

Лаб: Условни конструкции

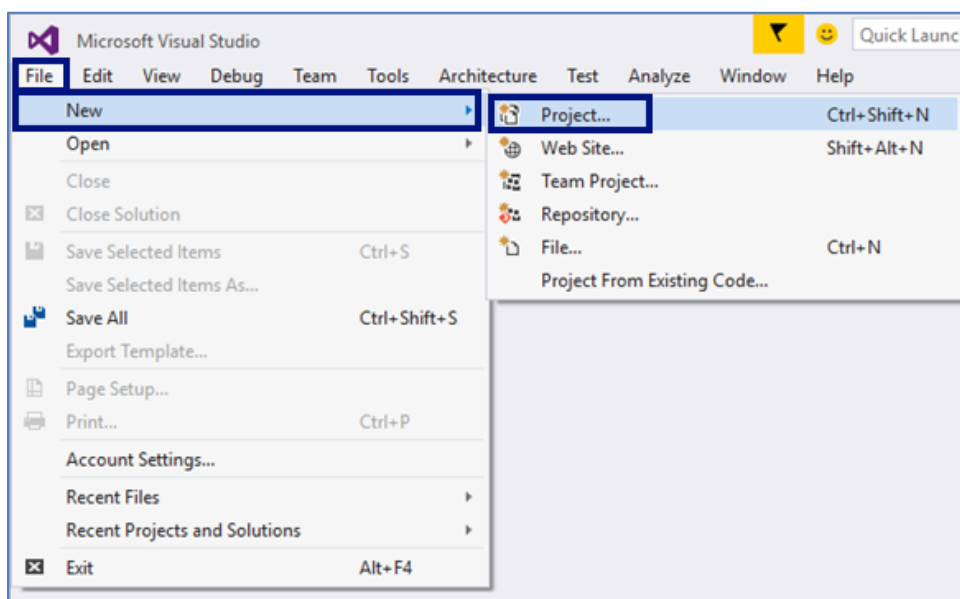
Задачи за упражнение в клас и за домашно към курса ["Основи на програмирането" @ СофтУни](#).

0. Празно Visual Studio решение (Blank Solution)

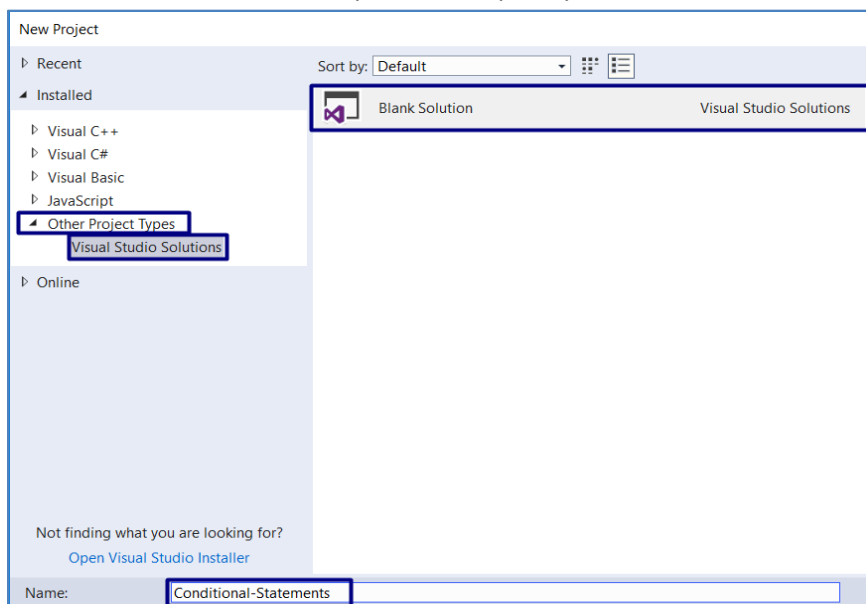
Създайте празно решение (**Blank Solution**) във Visual Studio. Решенията (solutions) във Visual Studio обединяват **група проекти**. Тази възможност е изключително удобна, когато искаме да работим по няколко проекта и бързо да превключваме между тях или искаме да обединим логически няколко взаимосвързани проекта.

