

# Упражнение: Повторения с цикли – For-цикъл

Задачи за упражнение в клас и за домашно към курса ["Основи на програмирането" @ СофтУни](#).

Тествайте решенията си в Judge системата: <https://judge.softuni.bg/Contests/2418>

## 1. Числа до 1000, завършващи на 7

Напишете програма, която отпечата числата в диапазона **от 1 до 1000**, които **завършват на 7**.

вход	изход
(няма)	7 17 27 ... 997

### Насоки

1. Едно число завършва на 7, когато резултатът от **модулното деление на числото и 10** е **равен на 7**;
2. Направете **for цикъл** от 1 до 1000 и проверете за всяко число дали завършва на 7. Изпълните условията числа принтирайте на конзолата

```
for num in range(1, 1001):  
    if num % 10 == 7:  
        print(num)
```

Можете да използвате и **цикъл със стъпка**:

```
for num in range(7, 1001, 10):  
    print(num)
```

## 2. Елемент, равен на сумата на останалите

Да се напише програма, която чете **n-на брой** цели числа, въведени от потребителя, и проверява дали сред тях съществува число, което е равно на сумата на всички останали.

- Ако има такъв елемент печата **"Yes"** и на нов ред **"Sum = " + неговата стойност**
- Ако няма такъв елемент печата **"No"** и на нов ред **"Diff = " + разликата между най-големия елемент и сумата на останалите** (по абсолютна стойност)

### Примерен вход и изход

вход	изход	коментари
7 3 4 1 1 2 12	Yes Sum = 12	3 + 4 + 1 + 2 + 1 + 1 = 12

1		
4 6 1 2 3	Yes Sum = 6	$1 + 2 + 3 = 12$
3 1 1 10	No Diff = 8	$ 10 - (1 + 1)  = 8$
3 5 5 1	No Diff = 1	$ 5 - (5 + 1)  = 1$
3 1 1 1	No Diff = 1	

## Насоки

- Създайте две помощни променливи:
  - "max\_num" с много ниска първоначална стойност, в която да пазите най-голямото от прочетените числа;
  - "sum\_numbers" с първоначална стойност 0, в която да пазите сумата от прочетените числа.

```
max_num = -sys.maxsize
sum_numbers = 0
```

- Прочетете броя числа, които ще се въведат на конзолата – **n**, и направете **for** цикъл от **0 до n**, като на всяко завъртане четете число **num**:

```
n = int(input())

for i in range(0, n):
    num = int(input())
```

- Направете проверка дали прочетеното число е по-голямо от "max\_num". Ако е по-голямо, приравнете стойността на "max\_num" към неговата. След което добавете стойността на прочетеното число към "sum\_numbers":

```

for i in range(0, n):
    num = int(input())

    if num > max_num:
        max_num = num

    sum_numbers += num

```

4. След цикъла проверете дали "max\_num" е равно на сумата от всички числа, от която е извадено "max\_num":

```

if max_num == sum_numbers - max_num:
    print("Yes")
    print(f"Sum= {sum_numbers}")
else:
    print("No")
    sum_numbers = sum_numbers - max_num
    print(f"Diff= {abs(max_num - sum_numbers)}")

```

## Примерни изпитни задачи

### 3. Хистограма

Дадени са **n** цели числа в интервала [1...1000]. От тях някакъв процент **p1** са под 200, друг процент **p2** са от 200 до 399, друг процент **p3** са от 400 до 599, друг процент **p4** са от 600 до 799 и останалите **p5** процента са от 800 нагоре. Да се напише програма, която изчислява и отпечатва процентите **p1**, **p2**, **p3**, **p4** и **p5**.

