

A decorative network diagram in the top-left corner of the slide. It features a complex web of interconnected nodes and edges. The nodes are represented by small circles, some of which are solid blue, green, or red, while others are hollow with a blue or green outline. The edges are thin, light gray lines. The overall shape of the network is roughly triangular, pointing towards the top-left corner.

1 Workshop

A decorative network diagram in the bottom-right corner of the slide. It features a complex web of interconnected nodes and edges. The nodes are represented by small circles, some of which are solid blue, green, or red, while others are hollow with a blue or green outline. The edges are thin, light gray lines. The overall shape of the network is roughly triangular, pointing towards the bottom-right corner.

Задача наполнения рюкзака



1.3 кг 2500\$



0.25 кг 1100\$



0.25 кг 1100\$



1.6 кг 1900\$



0.48 кг 1200\$



0.04 кг 430\$



0.04 кг 430\$



Вместимость
рюкзак 1.8 кг

<https://www.apple.com/>

Задача наполнения рюкзака

- Решить задачу наполнения рюкзака для различного количества предметов

Maximize $\sum_{i \in I} v_i x_i$

При этом, $\sum_{i \in I} w_i x_i \leq K$, $x_i \in \{0,1\}$, $(i \in 1..n)$

Вход

n	K
v_1	w_1
v_2	w_2
...	...
v_n	w_n

Выход

Целевая функция		Оптимальность			
x1	x2	x3	...	xn	

Задача наполнения рюкзака

- Решить задачу наполнения рюкзака для различного количества предметов

Maximize $\sum_{i \in I} v_i x_i$

При этом, $\sum_{i \in I} w_i x_i \leq K, x_i \in \{0,1\}, (i \in 1..n)$

Вход

n	K
v_1	w_1
v_2	w_2
...	...
v_n	w_n

4	11
8	4
10	5
15	8
4	3

Выход

Целевая функция	Оптимальность
19	0

x1	x2	x3	x4
0	0	1	1

Синтаксис PySCIPOpt и MIP

https://scipopt.github.io/PySCIPOpt/docs/html/md_README.html

<https://python-mip.readthedocs.io/en/latest/quickstart.html>

Импорт

```
from pycipopt import Model, quicksum
```

```
from mip import Model, xsum, maximize, BINARY
```

Объявление модели

```
m = Model()
```

```
m = Model("name")
```

Объявление переменных

```
select={}
for j in COLUMN:
    select[j] = m.addVar(vtype="B"|"I", name="select(%s)"%j)
```

```
x = [m.add_var(var_type=BINARY|INTEGER) for i in COLUMN]
```

Ограничение

```
m.addCons(quicksum(EXPRESSION
                    for i in COLUMN) <= capacity, name="Capacity_limit")
```

```
m += xsum(EXPRESSION for i in COLUMN) <= capacity
```

Целевая функция

```
obj = quicksum(EXPRESSION
                for i in COLUMN)
m.setObjective(obj, "maximize")
m.optimize()
```

```
m.objective = maximize(EXPRESSION for i in COLUMN)
m.optimize()
```

Extract в Excel. Пример для PySCIPOpt

```
: out=[]  
  for i in things_data['thing_id']:  
      out.append(round(int(m.getVal(select[i]))))  
  
: df = pd.DataFrame(out,columns=["var"])  
  display(df)  
  df.to_excel (r'C:\Users\VZ\Desktop\solution.xlsx', index = False, header=True)
```