В настоящото практическо занимание ще използваме **Blank Solution с няколко проекта** за да организираме решенията на задачите от упражненията – всяка задача в отделен проект и всички проекти в общ solution.

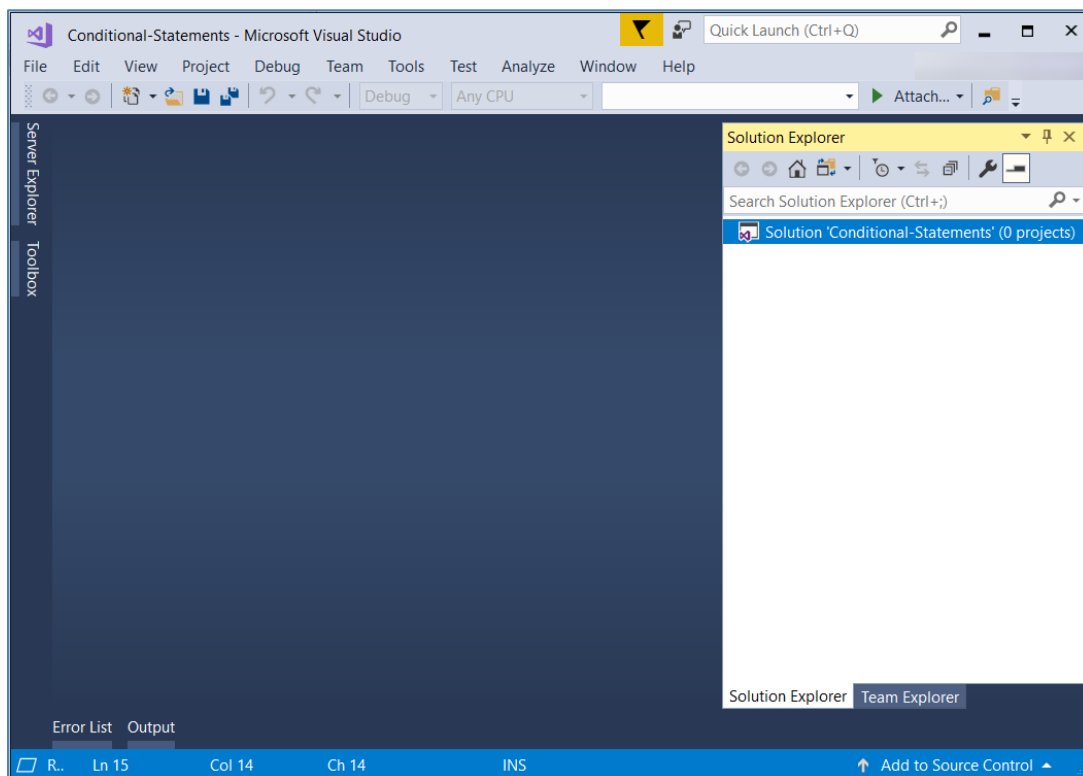
1. Стартирайте Visual Studio.
2. Създайте нов **Blank Solution**: [File]→ [New] → [Project].



3. Изберете от диалоговия прозорец [Other Project Types] → [Visual Studio Solutions] → [**Blank Solution**] и дайте подходящо име на проекта, например "Conditional-Statements":



Сега имате създаден **празен Visual Studio Solution** (с 0 проекта в него):