**Пример:** имаме **n = 20** числа: 53, 7, 56, 180, 450, 920, 12, 7, 150, 250, 680, 2, 600, 200, 800, 799, 199, 46, 128, 65. Получаваме следното разпределение и визуализация:

Диапазон	Числа в диапазона	Брой числа	Процент
< 200	53, 7, 56, 180, 12, 7, 150, 2, 199, 46, 128, 65	12	$p1 = 12 / 20 * 100 = 60.00\%$
200 ... 399	250, 200	2	$p2 = 2 / 20 * 100 = 10.00\%$
400 ... 599	450	1	$p3 = 1 / 20 * 100 = 5.00\%$
600 ... 799	680, 600, 799	3	$p4 = 3 / 20 * 100 = 15.00\%$
≥ 800	920, 800	2	$p5 = 2 / 20 * 100 = 10.00\%$

### Вход

На първия ред от входа стои цялото число **n** ( $1 \leq n \leq 1000$ ) – брой числа. На следващите **n** реда стои по едно цяло число в интервала [1...1000] – числата върху които да бъде изчислена хистограмата.

### Изход

Да се отпечата на конзолата **хистограмата** – **5 реда**, всеки от които съдържа число между 0% и 100%, с точност две цифри след десетичната точка, например 25.00%, 66.67%, 57.14%.

## Примерен вход и изход

Вход	Изход	Вход	Изход	Вход	Изход	Вход	Изход	Вход	Изход
3	66.67%	4	75.00%	7	14.29%	9	33.33%	14	57.14%
1	0.00%	53	0.00%	800	28.57%	367	33.33%	53	14.29%
2	0.00%	7	0.00%	801	14.29%	99	11.11%	7	7.14%
999	0.00%	56	0.00%	250	14.29%	200	11.11%	56	14.29%
	33.33%	999	25.00%	199	28.57%	799	11.11%	180	7.14%
				399		999		450	
				599		333		920	
				799		555		12	
						111		7	
						9		150	
								250	
								680	
								2	
								600	
								200	

## 4. Умната Лили

Лили вече е на **N години**. За всеки свой **рожден ден** тя получава подарък.

- За нечетните рождени дни (**1, 3, 5...n**) получава **играчки**.
- За четните рождени дни (**2, 4, 6...n**) получава **пари**.

За **втория рожден ден** получава **10.00 лв.**, като **сумата се увеличава с 10.00 лв.**, за всеки следващ **четен рожден ден** (**2 -> 10, 4 -> 20, 6 -> 30...** и т.н.). През годините Лили тайно е спестявала парите. **Братът на Лили, в годините**, които тя **получава пари**, **взима по 1.00 лев** от тях. Лили **продала играчките** получени през годините, **всяка за P лева** и **добавила сумата към спестените пари**. С парите искала да си **купи пералня за X лева**. Напишете програма, която да пресмята, **колко пари е събрала** и дали **й стигат да си купи пералня**.

## Вход

Програмата прочита **3 числа**, въведени от потребителя, на отделни редове:

- **Възрастта на Лили** - **цяло число** в интервала **[1...77]**
- **Цената на пералнята** - **число** в интервала **[1.00...10 000.00]**
- **Единична цена на играчка** - **цяло число** в интервала **[0...40]**

## Изход

Да се отпечата на конзолата един ред:

- Ако парите на Лили са достатъчни:
  - **"Yes! {N}"** - където **N** е остатъка пари след покупката
- Ако парите не са достатъчни:
  - **"No! {M}"** - където **M** е сумата, която не достига

Числата **N** и **M** трябва да са **форматирани до вторият знак след десетичната запетая**.

