.where[arcs[j, i]], incid.m[j, i]) >= 1

display(t.toList())

print(f"The path is {critical.toList()}")
```

```
[('A', 0.0),
 ('B', 4.0),
 ('C', 9.0),
 ('D', 9.0),
 ('E', 16.0),
 ('F', 26.0),
 ('G', 29.0),
 ('H', 29.0)]
The path is ['A', 'B', 'D', 'E', 'F', 'H']
```

In [7]:
```
d = Variable(m, 'd', domain=[i, j])
cost = Variable(m, 'cost')

incid = Equation(m, 'incidence', domain=[i, j])
incid[arcs[i, j]] = t[j] >= t[i] + d[i, j]

d.lo[i, j] = data[i, j, 'dlo']
d.up[i, j] = data[i, j, 'dup']

cost_eq = Equation(m, 'cost_eq')
cost_eq[:] = cost == Sum(arcs[i, j], (2 / (data[i, j, 'dup'] - data[i, j, 'c

cpmred = Model(m, 'cpmred',
    equations=m.getEquations(),
    problem=Problem.LP,
    sense=Sense.MIN,
    objective=cost,
)
```

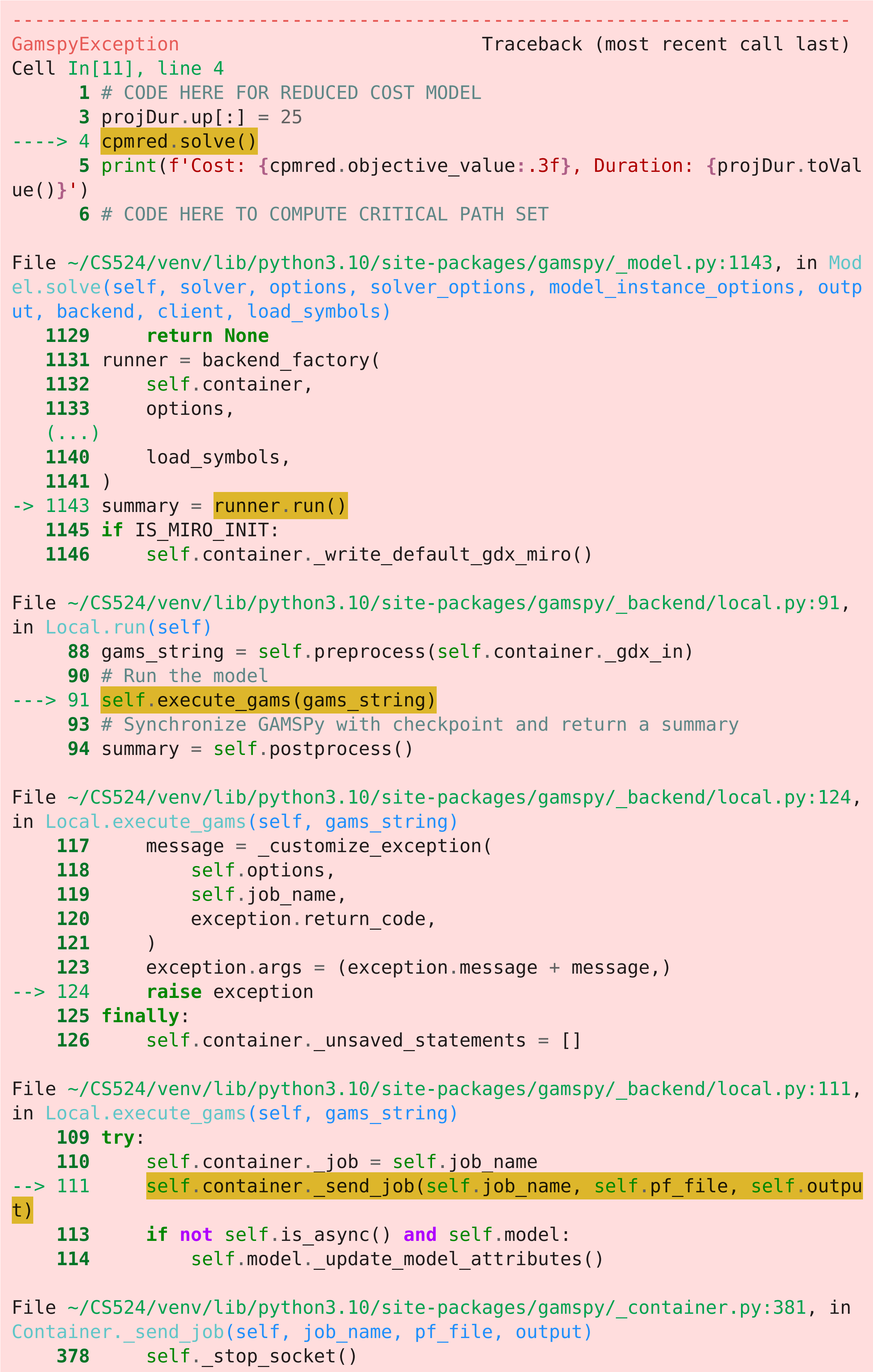
In [10]:
```
print("I am confident in my above code, I do not know why I am getting a Gam
```