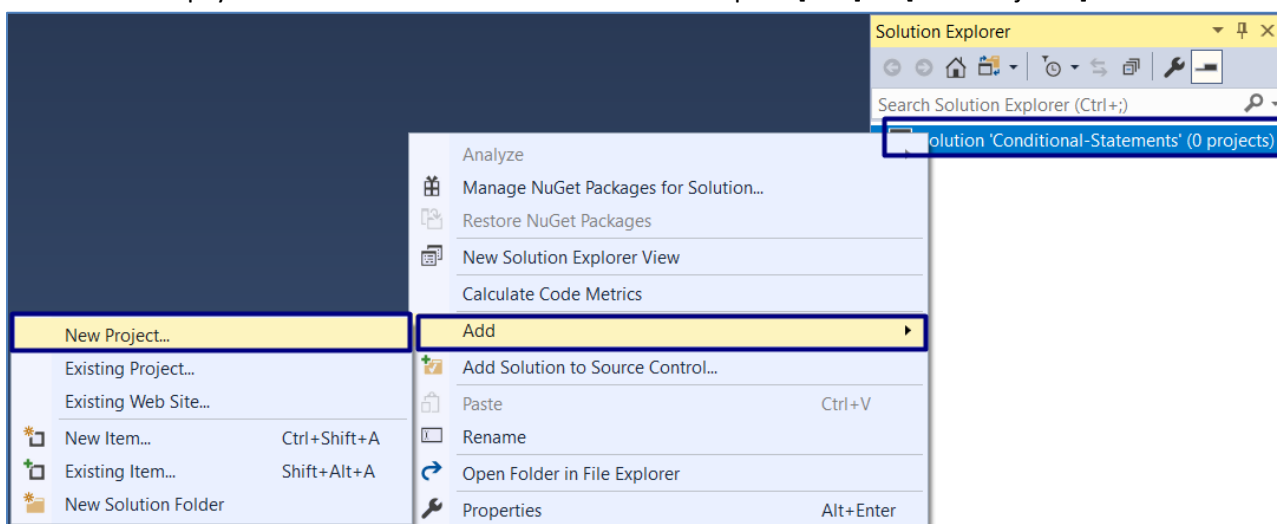
Целта на този blank solution е да добавяте в него **по един проект за всяка задача** от упражненията.

1. Проверка за отлична оценка

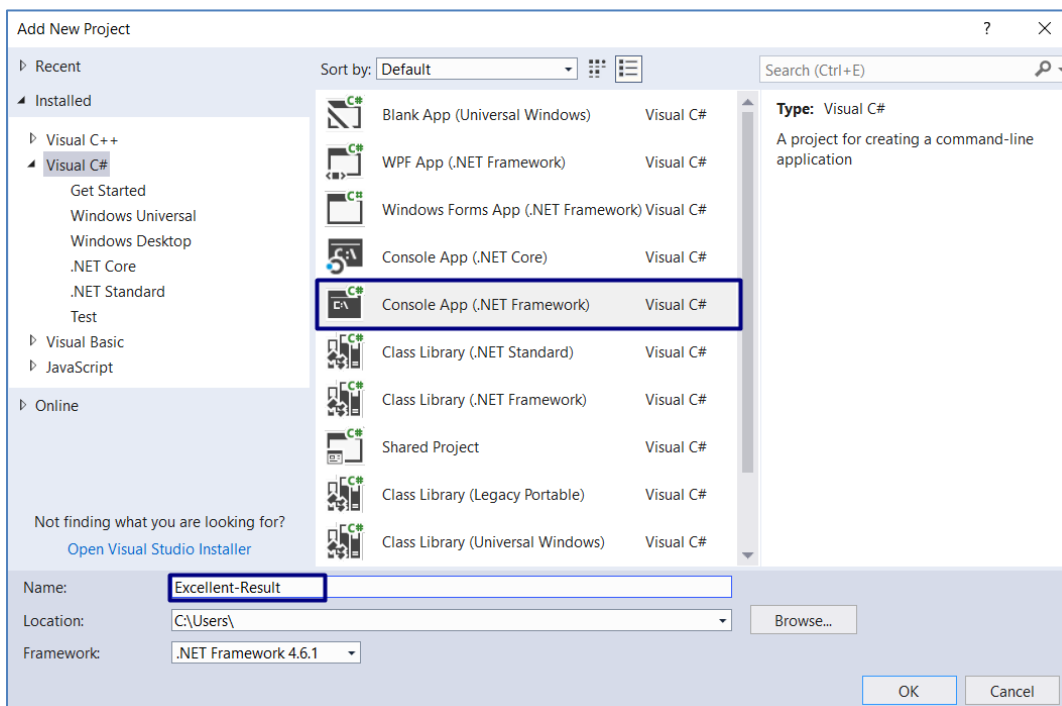
Първата задача от тази тема е да се напише **конзолна програма**, която **чете оценка** (десетично число), въведена от потребителя, и отпечатва **"Excellent!"** ако оценката е **5.50** или по-висока.

