

# 1. Minecraft Adventure

С приятелите си в **Minecraft** искате да построите огромна мина под земята. Но за да не използвате **диамантите** и **желязото** си за кирки, решавате да използвате по-неиздръжливите, но евтини, **каменни кирки** */stone pickaxes/*.

Но ще ви трябват доста, защото те лесно се чупят. Вашата програма трябва да калкулира **колко каменни кирки** могат да бъдат създадени **преди да свършат материалите**, с които разполагате.

**Рецепта за изработка** */crafting recipe/* (за една каменна кирка):

- Cobblestone (x3) */калдъръм/*
- Stick (x2) */пръчка/*

**Не се броят материали, различни от:**

- "Cobblestone" */калдъръм/*
- "Sticks" */пръчка/*
- "Wood" */дърво/ (!)* от 1 дърво могат да бъдат направени 4 пръчки!

**Изведете в конзолата максималния брой каменни кирки, които можете да изработите, преди да ви свършат пръчките и/или калдъръма.** */sticks & cobblestones/*

Input	Logic	Output
Sticks Sticks Cobblestone Cobblestone Cobblestone END	Sticks x2 + Cobblestone x3 = 1 Stone pickaxes	Amount of stone pickaxes: 1
Wood Cobblestone Cobblestone Cobblestone END		Amount of stone pickaxes: 1
Sticks Wool Cobblestone END		Amount of stone pickaxes: 0

## 2. Dragon Numbers

Едно число е драконово, когато всички от следните условия са изпълнени:

- положително число
- цяло число
- по-голямо или равно на 100

Когато бъде пренаредено във всичките му пермутации и всяка пермутация\* бъде разделена на две части, поне едно от произведенията на двете половини трябва да е равно на подаденото число.

*\*Произволна редица от  $n$  на брой елементи без повторение, в която всеки от тези елементи се среща само веднъж.*

- ➔ Ако числото има четен брой цифри, лявата и дясната част ще имат еднаква дължина във всяка пермутация.
- ➔ Ако числото има нечетен брой цифри и поне три цифри, лявата и дясната част ще имат различна дължина за всяка възможна пермутация, редувайки ги в диапазона +1 и -1. (Пример: Ако числото има три цифри, в една пермутация едната половина ще съдържа 2 цифри, а другата – 1, а в друга пермутация – в двете половини може да има по 2 цифри (Виж пример 2 в таблицата)).

При подадено позитивно число, имплементирайте функция, която връща какъв тип е числото (като текст):

- 'Normal Number' - ако числото е по-малко от 100.
- 'Pseudodragon' - ако числото не е Драконово, заради нечетен брой на цифрите, от които се състои.
- 'True Dragon' - ако числото е Драконово.

Input	Logic	Output
1260	(!) Има четен брой цифри $(1260) = 12 * 60 = 720$ $(1620) = 16 * 20 = 320$ $(1026) = 10 * 26 = 260$ $(2160) = 21 * 60 = 1260$	True Dragon
126	(!) Има нечетен брой цифри $(126) = 12 * 6 = 72$ $(126) = 1 * 26 = 26$ $(1026) = 10 * 26 = 260$ $(216) = 21 * 6 = 126$	Pseudodragon
67	(!) По-малко от 100 $6 * 7 = 7 * 6 = 42$	Normal Number



### 3. Star Wars

Имате задача да **ремонтирате флота на Бунтовниците** след поражение в резултат на неочаквано нападение от Императора!

За да помогнете на Бунтовниците, вие първо трябва да **създадете чертежи за бойните кораби** за предстоящата битка.

