

Работа с вложени цикли

По-сложни задачи



СофтУни

Преподавателски екип



SoftUni



Софтуерен университет

<https://softuni.bg>

1. Преговор
2. Вложени цикли
3. Решаване на задачи





Преговор

1. Колко пъти ще се изпише "SoftUni" на конзолата след изпълнението на следния код:

```
i = 0  
while i <= 5:  
    print("SoftUni")  
    i += 1
```

5

0

6

4

2. Колко пъти ще се изпише "SoftUni" на конзолата след изпълнението на следния код:

```
i = 0
while i == 0:
    print("SoftUni")
    if i == 1:
        break
```

0

1

Безброй
МНОГО ПЪТИ

100000

3. Какъв ще е резултатът от изпълнението на следния код:

```
i = 0
while i <= 6:
    i += 1
    if i % 2 == 0:
        print(i, end="")
```

024

24

246

123456

4. Какъв ще е резултатът от изпълнението на следния код:

```
i = 0

while i < 5:
    if i == 0:
        print(i)
        break
    i += 1
```

1

Error

0

01234



Вложени цикли

Пример – часовник (1)

Часовете се променят
когато минутите
надвишат 59

19:03

Докато минутите се
променят часовете
остават същите



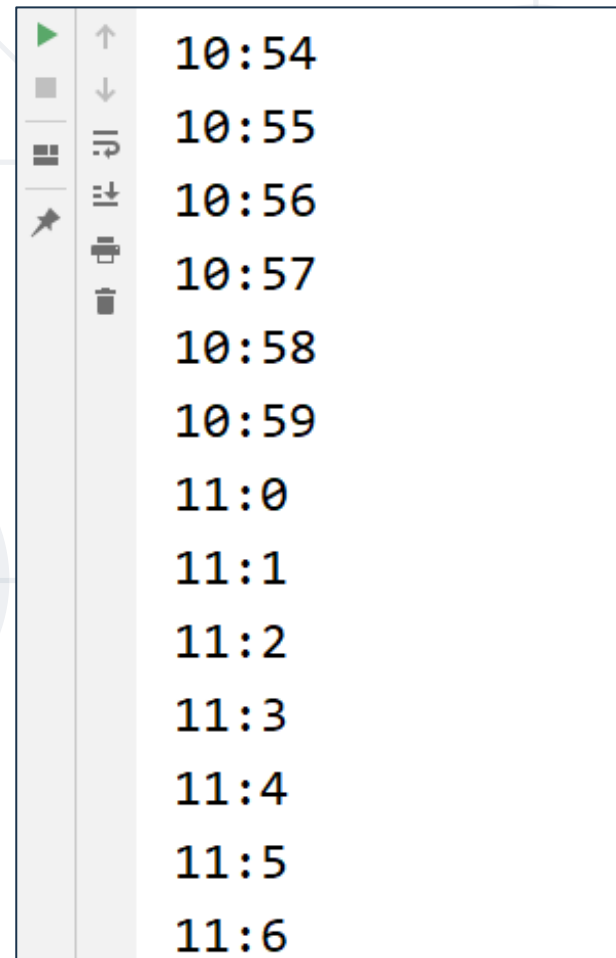
Как може да си направим часовник

с код?
Демо

Пример – часовник (2)

- Външният цикъл отговаря за часовете, а вътрешния за минутите

```
for h in range(24):  
    for m in range(60):  
        print(f"{h}:{m}")
```



```
10:54  
10:55  
10:56  
10:57  
10:58  
10:59  
11:0  
11:1  
11:2  
11:3  
11:4  
11:5  
11:6
```

- За всяка итерация на външния цикъл вложения се изпълнява n - на брой пъти

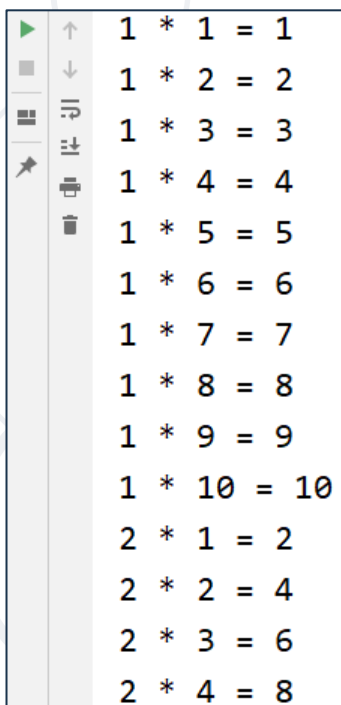
```
for i in range(n):  
    for j in range(n):  
        ...
```

Имената на
променливите трябва
да бъдат различни



Таблица за умножение – условие

- Отпечатайте на конзолата таблицата за умножение за числата от 1 до 10
- Изход:



```
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
1 * 4 = 4
1 * 5 = 5
1 * 6 = 6
1 * 7 = 7
1 * 8 = 8
1 * 9 = 9
1 * 10 = 10
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
```



```
for x in range(1, 11):  
    for y in range(1, 11):  
        product = x * y  
        print(f"{x} * {y} = {product}")
```