ВХОД	ИЗХОД	ВХОД	ИЗХОД	ВХОД	ИЗХОД	ВХОД	ИЗХОД
6	Excellent!	5	(НЯМА ИЗХОД)	5.50	Excellent!	5.49	(НЯМА ИЗХОД)

1. Създайте **нов проект** в съществуващото Visual Studio решение. В Solution Explorer кликнете с десен бутон на мишката върху **Solution "Conditional-Statements"**. Изберете [Add] → [New Project...]:



2. Ще се отвори диалогов прозорец за избор на тип проект за създаване. Изберете C# конзолно приложение с име "Excellent-Result":



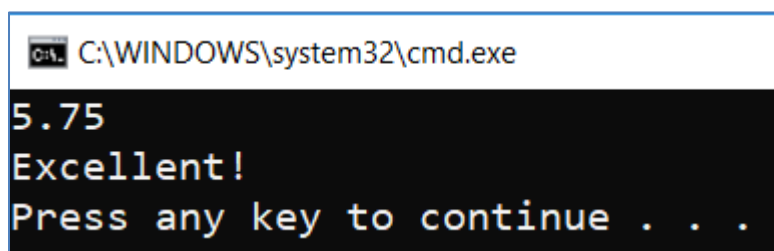
Вече имате solution с едно конзолно приложение в него. Остава да напишете кода за решаване на задачата.

3. Отидете в тялото на метода `Main(string[] args)` и напишете кода от картинката по-долу:

```
namespace Excellent_Result
{
    class Program
    {
        static void Main(string[] args)
        {
            double grade = double.Parse(Console.ReadLine());

            if(grade >= 5.50)
            {
                Console.WriteLine("Excellent!");
            }
        }
    }
}
```

Кодът въвежда реално число с `grade = double.Parse(Console.ReadLine())`, след това проверява дали е по-голямо или равно на 5.50 и ако е принтира "Excellent". **Стартирайте** програмата с [Ctrl+F5] и я **тествайте** с различни входни стойности:



```
C:\WINDOWS\system32\cmd.exe
4.80
Press any key to continue . . .
```

4. Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1012#0>

01. Excellent Result

```
1 using System;
2 namespace Excellent_Result
3 {
4     class Program
5     {
6         static void Main(string[] args)
7         {
8             double grade = double.Parse(Console.ReadLine());
9
10            if(grade >= 5.50)
11            {
12                Console.WriteLine("Excellent!");
13            }
14        }
15    }
16 }
```

Allowed working time: 0.200 sec.
Allowed memory: 16.00 MB
Size limit: 16.00 KB
Checker: Case-Insensitive

C# code Submit

5. Трябва да получите **100 точки** (напълно коректно решение):

Submissions			
<div>⏮ ⏪ 1 ⏩ ⏭</div>			
Points	Time and memory used	Submission date	
✓✓✓✓✓✓ 100 / 100	Memory: 7.42 MB Time: 0.046 s	20:27:32 10.09.2018	Details
<div>⏮ ⏪ 1 ⏩ ⏭</div>			

2. Четно или нечетно

Да се напише програма, която чете **цяло число**, въведено от потребителя, и печата дали е **четно** или **нечетно**.
Примери:

ВХОД	ИЗХОД
2	even

ВХОД	ИЗХОД
3	odd

ВХОД	ИЗХОД
25	odd

ВХОД	ИЗХОД
1024	even

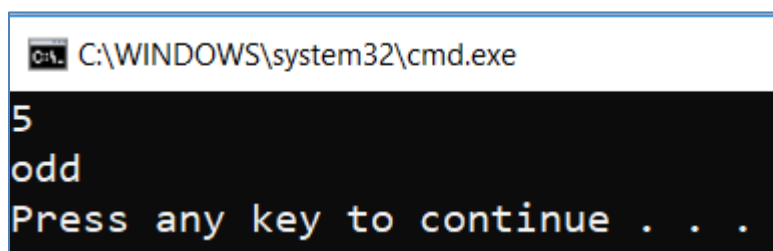
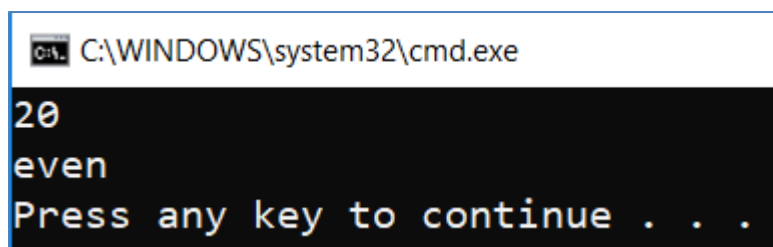
1. Първо създайте нов **C# конзолен проект** в решението "**Conditional-Statements**" с име "**Even-Or-Odd**".