## Примерен вход и изход

вход	изход	Коментари
10	Yes! 5.00	Първи рожден ден получава играчка

170.00 6		<p>Втори рожден ден получава пари (10 лв.)</p> <p>Трети рожден ден получава играчка</p> <p>Четвърти рожден ден получава пари (20 лв.)</p> <p>Пети рожден ден получава играчка</p> <p>Шести рожден ден получава пари (30 лв.)</p> <p>Седми рожден ден получава играчка</p> <p>Осми рожден ден получава пари (40 лв.)</p> <p>Девети рожден ден получава играчка</p> <p>Десети рожден ден получава пари (50 лв.)</p> <p>Спестила е: <math>10 + 20 + 30 + 40 + 50 = 150</math> лв.</p> <p>Продаде е 5 броя играчки * 6 лв. = 30 лв.</p> <p>Брат ѝ взел 5 пъти * 1 лев = 5 лв.</p> <p>Общо спестени пари: <math>(150 + 30) - 5 = 175</math> лв.</p> <p>175 &gt;= 170 (цената на пералнята)</p> <p>=&gt; успяла е да я купи</p> <p>Остават: <math>175 - 170 = 5</math> лв.</p>
21 1570.98 3	No! 997.98	<p>Спестила е 550 лв.</p> <p>Продаде е 11 играчки * 3 лв. = 33 лв.</p> <p>Брат ѝ взимал 10 години * 1 лев = 10 лв.</p> <p>Общо спестени пари: <math>(550 + 33) - 10 = 573</math> лв.</p> <p>573 &lt; 1570.98 – не е успяла да купи пералня</p> <p>Нужни пари: <math>1570.98 - 573 = 997.98</math> лв.</p>

## 5. Заплата

Шеф на компания забелязва че все повече служители прекарват време в сайтове, които ги разсейват. За да предотврати това, той въвежда изненадващи проверки на отворените табове на брауъра на служителите си.

Според отворения сайт в таба се налагат следните глоби:

- "Facebook" -> 150 лв.
- "Instagram" -> 100 лв.
- "Reddit" -> 50 лв.

От конзолата се четат два реда:

- Брой отворени табове в брауъра n - цяло число в интервала [1...10]
- Заплата - число в интервала [500...1500]

След това n – на брой пъти се чете име на уебсайт – текст

## Изход

- Ако по време на проверката заплатата стане по-малка или равна на 0 лева, на конзолата се изписва "You have lost your salary." и програмата приключва.
- В противен случай след проверката на конзолата се изписва остатъкът от заплатата (да се изпише като цяло число).

## Примерен вход и изход

Вход	Изход	Обяснения
10 750	You have lost your salary.	Има 10 отворени таба в брауъра. Първоначалната заплата е: 750

Facebook Dev.bg Instagram Facebook Reddit Facebook Facebook		За първия таб -> Facebook -> глоба <b>150</b> лв. (заплата: <b>750</b> – <b>150</b> = <b>600</b> ) За втория таб -> Dev.bg -> няма глоба За третия таб -> Instagram -> глоба <b>100</b> лв. (заплата: <b>600</b> – <b>100</b> = <b>500</b> ) За четвъртия таб -> Facebook -> глоба <b>150</b> лв. (заплата: <b>500</b> – <b>150</b> = <b>350</b> ) За петия таб -> Reddit -> глоба <b>50</b> лв. (заплата: <b>350</b> – <b>50</b> = <b>300</b> ) За шестия таб -> Facebook -> глоба <b>150</b> лв. (заплата: <b>300</b> – <b>150</b> = <b>150</b> ) За седмия таб -> Facebook -> глоба <b>150</b> лв. (заплата: <b>150</b> – <b>150</b> = <b>0</b> ) Заплатата е равна на <b>0</b> => изписва съответният изход и програмата приключва	
Вход	Изход	Вход	Изход
3 500 Github.com Stackoverf low.com softuni.bg	500	3 500 Facebook Stackoverflow.com softuni.bg	350

## 6. Оскари

Поканени сте от академията да напишете софтуер, който да пресмята точките за актьор/актриса. Академията ще ви даде първоначални **точки за актьора**. След това **всеки оценяващ** ще дава **своята оценка**. Точките, които **актьора получава** се формират от: **дължината на името на оценяващия умножено по точките**, които **дава делено на две**.

Ако резултатът в някой момент надхвърли **1250.5** програмата трябва да прекъсне и да се отпечата, че **дадения актьор е получил номинация**.