- За прекъсване на вложени цикли, използваме булеви променливи.

```
flag = False
for i in range(n):
    for j in range(n):
        if condition:
            flag = True
            break
    if flag:
        break
```

Външният цикъл ще се прекъсне, само ако стойността на flag бъде true

- Напишете програма, която проверява всички възможни комбинации от двойка числа в даден интервал
 - Ако се намери комбинация, чийто **сбор от числата е равен** на дадено **магическо число** на изхода **се отпечатва съобщение** и програмата приключва изпълнение
 - Ако не се намери **ниито една комбинация**, отговаряща на условието се отпечатва **съобщение, че не е намерено**

Сума от две числа – условие (2)

- Примерен вход и изход:

1
10
5



Combination N:4 (1 + 4 = 5)

23
24
20



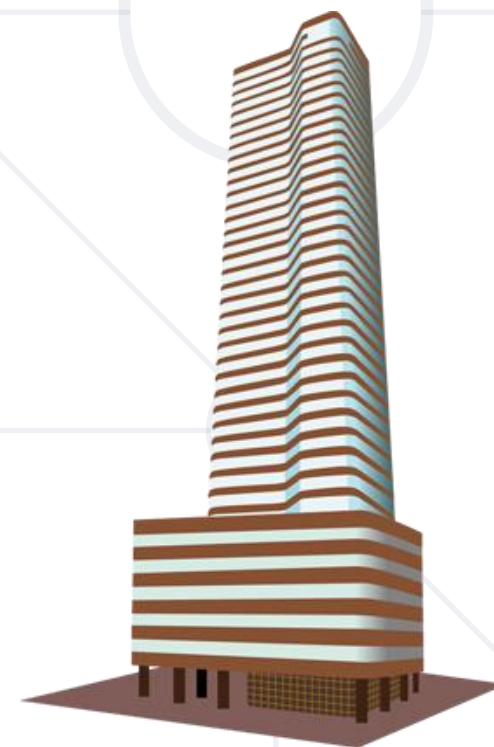
4 combinations - neither equals 20

Сума от две числа – решение

```
starting_number = int(input())
final_number = int(input())
magic_number = int(input())
combinations = 0
is_found = False
for i in range(starting_number, final_number + 1):
    for j in range(starting_number, final_number + 1):
        combinations += 1
        if i + j == magic_number:
            print(f"Combination N:{combinations} ({i} + {j} = {magic_number})")
            is_found = True
            break
    if is_found:
        break
# finish logic
```

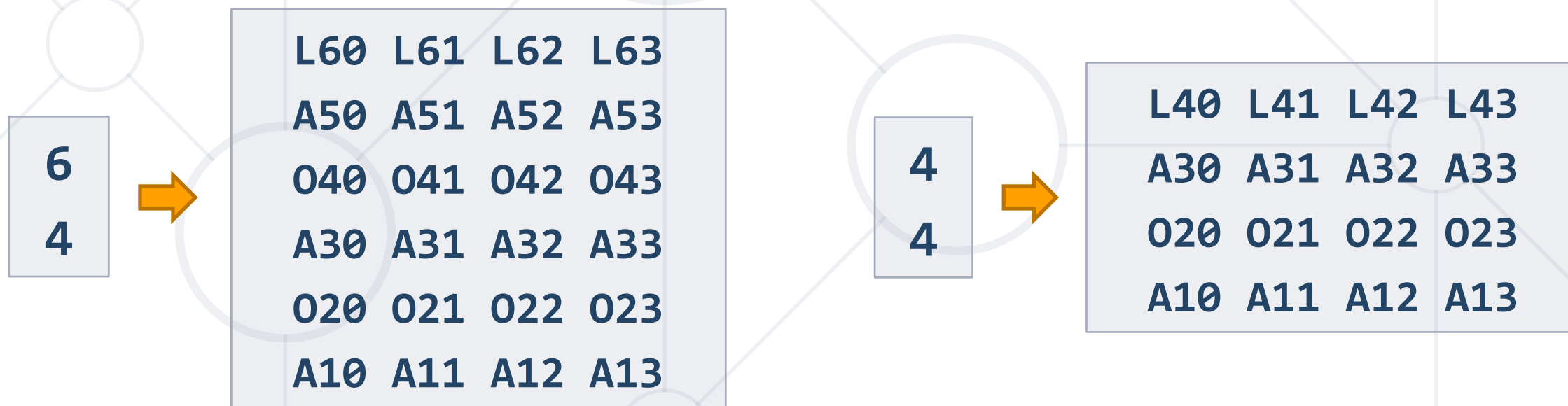
Ако намерим комбинация, прекъсваме вътрешният цикъл