2. **Напишете код** на програмата. Може да си помогнете с примерния код от картинката:

```
namespace Even_Or_Odd
{
    class Program
    {
        static void Main(string[] args)
        {
            int grade = int.Parse(Console.ReadLine());

            if (grade % 2 == 0)
            {
                Console.WriteLine("even");
            }
            else
            {
                Console.WriteLine("odd");
            }
        }
    }
}
```

3. **Стартирайте** програмата с [Ctrl+Shift+F10] и я тествайте с различни стойности:



4. Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1012#1>

3. Намиране на по-голямото число

Да се напише програма, която чете **две цели числа**, въведени от потребителя, и отпечатва по-голямото от двете. Примери:

вход	изход	вход	изход	вход	изход	вход	изход
5	5	3	5	10	10	-5	5
3		5		10		5	

- Първо създайте **нов C# конзолен проект** в решението "**Conditional-Statements**".
- Напишете код** на програмата. Необходима е единична **if-else** конструкция.

3. **Стартирайте** програмата с **[Ctrl+Shift+F10]** и я тествайте:

```
C:\WINDOWS\system32\cmd.exe
5
3
5
Press any key to continue . . .
```

```
C:\WINDOWS\system32\cmd.exe
10
10
10
Press any key to continue . . .
```

4. Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1012#2>

Подсказка: може да си помогнете частично с кода от картинката, който е нарочно замъглен, за да помислите как да го напишете сами:

```
namespace Bigger_Integer
{
    class Program
    {
        static void Main(string[] args)
        {
            int num1 = int.Parse(Console.ReadLine());
            int num2 = int.Parse(Console.ReadLine());

            // TODO: Write your code here
            // TODO: Write your code here
            // TODO: Write your code here
        }
    }
}
```

4. Изписване на число до 9 с думи

Да се напише програма, която чете **цяло число в диапазона [0...10]**, въведено от потребителя, и го **изписва с думи** на английски език. Ако числото е извън диапазона, изписва **"number too big"**. Примери:

вход	изход	вход	изход	вход	изход	вход	изход
5	five	1	one	9	nine	10	number too big

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1012#3>

Подсказка: можете да напишете дълга **if-else-if-else...else**, с която да разгледате възможните **11** случая.

5. Познай паролата

Да се напише програма, която **чете парола** (един ред с произволен текст), въведена от потребителя, и проверява дали въведеното **съвпада** с фразата **"s3cr3t!P@ssw0rd"**. При съвпадение да се изведе **"Welcome"**. При несъвпадение да се изведе **"Wrong password!"**. Примери:

вход	изход	вход	изход	вход	изход
qwerty	Wrong password!	s3cr3t!P@ssw0rd	Welcome	s3cr3t!p@ss	Wrong password!

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1012#4>

Подсказка: използвайте **if-else** конструкцията.

6. Число от 100 до 200

Да се напише програма, която **чете цяло число**, въведено от потребителя, и проверява дали е **под 100**, **между 100 и 200** или **над 200**. Да се отпечатаат съответно съобщения като в примерите по-долу:

вход	изход	вход	изход	вход	изход
95	Less than 100	120	Between 100 and 200	210	Greater than 200

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1012#5>

Подсказка: използвайте **if-else-if-else** конструкция за да проверите всеки от трите случая.

7. Еднакви думи

Да се напише програма, която **чете две думи**, въведени от потребителя, и проверява дали са еднакви. Да не се прави разлика между главни и малки думи. Да се изведе **"yes"** или **"no"**. Примери:

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
Hello Hello	yes	SoftUni softuni	yes	Soft Uni	no	beer vodka	no	HeLLo hELLo	yes

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1012#6>

Подсказка: използвайте **if-else** конструкция. Преди сравняване на думите ги обърнете в долен регистър: **word = word.toLowerCase()**.