### Вход

- Име на актьора - **текст**
- Точки от академията - **реално число в интервала [2.0... 450.5]**
- Брой оценяващи **n** - **цяло число в интервала [1... 20]**

На следващите **n-на брой реда**:

- Име на оценяващия - **текст**
- Точки от оценяващия - **реално число в интервала [1.0... 50.0]**

### Изход

Да се отпечата на конзолата един ред:

- Ако точките са над **1250.5**:  
"Congratulations, {име на актьора} got a nominee for leading role with {точки}!"
- Ако точките **не са достатъчни**:  
"Sorry, {име на актьора} you need {нужни точки} more!"

Резултатът да се форматирана до първата цифра след десетичния знак!

### Примерен вход и изход

Вход	Изход	Обяснения
Zahari Baharov <b>205</b> <b>4</b>	Sorry, Zahari Baharov you need 247.5 more!	Zahari Baharov започва с <b>205</b> точки, като <b>4</b> човека ще го оценяват. Първи е Johnny Depp

Johnny Depp 45 Will Smith 29 Jet Lee 10 Matthew Mcconaughey 39		$\Rightarrow 205 + ((11 * 45) / 2) = 452.5$ Втори е Will Smith $\Rightarrow 452.5 + ((10 * 29) / 2) = 597.5$ Трети е Jet Lee $\Rightarrow 597.5 + ((7 * 10) / 2) = 632.5$ Четвърти е Matthew Mcconaughey $\Rightarrow 632.5 + ((19 * 39) / 2) = 1003.0$ 1003.0 < 1250.5 $\Rightarrow$ Точките не са достатъчни. Нужни точки: 1250.5 - 1003.0 = 247.5
Sandra Bullock 340 5 Robert De Niro 50 Julia Roberts 40.5 Daniel Day-Lewis 39.4 Nicolas Cage 29.9 Stoyanka Mutafova 33	Congratulations, Sandra Bullock got a nominee for leading role with 1268.5!	

## 7. Трекинг мания

Катерачи от цяла България се събират на групи и набеязват следващите върхове за изкачване. Според **размера на групата**, катерачите ще изкачват **различни върхове**.

- Група **до 5 човека** – изкачват **Мусала**
- Група **от 6 до 12 човека** – изкачват **Монблан**
- Група **от 13 до 25 човека** – изкачват **Килиманджаро**
- Група **от 26 до 40 човека** – изкачват **K2**
- Група **от 41 или повече човека** – изкачват **Еверест**

Да се **напише програма**, която **изчислява процента** на катерачите изкачващи всеки връх.

### Вход

От конзолата се четат **поредица от числа**, всяко на отделен ред:

- На **първия ред** – **броя на групите** от катерачи – цяло число в интервала [1...1000]
- За **всяка една група** на отделен ред – **броя на хората в групата** – цяло число в интервала [1...1000]

### Изход

Да се отпечата на конзолата **5 реда**, всеки от които съдържа **процент между 0.00% и 100.00%** с точност до **втората цифра след десетичната запетая**.

- Първи ред** - процентът изкачващи **Мусала**
- Втори ред** – процентът изкачващи **Монблан**
- Трети ред** – процентът изкачващи **Килиманджаро**