- Напишете програма, която извежда номерата на стаите в една сграда (в низходящ ред)
 - На всеки **четен** етаж има само **офиси**
 - На всеки **нечетен** етаж има само **апартаменти**
- Етажите се означават по следния начин:
 - Апартаменти: "А{номер на етажа}{номер на апартамента}"
 - Офиси: "О{номер на етажа}{номер на офиса}"
 - Номерата им винаги започват с 0



Сграда – условие (2)

- На последният етаж винаги има големи апартаменти, които се означават с 'L', вместо с 'A'
- Ако има само един етаж, то има само големи апартаменти
- Примерен вход и изход:



```
floors = int(input())  
rooms = int(input())  
for i in range(floors, 0, -1):
```

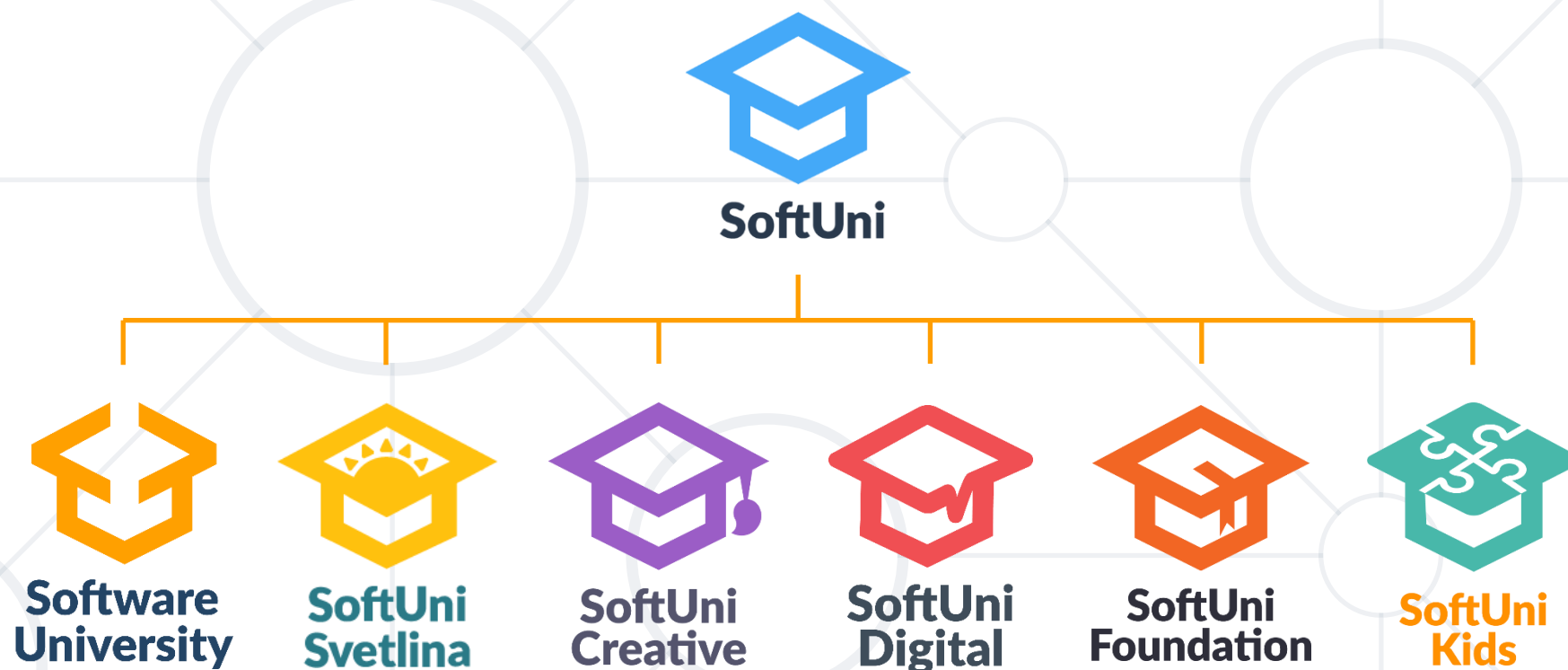
```
    for j in range(0, rooms):  
        if i == floors:  
            print("L{0}{1} ".format(i, j), end = "")  
            # TODO: print according to floor number  
        print("")
```

Вложеният
цикъл итерира
стаите

- Какво представляват вложените цикли
- Конструкция на вложени цикли
- Прекъсване на вложени цикли



Въпроси?



- Този курс (презентации, примери, демонстрационен код, упражнения, домашни, видео и други активи) представлява **защитено авторско съдържание**
- Нерегламентирано копиране, разпространение или използване е незаконно
- © СофтУни – <https://softuni.org>
- © Софтуерен университет – <https://softuni.bg>



- Софтуерен университет – качествено образование, професия и работа за софтуерни инженери
 - softuni.bg
- Фондация "Софтуерен университет"
 - softuni.foundation
- Софтуерен университет @ Facebook
 - facebook.com/SoftwareUniversity
- Дискусионни форуми на СофтУни
 - forum.softuni.bg



Software University