8. Лица на фигури

Да се напише програма, в която потребителят **въвежда вида и размерите на геометрична фигура** и пресмята лицето ѝ. Фигурите са четири вида: квадрат (**square**), правоъгълник (**rectangle**), кръг (**circle**) и триъгълник (**triangle**). На първия ред на входа се чете вида на фигурата (**square**, **rectangle**, **circle** или **triangle**). Ако фигурата е **квадрат**, на следващия ред се чете едно число - дължина на страната му. Ако фигурата е **правоъгълник**, на следващите два реда четат две числа - дължините на страните му. Ако фигурата е **кръг**, на следващия ред чете едно число - радиусът на кръга. Ако фигурата е **триъгълник**, на следващите два реда четат две числа - дължината на страната му и дължината на височината към нея. Резултатът да се закръгли до **3 цифри след десетичната точка**. Примери:

вход	изход	вход	изход	вход	изход	вход	изход
------	-------	------	-------	------	-------	------	-------



square 5	25	rectangle 7 2.5	17.5	circle 6	113.097	triangle 4.5 20	45
-------------	----	-----------------------	------	-------------	---------	-----------------------	----

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1012#7>

Подсказка: използвайте серия от **if-else-if-else-...** конструкции, за да обработите 4-те вида фигури.

9. Еднакви 3 числа

Да се въведат 3 числа и да се отпечата дали са еднакви (yes / no)

Примери:

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
1 1 1	yes	5 5 5	yes	1 2 3	no	11 8 5	no	13 14 99	no

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1012#8>

Примерна изпитна задача

10. Магазин за детски играчки

Тествайте решението си [тук](#).

Петя има магазин за детски играчки. Тя получава голяма поръчка, която трябва да изпълни. С парите, които ще спечели иска да отиде на екскурзия. Да се напише програма, която пресмята печалбата от поръчката.

Цени на играчките:

- Пъзел - 2.60 лв.
- Говореща кукла - 3 лв.
- Плюшено мече - 4.10 лв.
- Миньон - 8.20 лв.
- Камионче - 2 лв.

Ако поръчаните играчки са **50 или повече** магазинът прави **отстъпка 25% от общата цена**. От спечелените пари Петя трябва да даде **10% за наема** на магазина. Да се пресметне дали парите ще ѝ стигнат да отиде на екскурзия.

Вход

От конзолата се четат **6 реда**:

1. Цена на екскурзията - реално число в интервала [1.00 ... 10000.00]
2. Брой пъзели - цяло число в интервала [0... 1000]
3. Брой говорещи кукли - цяло число в интервала [0 ... 1000]
4. Брой плюшени мечета - цяло число в интервала [0 ... 1000]
5. Брой миньони - цяло число в интервала [0 ... 1000]



6. Брой камиончета - цяло число в интервала [0 ... 1000]

Изход

На конзолата се отпечатва:

- Ако парите са достатъчни се отпечатва:
 - "Yes! {оставащите пари} lv left."
- Ако парите НЕ са достатъчни се отпечатва:
 - "Not enough money! {недостигащите пари} lv needed."

Резултатът трябва да се форматира до втория знак след десетичната запетая.

Примерен вход и изход

Вход	Изход	Обяснения
40.8 20 25 30 50 10	Yes! 418.20 lv left.	Сума: $20 * 2.60 + 25 * 3 + 30 * 4.10 + 50 * 8.20 + 10 * 2 = 680$ лв. Брой на играчките: $20 + 25 + 30 + 50 + 10 = 135$ $135 > 50 \Rightarrow 25\%$ отстъпка; 25% от $680 = 170$ лв. отстъпка Крайна цена: $680 - 170 = 510$ лв. Наем: 10% от 510 лв. = 51 лв. Печалба: $510 - 51 = 459$ лв. $459 > 40.8 \Rightarrow 459 - 40.8 = 418.20$ лв. остават
Вход	Изход	Обяснения
320 8 2 5 5 1	Not enough money! 238.73 lv needed.	Сума: 90.3 лв. Брой на играчките: 21 $21 < 50 \Rightarrow$ няма отстъпка Наем: 10% от $90.3 = 9.03$ лв. Печалба: $90.3 - 9.03 = 81.27$ лв. $81.27 < 320 \Rightarrow 320 - 81.27 = 238.73$ лв. не достигат