Name:-Tej Sutar Batch:-H Roll no:-176 Date:-25/01/25 Practical no:-11 & 12 Practical name:- Study of Operation Research in Python(LPP)

Q1) Write a python program to display the following LPP by using pulp module and simplex method. Find its optimal solution if exist. Max Z = x + 2y + z subject to x + 2y + 2z ≤ 1, 3x + 2y + z ≥ 8, x ≥ 0,y ≥ 0,z ≥ 0

In [1]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
z=LpVariable("z",lowBound=0)
lpp+=(x+2*y+2*z<=1)
lpp+=(3*x+2*y+z>=8)
lpp+=(x+2*y+z)
lpp
```

Out[1]:
```
LPP:
MAXIMIZE
1*x + 2*y + 1*z + 0
SUBJECT TO
_C1: x + 2 y + 2 z <= 1

_C2: 3 x + 2 y + z >= 8

VARIABLES
x Continuous
y Continuous
z Continuous
```

In [2]:
```
lpp.solve()
```

Out[2]:
```
-1
```

In [3]:
```
lpp.objective.value()
```

Out[3]:
```
2.6666667
```

In [4]:
```
x.value()
```

Out[4]:
```
2.6666667
```

In [5]:
```
y.value()
```

Out[5]:
```
0.0
```

Q2) Write a python program to display the following LPP by using pulp module and simplex method. Find its optimal solution if exist. Max Z = 3x + 5y + 4z subject to 2x + 3y ≤ 8 2y + 5z ≤ 10 3x + 2y + 4z ≤ 15 x ≥ 0,y ≥ 0,z ≥ 0.

In [6]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
z=LpVariable("z",lowBound=0)
lpp+=(2*x+3*y<=8)
lpp+=(2*y+5*z<=10)
lpp+=(3*x+2*y+4*z<=15)
lpp+=(3*x+5*y+4*z)
lpp
```

Out[6]:
```
LPP:
MAXIMIZE
3*x + 5*y + 4*z + 0
SUBJECT TO
_C1: 2 x + 3 y <= 8

_C2: 2 y + 5 z <= 10

_C3: 3 x + 2 y + 4 z <= 15

VARIABLES
x Continuous
y Continuous
z Continuous
```

In [7]:
```
lpp.solve()
```

Out[7]:
```
1
```

In [8]:
```
lpp.objective.value()
```

Out[8]:
```
18.658536500000004
```

In [9]:
```
x.value()
```

Out[9]:
```
2.1707317
```

In [10]:
```
y.value()
```

Out[10]:
```
1.2195122
```

Q3) Write a python program to display the following LPP by using pulp module and simplex method. Find its optimal solution if exist. Max Z = 4x + y + 3z + 5w subject to 4x + 6y−5z−4w ≥ 20 −3x−2y + 4z + w ≤ 10 −8x−3y + 3z + 2w ≤ 20 x ≥ 0,y ≥ 0,z ≥ 0,w ≥ 0.

In [11]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
z=LpVariable("z",lowBound=0)
w=LpVariable("w",lowBound=0)
lpp+=(4*x+6*y-5*z-4*w>=20)
lpp+=((-3)*x-2*y+4*z+w<=10)
lpp+=((-8)*x-3*y+3*z+2*w<=20)
lpp+=(4*x+y+3*z+5*w)
lpp
```

Out[11]:
```
LPP:
MAXIMIZE
5*w + 4*x + 1*y + 3*z + 0
SUBJECT TO
_C1: - 4 w + 4 x + 6 y - 5 z >= 20

_C2: w - 3 x - 2 y + 4 z <= 10

_C3: 2 w - 8 x - 3 y + 3 z <= 20

VARIABLES
w Continuous
x Continuous
y Continuous
z Continuous
```

In [12]:
```
lpp.solve()
```

Out[12]:
```
-2
```

In [13]:
```
lpp.objective.value()
```

Out[13]:
```
3.3333333
```

In [14]:
```
x.value()
```

Out[14]:
```
0.0
```

In [15]:
```
y.value()
```

Out[15]:
```
3.3333333
```

Q4) Write a python program to display the following LPP by using pulp module and simplex method. Find its optimal solution if exist. Max Z = 3x + 2y + 5z subject to x + 2y + z ≤ 430 3x + 4z ≤ 460 x + 4y ≤ 120 x ≥ 0,y ≥ 0,z ≥ 0

In [16]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMaximize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
z=LpVariable("z",lowBound=0)
lpp+=(x+2*y+z<=430)
lpp+=(3*x+4*z<=460)
lpp+=(x+4*y<=120)
lpp+=(3*x+2*y+5*z)
lpp
```

Out[16]:
```
LPP:
MAXIMIZE
3*x + 2*y + 5*z + 0
SUBJECT TO
_C1: x + 2 y + z <= 430

_C2: 3 x + 4 z <= 460

_C3: x + 4 y <= 120

VARIABLES
x Continuous
y Continuous
z Continuous
```

In [17]:
```
lpp.solve()
```

Out[17]:
```
1
```

In [18]:
```
lpp.objective.value()
```

Out[18]:
```
635.0
```

In [19]:
```
x.value()
```

Out[19]:
```
0.0
```

In [20]:
```
y.value()
```

Out[20]:
```
30.0
```

Q5) Write a Python program to solve the following LPP: Min Z = x + y subject to x ≥ 6 y ≥ 6 x + y ≤ 11 x ≥ 0,y ≥ 0

In [21]:
```
from pulp import*
lpp=LpProblem(name='LPP',sense=LpMinimize)
x=LpVariable("x",lowBound=0)
y=LpVariable("y",lowBound=0)
lpp+=(x>=6)
lpp+=(y>=6)
lpp+=(x+y<=11)
lpp+=(x+y)
lpp
```

Out[21]:
```
LPP:
MINIMIZE
1*x + 1*y + 0
SUBJECT TO
_C1: x >= 6

_C2: y >= 6

_C3: x + y <= 11

VARIABLES
x Continuous
y Continuous
```

In [22]:
```
lpp.solve()
```

Out[22]:
```
-1
```

In [23]:
```
lpp.objective.value()
```

Out[23]:
```
12.0
```

In [24]:
```
x.value()
```

Out[24]:
```
6.0
```

In [25]:
```
y.value()
```

Out[25]:
```
6.0
```