За да спестите време и ресурси, **трябва да създадете тези чертежи с помощта на класове, спазвайки ООП принципите.**

*Използвайте подходящи типове за полетата. Всички числови полета са целочислени /integers/.*

#### Battleship

Всеки боен кораб трябва да има:

- **name** /име/
- **alliance** (Rebels / Empire) /съюз – Бунтовници/Империя /
- **shield capacity** /капацитет на щита/
- **blaster damage** /сила на бластера /
- **speed** /скорост/
- **light speed module** (Yes / No) /модул за светлинна скорост/

И функции за:

- **travel** (to destination)
- **fight** (another spaceship)

Имаме нужда от чертежи за различни типове космически кораби, всеки със своите допълнителни полета */properties/*:

#### XWing

Допълнителни полета:

- **maneuverability** /маневреност/

Функции:

- **evade**: /избегни/
  - Приема: друг XWing кораб (като противник)
  - Връща: true/false (в зависимост от това дали маневреността е по-висока от тази на противника)
- **travel**:
  - Приема: дестинация
  - Принтира:  
"XWing {name} is en route to {destination}"

Ако XWing корабът има вграден модул за светлинна скорост, тогава да се принтира:

"XWing {name} enters lightspeed hyperspace en route to {destination}"

- **fight**:
  - Приема: друг кораб
  - Логика:
    1. Проверка, дали корабът е от противниковия съюз */alliance/*. Ако не е, функцията не трябва да прави нищо.

2. Проверка, дали корабът е от тип **Star Destroyer** /звезда на смъртта/. Не можем да повредим Star Destroyer, затова функцията не трябва да прави нищо.
3. Проверка, дали корабът има **сила на бластера** /*blaster damage*/ по-голяма от **капацитета на щита** /*shield capacity*/ на противниковия кораб.
  - a. **Ако е така**, нека принтира:  
"Victory!"
  - b. **Ако не е така**:  
Проверка, дали **силата на бластера на противниковия кораб** е по-голяма от **капацитета на щита** ни.  
**Ако е така**, нека принтира:  
"Defeat!"
4. Накрая, ако **нито** нашият кораб, **нито** противниковият може да проникне през защитата на другия, проверяваме дали можем да избегнем битката /*evade*/.
  - a. **Ако можем**, нека принтира:  
"Battle Evaded!"
  - b. **Ако не можем**, нека принтира:  
"Defeat!"

## Star Destroyer

Полета:

- **crew capacity** /капацитет за екипаж/
- **amount of guns** /количество оръжия/

Функции:

- **deploy troops:**
  - Приема: локация
  - Принтира:  
"*{name} deploys troops to {location}*"

Ако Star Destroyer корабът има вграден модул за светлинна скорост, тогава да се принтира:

"*Star Destroyer {name} enters lightspeed hyperspace en route to {destination}*"

- **fight:**
  - Приема: друг кораб
  - Логика:
    1. Проверка, дали корабът е от противниковия съюз /*alliance*/. Ако не е, функцията не трябва да прави нищо.
    2. Проверка, дали корабът е от тип **XWing**. Не можем да атакуваме толкова малка мишена, затова функцията не трябва да прави нищо.
    3. Проверка, дали **общата сила на кораба** е по-голяма от **капацитета на щита** на противниковия кораб.  
Общата сила на кораба: **amount of guns \* blaster damage**
      - a. **Ако е така**, нека принтира:  
"Victory!"
      - b. **Ако не е така**:  
Проверка, дали **общата сила на противниковия кораб** е повече от **капацитета на щита** ни.  
**Ако е така**, нека принтира:  
"Defeat!"
    4. Накрая, ако **нито** нашият кораб, **нито** противниковият може да проникне през защитата на другия, нека принтира:  
"Both ships damaged!"



# Spaceship Factory

Трябва да има методи за създаване на два типа космически кораба, приемайки всичко нужно като параметър.

Трябва да има следните функции:

- **Create XWing:**

Приема: всички нужни параметри за създаване на XWing кораб;

Връща: XWing обект.

- **Create StarDestroyer:**

Приема: всички нужни параметри за създаване на StarDestroyer кораб;

Връща: StarDestroyer обект.

**След създаване на класовете, инициализирайте 2-3 обекта и тествайте системата, която сте създали!**

*"If fear is the path to the dark side, then a well written OOP class system should be the true way of the Force!"*

- Master Yoda, 25000 BBY

## 4. Train tickets system (Bonus)

В Правец е открита ЖП гара и на вас се пада задачата да имплементирате система, която да пресмята колко пари струват билетите за всяка дестинация.

