HW 5 #4-(d)

the LP problem.

$$
\begin{array}{rlllll}
\text{maximize } z = 300x_1 + 400x_2 + 1000x_3 & & & & & \\
 & x_1 & +x_2 & & \leq & 40 \\
 & & x_2 & +2x_3 & \leq & 30 \\
\text{subject to} & 2x_1 & & & \leq & 20 \\
 & x_1 & & +2x_3 & \leq & 10 \\
 & & 2x_2 & +x_3 & \leq & 10 \\
 & x_1, x_2, x_3 \geq 0 & & & &
\end{array}
$$

In [1]:

```
import sys
!{sys.executable} -m pip install pulp
```

```
Requirement already satisfied: pulp in /opt/conda/lib/python3.8/site
-packages (2.3.1)
Requirement already satisfied: amply>=0.1.2 in /opt/conda/lib/python
3.8/site-packages (from pulp) (0.1.4)
Requirement already satisfied: pyparsing in /opt/conda/lib/python3.8
/site-packages (from amply>=0.1.2->pulp) (2.4.7)
Requirement already satisfied: docutils>=0.3 in /opt/conda/lib/pytho
n3.8/site-packages (from amply>=0.1.2->pulp) (0.15.2)
```

In [2]:

```
import pulp
```

In [3]:

```
# Create a LP Minimization problem
Lp_prob = pulp.LpProblem('Your_LP_Problem', pulp.LpMaximize)  # We set
up theproblem using the command LpProblem in the PuLP package.
```

In [4]:

```
# Create problem Decision Variables
x_1 = pulp.LpVariable("x_1")
x_2 = pulp.LpVariable("x_2")
x_3 = pulp.LpVariable("x_3")
```

In [5]:

```
# Objective Function
Lp_prob += 300* x_1 + 400*x_2 + 1000 * x_3

# Constraints:
Lp_prob += x_1 + x_2  <= 40
Lp_prob += x_2 + 2*x_3  <= 30
Lp_prob += 2*x_1  <= 20
Lp_prob += x_1 + 2*x_3 <= 10
Lp_prob += 2*x_2 + x_3 <= 10
Lp_prob += x_1 >= 0
Lp_prob += x_2 >= 0
Lp_prob += x_3 >= 0
```

In [6]:

```
# Display the problem
print(Lp_prob)
```

```
Your_LP_Problem:
MAXIMIZE
300*x_1 + 400*x_2 + 1000*x_3 + 0
SUBJECT TO
_C1: x_1 + x_2 <= 40

_C2: x_2 + 2 x_3 <= 30

_C3: 2 x_1 <= 20

_C4: x_1 + 2 x_3 <= 10

_C5: 2 x_2 + x_3 <= 10

_C6: x_1 >= 0

_C7: x_2 >= 0

_C8: x_3 >= 0

VARIABLES
x_1 free Continuous
x_2 free Continuous
x_3 free Continuous
```

In [7]:

```
Lp_prob.solve()
pulp.LpStatus[Lp_prob.status]
```

Out[7]:

```
'Optimal'
```

In [8]:

```
# Printing the final solution
print("x_1=", pulp.value(x_1), "x_2=", pulp.value(x_2),  "x_3=", pulp.
value(x_3),  "z=", pulp.value(Lp_prob.objective))
```

```
x_1= 0.0 x_2= 2.5 x_3= 5.0 z= 6000.0
```

Another way to show the solutions:

In [9]:

```
for a in Lp_prob.variables():
        print(a.name, "=", a.varValue)
print("Optimal value is z = ", pulp.value(Lp_prob.objective))
```

```
x_1 = 0.0
x_2 = 2.5
x_3 = 5.0
Optimal value is z =  6000.0
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: