

Упражнения: По-сложни проверки

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 за да организирате решенията на задачите от упражненията. Всяка задача ще бъде в отделен проект и всички проекти ще бъдат в общ solution.

1. Стартирайте **Visual Studio**.
2. Създайте нов **Blank Solution**: [File] → [New] → [Project].
3. Изберете от диалоговия прозорец [Templates] → [Other Project Types] → [Visual Studio Solutions] → [**Blank Solution**] и дайте подходящо име на проекта, например “**Complex-Conditions**”:

Сега имате създаден **празен Visual Studio Solution** (без проекти в него):

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

1. Обръщение според възраст и пол

Първата задача от тази тема е да се напише **конзолна програма**, която **прочита възраст** (десетично число) и **пол** (“m” или “f”), въведени от потребителя, и отпечатва **обръщение** измежду следните:

- “Mr.” – мъж (пол “m”) на 16 или повече години
- “Master” – момче (пол “m”) под 16 години
- “Ms.” – жена (пол “f”) на 16 или повече години
- “Miss” – момиче (пол “f”) под 16 години

Примери:

ВХОД	ИЗХОД	ВХОД	ИЗХОД	ВХОД	ИЗХОД	ВХОД	ИЗХОД
12 f	Miss	17 m	Mr.	25 f	Ms.	13.5 m	Master

1. Създайте **нов проект** в съществуващото Visual Studio решение. В Solution Explorer кликнете с десен бутон на мишката върху **Solution** реда и изберете [Add] → [New Project...]:
2. Ще се отвори диалогов прозорец за избор на тип проект за създаване. Изберете **C# конзолно приложение** и задайте подходящо име, например “**Personal-Titles**”:

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 решението на задачата. Можете да си помогнете с кода от картинката по-долу:

4. **Стартирайте** програмата с [Ctrl+F5] и я **тествайте** с различни входни стойности:

5. **Тествайте** решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/153#0>. Трябва да получите **100 точки** (напълно коректно решение):

2. Квартално магазинче

Следващата задача има за цел да тренира работата с **вложени проверки** (nested **if**). Ето го и условието: предприемчив българин отваря **квартални магазинчета** в **няколко града** и продава на **различни цени**:

град / продукт	coffee	water	beer	sweets	peanuts
Sofia	0.50	0.80	1.20	1.45	1.60
Plovdiv	0.40	0.70	1.15	1.30	1.50
Varna	0.45	0.70	1.10	1.35	1.55

Напишете програма, която чете **град** (стринг), **продукт** (стринг) и **количество** (десетично число), въведени от потребителя, и пресмята и отпечатва **колко струва** съответното количество от избрания продукт в посочения град. Примери:

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
coffee Varna 2	0.9	peanuts Plovdiv 1	1.5	beer Sofia 6	7.2	water Plovdiv 3	2.1	sweets Sofia 2.23	3.2335

1. Създайте **нов проект** в съществуващото Visual Studio решение. В Solution Explorer кликнете с десен бутон на мишката върху **Solution** реда и изберете [Add] → [New Project...]:

2. Ще се отвори диалогов прозорец за избор на тип проект за създаване. Изберете **C# конзолно приложение** и задайте подходящо име, например **"Small-Shop"**:

Вече имате ново конзолно приложение и остава да напишете кода за решаване на задачата.

3. Отидете в тялото на метода **Main(string[] args)** и напишете решението на задачата. Можете да си помогнете с кода от картинката по-долу. Можете да прехвърлите всички букви в долен регистър с **.ToLower()** за да сравнявате продукти и градове без значение на малки / главни букви:

4. За да **активирате текущия проект** да стартира при [Ctrl+F5], изберете **"Set StartUp Projects..."**:

5. **Стартирайте** програмата с [Ctrl+F5] и я **тествайте** с различни входни стойности:

6. **Тествайте** решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/153#1>.

3. Точка в правоъгълник

Напишете програма, която проверява дали **точка** $\{x, y\}$ се намира **вътре в правоъгълник** $\{x_1, y_1\} - \{x_2, y_2\}$. Входните данни се четат от конзолата и се състоят от 6 реда, въведени от потребителя: десетичните числа x_1, y_1, x_2, y_2, x и y (като се гарантира, че $x_1 < x_2$ и $y_1 < y_2$). Една точка е вътрешна за даден правоъгълник, ако се намира някъде във вътрешността му или върху някоя от страните му. Отпечатайте **"Inside"** или **"Outside"**. Примери:

вход	изход	визуализация
2 -3 12 3 8 -1	Inside	

вход	изход	визуализация
2 -3 12 3 11 -3.5	Outside	

вход	изход	визуализация
-1 -3 4 1 0.5 1	Inside	

вход	изход	визуализация
-1 -3 4 1 -1.2 1.4	Outside	

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/153#2>.

* **Подсказка**: една точка е вътрешна за даден многоъгълник, ако едновременно са изпълнени следните четири условия (можете да ги проверите с **if** проверка с логическо „и“ – оператор **&&**):

- Точката е надясно от лявата стена на правоъгълника ($x \geq x_1$)
- Точката е наляво от дясната стена на правоъгълника ($x \leq x_2$)
- Точката е надолу от горната стена на правоъгълника ($y \geq y_1$)
- Точката е нагоре от долната стена на правоъгълника ($y \leq y_2$)

4. Плод или зеленчук?

Да се напише програма, която **чете име на продукт**, въведено от потребителя, и проверява дали е **плод** или **зеленчук**.

- Плодовете **"fruit"** са **banana, apple, kiwi, cherry, lemon** и **grapes**
- Зеленчуците **"vegetable"** са **tomato, cucumber, pepper** и **carrot**
- Всички останали са **"unknown"**

Да се изведе **"fruit"**, **"vegetable"** или **"unknown"** според въведения продукт. Примери:

вход	изход
banana	fruit

вход	изход
apple	fruit

вход	изход
tomato	vegetable

вход	изход
water	unknown

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Practice/Index/153#3>.

* Подсказка: използвайте условна `if` проверка с логическо „или“ – operator `||`.

5. Невалидно число

Дадено число е валидно, ако е в диапазона `[100...200]` или е `0`. Да се напише програма, която чете цяло число, въведено от потребителя, и печата `invalid` ако въведеното число не е валидно. Примери:

вход	изход
75	invalid

вход	изход
150	(няма изход)

вход	изход
220	invalid

вход	изход
199	(няма изход)

вход	изход
-1	invalid

вход	изход
100	(няма изход)

вход	изход
200	(няма изход)

вход	изход
0	(няма изход)

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Practice/Index/153#4>.

* Подсказка: използвайте условна `if` проверка с отрицание и логически операции.

6. Точка върху страната на правоъгълник

Напишете програма, която проверява дали точка `{x, y}` се намира върху някоя от страните на правоъгълник `{x1, y1} – {x2, y2}`. Входните данни се четат от конзолата и се състоят от 6 реда въведени от потребителя: десетичните числа `x1, y1, x2, y2, x` и `y` (като се гарантира, че `x1 < x2` и `y1 < y2`). Да се отпечата `Border` (точката лежи на някоя от страните) или `Inside / Outside` (в противен случай). Примери:

вход	изход	визуализация
2 -3 12 3 8 -1	Inside / Outside	

вход	изход	визуализация
2 -3 12 3 12 -1	Border	

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Practice/Index/153#5>.

* Подсказка: използвайте една или няколко условни `if` проверки с логически операции. Точка `{x, y}` лежи върху някоя от страните на правоъгълник `{x1, y1} – {x2, y2}`, ако е изпълнено едно от следните условия:

- `x` съвпада с `x1` или `x2` и същевременно `y` е между `y1` и `y2`
- `y` съвпада с `y1` или `y2` и същевременно `x` е между `x1` и `x2`

Можете да проверите горните условия с една по-сложна **if-else** конструкция или с няколко по-прости проверки или с **вложени if-else** проверки.

7. Магазин за плодове

Магазин за плодове през **работните дни** работи на следните **цени**:

плод	banana	apple	orange	grapefruit	kiwi	pineapple	grapes
цена	2.50	1.20	0.85	1.45	2.70	5.50	3.85

Събота и неделя магазинът работи на **по-високи цени**:

плод	banana	apple	orange	grapefruit	kiwi	pineapple	grapes
цена	2.70	1.25	0.90	1.60	3.00	5.60	4.20

Напишете програма, която чете от конзолата **плод** (banana / apple / orange / grapefruit / kiwi / pineapple / grapes), **ден от седмицата** (Monday / Tuesday / Wednesday / Thursday / Friday / Saturday / Sunday) и **количество** (десетично число) , въведени от потребителя, и пресмята **цената** според цените от таблиците по-горе.

Резултатът да се отпечата **закръглен с 2 цифри** след десетичната точка. При невалиден ден от седмицата или невалидно име на плод да се отпечата **"error"**. Примери:

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
apple Tuesday 2	2.40	orange Sunday 3	2.70	kiwi Monday 2.5	6.75	grapes Saturday 0.5	2.10	tomato Monday 0.5	error

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/153#6>.

* **Подсказки:**

- Прочетете входа и обърнете името на плода и деня от седмицата в **малки букви**:
- Първоначално задайте цена **-1**:
- Използвайте вложени **if** проверки, за да изчислите цената за дадения плод и ден от седмицата:
- Накрая проверете цената. Ако все още е **-1**, значи даденият плод или денят от седмицата е **невалиден**. За да отпечатате точно **2 цифри след десетичната точка** (със закръгляне), използвайте форматиращ низ **"{0:f2}"**. Кодът може да е подобен на следния:

8. Търговски комисионни

Фирма дава следните **комисионни** на търговците си според **града**, в който работят и обема на **продажбите s**:

Град	$0 \leq s \leq 500$	$500 < s \leq 1\,000$	$1\,000 < s \leq 10\,000$	$s > 10\,000$
Sofia	5%	7%	8%	12%
Varna	4.5%	7.5%	10%	13%
Plovdiv	5.5%	8%	12%	14.5%

Напишете **конзолна програма**, която чете име на **град** (string) и обем на **продажби** (десетично число) , въведени от потребителя, и изчислява и извежда размера на търговската **комисионна** според горната таблица. Резултатът да се изведе **закръглен с 2 цифри след десетичната точка**. При **невалиден** град или обем на продажбите (отрицателно число) да се отпечата **"error"**. Примери:

ВХОД	ИЗХОД
Sofia 1500	120.00

ВХОД	ИЗХОД
Plovdiv 499.99	27.50

ВХОД	ИЗХОД
Varna 3874.50	387.45

ВХОД	ИЗХОД
Kaspichan -50	error

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Practice/Index/153#7>.

* Подсказки:

- Прочетете входа и **обърнете града в малки букви** (като в предходната задача).
- Първоначално задайте **комисионна -1**. Тя ще бъде променена, ако градът и ценовият диапазон бъдат намерени в таблицата с комисионните.
- Използвайте вложени **if** проверки, за **да изчислите комисионната** според града и според обема на продажбите. Може да си помогнете с кода по-долу:
- Накрая проверете комисионната. Ако все още е **-1**, значи въведеният град или обем продажби не се срещат в таблицата с комисионните и трябва да се отпечата **"error"**. В противен случай трябва да се изчисли комисионната (процент комисионна по обем на продажбите) и да се отпечата със закръгляне с точно **2 цифри след десетичната точка**. Може да използвате **Console.WriteLine("{0:f2}", ...)**.

9. Ден от седмицата

Напишете програма, която чете **цяло число**, въведено от потребителя, и отпечата **ден от седмицата** (на английски език), в граници [1...7] или отпечата **"Error"** в случай, че въведеното число е **невалидно**. Примери:

Вход	Изход
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday
7	Sunday
-1	Error

*Подсказка: Използвайте условната конструкция [switch-case](#).

10. Клас животно

Напишете програма, която отпечата класа на животното според неговото име, въведено от потребителя.

- **dog -> mammal**
- **crocodile, tortoise, snake -> reptile**
- **others -> unknown**

Примери:

Вход	Изход
dog	mammal
snake	reptile
cat	unknown

*Подсказка: Използвайте условната конструкция **switch-case**.

11. Кино

В една кинозала столовете са наредени в правоъгълна форма в **r** реда и **c** колони. Има три вида прожекции с билети на различни цени:

- **Premiere** – премиерна прожекция, на цена **12.00** лева.
- **Normal** – стандартна прожекция, на цена **7.50** лева.
- **Discount** – прожекция за деца, ученици и студенти на намалена цена от **5.00** лева.

Напишете програма, която чете **тип прожекция** (стринг), брой **редове** и брой **колони** в залата (цели числа), въведени от потребителя, и изчислява общите приходи от билети при пълна зала. Резултатът да се отпечата във формат като в примерите по-долу, с 2 знака след десетичната точка. Примери:

вход	изход	вход	изход	вход	изход
Premiere 10 12	1440.00 leva	Normal 21 13	2047.50 leva	Discount 12 30	1800.00 leva

Тествайте решението си в **judge системата**: <https://judge.softuni.bg/Contests/Practice/Index/153#8>.

* **Подсказка**: използвайте прости проверки и елементарни изчисления. За да изведете резултата с точно 2 цифри след десетичната точка, използвайте **Console.WriteLine("{0:f2}", result)**.