- Четвърти ред – процентът изкачващи K2
- Пети ред – процентът изкачващи Еверест

## Примерен вход и изход

Вход	Изход	Обяснения
10	1.84%	Общ брой хора: $10 + 5 + 1 + 100 + 12 + 26 + 17 + 37 + 40 + 78 = 326$
10	6.75%	Изкачващи Мусала: $6 / 326 * 100 = 1.84\%$
5	5.21%	Изкачващи Монблан: $22 / 326 * 100 = 6.75\%$
1	31.60%	Изкачващи Килиманджаро: $17 / 326 * 100 = 5.21\%$
100	54.60%	Изкачващи K2: $103 / 326 * 100 = 31.60\%$
12		Изкачващи Еверест: $178 / 326 * 100 = 54.60\%$
26		
17		
37		
40		
78		
Вход	Изход	Обяснения
5	0.00%	Общ брой хора: $25 + 41 + 31 + 250 + 6 = 353$
25	1.70%	Изкачващи Мусала: $0 / 353 * 100 = 0.00\%$
41	7.08%	Изкачващи Монблан: $6 / 353 * 100 = 1.78\%$
31	8.78%	Изкачващи Килиманджаро: $25 / 353 * 100 = 7.08\%$
250	82.44%	Изкачващи K2: $31 / 353 * 100 = 8.78\%$
6		Изкачващи Еверест: $291 / 353 * 100 = 82.44\%$

## 8. Световна ранглиста по тенис

Григор Димитров е тенисист, чиято следваща цел е изкачването в световната ранглиста по тенис за мъже.

През годината Гришо участва в **определен брой турнири**, като за всеки турнир получава точки, които зависят от **позицията, на която е завършил в турнира**. Има **три варианта** за завършване на турнир:

- W - ако е победител получава 2000 точки
- F - ако е финалист получава 1200 точки
- SF - ако е полуфиналист получава 720 точки

Напишете програма, която **изчислява колко ще са точките на Григор след изиграване на всички турнири**, като знаете **с колко точки стартира сезона**. Също изчислете **колко точки средно печели** от всички изиграни турнири и **колко процента** от турнирите е спечелил.

### Вход

От конзолата първо се четат **два реда**:

- Брой турнири, в които е участвал – цяло число в интервала [1...20]
- Начален брой точки в ранглистата - цяло число в интервала [1...4000]

За всеки турнир се прочита отделен ред:

- Достигнат етап от турнира – текст – "W", "F" или "SF"

### Изход

Отпечатват се **три реда** в следния формат:

- "Final points: {брой точки след изиграните турнири}"
- "Average points: {средно колко точки печели за турнир}"
- "{процент спечелени турнири}%"

Средните точки да бъдат закръглени към най-близкото цяло число надолу, а процентът да се форматира до втората цифра след десетичния знак.



## Примерен вход и изход

Вход	Изход	Обяснения	
<b>5</b> <b>1400</b> <b>F</b> <b>SF</b> <b>W</b> <b>W</b> <b>SF</b>	Final points: 8040 Average points: 1328 40.00%	<p><b>5</b> турнира и начален брой точки: <b>1400</b></p> <p>1-ви турнир -&gt; финал (<b>F</b>) -&gt; точки = <b>1400</b> + <b>1200</b> = <b>2600</b></p> <p>2-ри турнир -&gt; полуфинал (<b>SF</b>) -&gt; точки = <b>2600</b> + <b>720</b> = <b>3320</b></p> <p>3-ти турнир -&gt; победител (<b>W</b>) -&gt; точки = <b>3320</b> + <b>2000</b> = <b>5320</b></p> <p>4-ти турнир -&gt; победител (<b>W</b>) -&gt; точки = <b>5320</b> + <b>2000</b> = <b>7320</b></p> <p>5-ти турнир -&gt; полуфинал (<b>SF</b>) -&gt; точки = <b>7320</b> + <b>720</b> = <b>8040</b></p> <p>Точки след изиграване на турнирите: <b>8040</b></p> <p>Средно спечелени точки за турнир:  <math>(1200 + 720 + 2000 + 2000 + 720) / 5 = 6640 / 5 = 1328</math></p> <p>Брой спечелени турнири: <b>2</b>            Процент спечелени турнири: <math>(2 / 5) * 100 = 40 \%</math></p>	
Вход	Изход	Вход	Изход
4 750 SF W SF W	Final points: 6190 Average points: 1360 50.00%	7 1200 SF F W F W SF W	Final points: 11040 Average points: 1405 42.86%