```
I am confident in my above code, I do not know why I am getting a GamspyExce
ption
```

Now allow reduction of durations at a cost

In [11]:
```
# CODE HERE FOR REDUCED COST MODEL

projDur.up[:] = 25
cpmred.solve()
```

```
print(f'Cost: {cpmred.objective_value:.3f}, Duration: {projDur.toValue()}')
# CODE HERE TO COMPUTE CRITICAL PATH SET

display(t.toList(),critical.toList())

projDur.up[:] = 20
cpmred.solve()
print(f'Cost: {cpmred.objective_value:.3f}, Duration: {projDur.toValue()}')
# CODE HERE TO COMPUTE NEW CRITICAL PATH SET

display(t.toList(),critical.toList())
```



```
---------------------------------------------------------------------------
GamspyException                           Traceback (most recent call last)
Cell In[11], line 4
      1 # CODE HERE FOR REDUCED COST MODEL
      3 projDur.up[:] = 25
----> 4 cpmred.solve()
      5 print(f'Cost: {cpmred.objective_value:.3f}, Duration: {projDur.toVal
ue()}')
      6 # CODE HERE TO COMPUTE CRITICAL PATH SET

File ~/CS524/venv/lib/python3.10/site-packages/gamspy/_model.py:1143, in Mod
el.solve(self, solver, options, solver_options, model_instance_options, outp
ut, backend, client, load_symbols)
   1129     return None
   1131 runner = backend_factory(
   1132     self.container,
   1133     options,
   (...)
   1140     load_symbols,
   1141 )
-> 1143 summary = runner.run()
   1145 if IS_MIRO_INIT:
   1146     self.container._write_default_gdx_miro()

File ~/CS524/venv/lib/python3.10/site-packages/gamspy/_backend/local.py:91,
in Local.run(self)
     88 gams_string = self.preprocess(self.container._gdx_in)
     90 # Run the model
---> 91 self.execute_gams(gams_string)
     93 # Synchronize GAMSPy with checkpoint and return a summary
     94 summary = self.postprocess()

File ~/CS524/venv/lib/python3.10/site-packages/gamspy/_backend/local.py:124,
in Local.execute_gams(self, gams_string)
    117     message = _customize_exception(
    118         self.options,
    119         self.job_name,
    120         exception.return_code,
    121     )
    123     exception.args = (exception.message + message,)
--> 124     raise exception
    125 finally:
    126     self.container._unsaved_statements = []

File ~/CS524/venv/lib/python3.10/site-packages/gamspy/_backend/local.py:111,
in Local.execute_gams(self, gams_string)
    109 try:
    110     self.container._job = self.job_name
--> 111     self.container._send_job(self.job_name, self.pf_file, self.outpu
t)
    113     if not self.is_async() and self.model:
    114         self.model._update_model_attributes()

File ~/CS524/venv/lib/python3.10/site-packages/gamspy/_container.py:381, in
Container._send_job(self, job_name, pf_file, output)
    378     self._stop_socket()
    379     return
--> 381 check_response(response, job_name)

File ~/CS524/venv/lib/python3.10/site-packages/gamspy/_container.py:181, in
check_response(response, job_name)
    179 except IndexError:
    180     info = ""
--> 181 raise GamspyException(
    182     f'{info} Check {job_name + ".lst"} for more information.',
    183     return_code,
    184 )

GamspyException: There was an execution error. Check /tmp/tmp05_yy5br/_b4792
743-0d52-490d-be2d-798b43a5c3d5.lst for more information.

=============
Error Summary
=============

**** Exec Error at line 295: division by zero (0)

---- cost_eq  =E=

cost_eq..  cost =E= UNDF ; (LHS = UNDF)

GAMS 47.6.0  c2de9d6d Sep 12, 2024          LEX-LEG x86 64bit/Linux - 10/28/
24 16:41:02 Page 3
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
```

```
In [ ]:
```