12. Волейбол

Влади е студент, живее в София и си ходи от време на време до родния град. Той е много запален по волейбола, но е зает през работните дни и играе **волейбол** само през **уикендите** и в **празничните дни**. Влади играе в **София** всяка **събота**, когато **не е на работа** и **не си пътува до родния град**, както и в **2/3 от празничните дни**. Той пътува до **родния си град h пъти** в годината, където играе волейбол със старите си приятели в **неделя**. Влади **не е на работа 3/4 от уикендите**, в които е в София. Отделно, през **високосните години** Влади играе с **15% повече** волейбол от нормалното. Приемаме, че годината има точно **48 уикенда**, подходящи за волейбол.

Напишете програма, която изчислява **колко пъти Влади е играл волейбол** през годината. **Закръглете резултата надолу до най-близкото цяло число (например 2.15 → 2; 9.95 → 9).**

Входните данни се въвеждат от потребителя, в следния вид:

- Първият ред съдържа думата **"leap"** (високосна година) или **"normal"** (невисокосна).
- Вторият ред съдържа цялото число **p** – брой празници в годината (които не са събота и неделя).
- Третият ред съдържа цялото число **h** – брой уикенди, в които Влади си пътува до родния град.

Примери:

вход	изход	Коментари
leap 5 2	45	<p>48 уикенда в годината, разделени по следния начин:</p> <ul style="list-style-type: none"> 46 уикенда в София $\rightarrow 46 * 3 / 4 \rightarrow 34.5$ съботни игри в София 2 уикенда в родния си град $\rightarrow 2$ недели $\rightarrow 2$ игри в неделя в родния град <p>5 празника:</p> <ul style="list-style-type: none"> $5 * 2/3 \rightarrow 3.333$ игри в София в празничен ден <p>Общо игри през уикенди и празници в София и в родния град: $34.5 + 2 + 3.333 \rightarrow 39.833$</p> <p>Годината е високосна:</p> <ul style="list-style-type: none"> Влади играе допълнителни $15\% * 39.833 \rightarrow 5.975$ игри волейбол <p>Общо игри през цялата година:</p> <ul style="list-style-type: none"> $39.833 + 5.975 = 45.808$ игри Резултатът е 45 (закръгля се надолу)

вход	изход
normal 3 2	38

вход	изход
leap 2 3	43

вход	изход
normal 11 6	44

вход	изход
leap 0 1	41

вход	изход
normal 6 13	43

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Practice/Index/153#9>.

* Подсказки:

- Пресметнете **уикендите в София** (48 минус уикендите в родния град). Пресметнете **броя игри в уикендите в София**: умножете уикендите в София с $(3.0 / 4)$. Обърнете внимание, че трябва да се използва **дробно деление** $(3.0 / 4)$, а не целочислено $(3 / 4)$.
- Пресметнете **броя игри в родния град**. Те са точно колкото са пътуванията до родния град.
- Пресметнете **броя игри в празничен ден**. Те са броя празници умножени по $(2.0 / 3)$.
- Сумирайте** броя на всички игри. Той е дробно число. Не бързайте да закръгляте още.
- Ако годината е **високосна**, добавете **15%** към общия брой игри.
- Накрая **закръглете** надолу до най-близкото цяло число с **Math.Truncate(result)**.

13. * Точка във фигурата

Фигура се състои от **6 блокчета с размер $h * h$** , разположени като на фигурата вдясно. Долният ляв ъгъл на сградата е на позиция $\{0, 0\}$. Горният десен ъгъл на фигурата е на позиция $\{2*h, 4*h\}$. На фигурата координатите са дадени при **$h = 2$** .

Напишете програма, която чете цяло число **h** и координатите на дадена **точка $\{x, y\}$** (цели числа), въведени от потребителя, и отпечатва дали точката е вътре във фигурата (**inside**), вън от фигурата (**outside**) или на някоя от стените на фигурата (**border**).

Примери:

вход	изход	визуализация
2 3	outside	<pre> 0 2 4 6 1 1 1 1 </pre>

вход	изход	визуализация
15 13	outside	<pre> 0 15 30 45 7 7 7 7 </pre>

10			55		
2			15		
3	inside		29	inside	
1			37		
2			15		
2	border		37	outside	
2			18		
2			15		
6	border		-4	outside	
0			7		
2			15		
0	outside		30	border	
6			0		

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Practice/Index/153#10>.

* Подсказки:

- Може да разделите фигурата на **два правоъгълника** с обща стена.
- Една точка е **външна (outside)** за фигурата, когато е едновременно **извън** двата правоъгълника.
- Една точка е **вътрешна (inside)** за фигурата, ако е вътре в някой от правоъгълниците (изключвайки стените им) или лежи върху общата им стена.
- В **противен случай** точката лежи на стената на правоъгълника (**border**).