На първия ред от програмата се въвежда броят на дестинациите. След това се въвеждат и самите дестинации, всяка на нов ред, във формат:

**<име на дестинация> <редовна цена на билет за тази дестинация>.**

След като дестинациите бъдат въведени, се преминава към въвеждането на закупените билети за всяка дестинация. За всеки билет се въвежда, на нов ред, следната информация:

**<възраст на пътника> <час на потегляне>.**

В зависимост от следните фактори, към ценообразуването на билетите се добавят различни отстъпки:

- При час на потегляне между 00:00 - 06:59 ч. и 19:00 - 23:59 ч., се добавя **5% отстъпка**.
- Деца до 7 години пътуват **безплатно**, деца от 7 до 12 години получават **20% отстъпка**, младежи от 12 до 18 години, както и възрастни хора над 64 години, получават **10% отстъпка**.
- При продажба на 4 или повече билета за една дестинация, към всеки от тези билети се добавя още **5% отстъпка**.

**Всички отстъпки се прилагат върху редовната цена на билета!**

За преминаване към следваща дестинация се въвежда **NEXT DESTINATION**.  
Когато всички дестинации бъдат обходени, на екрана се принтира „No more destinations left!“, последвано от изкараната сума за всяка дестинация.



Example 1	Logic	Example 2	Logic
Destinations: 3 Sofia 16bgn Varna 40bgn Ruse 35bgn  Sofia: 20y 13:30 NEXT DESTINATION  Varna: 40y 10:00 30y 13:30 NEXT DESTINATION  Ruse: 70y 05:25 NEXT DESTINATION  No more destinations left! Sofia: 16.00bgn Varna: 80.00bgn Ruse: 29.75bgn	Sofia -> 16.00лв. (без отстъпки) = 16.00лв  Varna -> 40.00лв. (без отстъпки) + 40.00лв (без отстъпки) = 80.00лв  Ruse -> 29.75лв. (35 - (15% от 35 = 5.25)) = 29.75лв	Destinations: 2 Sofia 15bgn Plovdiv 33.5bgn  Sofia: 5y 15:15 44y 15:15 14y 17:15 65y 06:00 10y 20:00 NEXT DESTINATION  Plovdiv: 2y 11:30 29y 11:30 NEXT DESTINATION  No more destinations left! Sofia: 50.5bgn Plovdiv: 33.5bgn	Sofia -> 0лв (под 7 години) + 14.25лв. (15 - (5% от 15 = 0.75)) + 12.75лв. (15 - (15% от 15 = 2.25)) + 12.00лв. (15 - (20% от 15 = 3)) + 11.50лв. (15 - (30% от 15 = 4.5)) = 50.50лв  Plovdiv -> 0лв (под 7 години) + 33.5лв (без отстъпки) = 33.5лв.

## 5. Agile (Bonus)

**Представете си, че попадате в следната ситуация:**

Имате да правите **проект** по даден предмет. Вашият клас е разделен на малки групи по **5-6 човека** и Вие сте в една от тези групи. Имате **2 седмици (10 учебни дни)** да завършите проекта. От това, което виждате в заданието, **този срок изглежда реалистичен**, ако всички от групата работят заедно.

В **началото на втората седмица** класният ръководител идва във вашата група и казва, че **директорът иска да се обнови** една от страниците на **сайта на гимназията** и това трябва да е случи веднага.

Обръща се към вашата група, защото най-често вие поддържате сайта. Такава работа **обикновено отнема 4-5 дни**.

Класният ръководител е човек, с когото можете спокойно да говорите за всичко, **разбира ви и винаги помага**. Директорът също е разбран, но и много авторитетен.

**Как ще подходите?** Опишете какви **действия** ще предприемете и как ще разрешите настъпилата ситуация.

**Помощни въпроси:**

- Ще се съгласите ли или ще откажете? Защо?
- Ще разговаряте ли с някого и с кого? За какво?
- Може ли да определите кое е по-важно - проектът или заявката на директора и да се обосновите?
- Можете ли да откажете да направите едно от двете?