

Упражнение – Допълнителни задачи

Problem 1. Превоз на товари

Дадено е транспортно предприятие за превоз на товари. Създайте два класа: клас Truck и клас Freight. Всеки камион трябва да има име, товароносимост и множество товари. Всеки товар трябва да има име и маса (тегло). Името не може да бъде празен низ. Товароносимостта и масата на товарите не може да бъдат отрицателно число.

Създайте програма, в която всяка команда отговаря на товарене на даден камион с даден товар. Ако камионът може да поеме товара - го товари на себе си. Ако камиона не разполага с достатъчна товароносимост, изведете подходящо съобщение ("[Truck] can't loaded [Freight]").

На първите два реда са дадени всички камиони с товароносимости, а на втория всички видове товари с масите си. След всички товарения да се изведат за всеки камион по реда на въвеждане всички товари, с които той е натоварен, също в реда на въвеждане на товарите. Ако нищо не е натоварено на даден камион, да се изведе името на камиона, последвано от **"Nothing loaded"**.

При въвеждане на невалидни (отрицателни маси на товари и товароносимости на камиони да се създаде изключение със съобщение: **"Weight cannot be negative"**) или празно име (празно име с изключение със съобщение : **"Name cannot be empty"**) за край на програмата с подходящо съобщение. Вижте примерите по-долу:

Примери

Вход	Изход
BigTruck=22;Truck=15;LittleTruck=7; Rock=18;Silk=12;Water=3 BigTruck Rock Truck Water Truck Water Truck Rock LittleTruck Water LittleTruck Water END	BigTruck loaded Rock Truck loaded Water Truck loaded Water Truck loaded Water LittleTruck loaded Water LittleTruck loaded Water BigTruck - Rock Truck - Water, Water, Water LittleTruck - Water, Water
BigTruck=20;Truck=10;LittleTruck=7; Rock=18;Silk=12;Water=3;Plastics=2; BigTruck Silk Truck Plastics BigTruck Rock Truck Rock LittleTruck Silk BigTruck Water END	BigTruck loaded Silk Truck loaded Plastics BigTruck can not load Rock Truck can not load Rock LittleTruck can not load Water LittleTruck loaded Water BigTruck loaded Water BigTruck - Silk, Water Truck - Plastics LittleTruck - Nothing loaded

Problem 2. Междузвездни войни – от 2 до 4 часа

Забележка: Тази задача, както и следваща може да бъдат предложени в различни нюанси – по-прости и по-усложнени. Накратко имаме:

Планети с ресурси – 1 или повече.

Сгради

- **Мини** , вида им зависи от вида на ресурсите, броят (нивото) им определя скоростта на добив на ресурсите от даден вид
- **Корабостроителници**, броят (нивото) им определя скоростта на изграждането на кораби

Кораби – няколко различни вида, за изграждането на които се иска наличието на определени ресурси на планетата. Времето за построяването им зависи от броя (нивото) на корабостроителниците на планетата.

Вашата задача е да съставите програма, която

- На 1 ред въвежда Планети с определени запаси от определени ресурси
- На 2 ред въвежда сгради и техните цена, като ресурси, които ще се похарчат за построяването ѝ
- На 3 ред въвежда кораби и техните цени като ресурси, които ще се похарчат за построяването им

Може да се ползва команда `TIME [time]`, която влияе на запасите от ресурсите на планетата, както и на напредъка в корабостроенето. По сценарий – колкото повече време работят мини и корабостроителници на по-високо ниво (или повече като брой), толкова по бързо добиват единица ресурс или построяват кораб.

`TIME [10]` – води до увеличаването на ресурсите и корабите пропорционална на 10.

По-долу са дадени два примерни сценария. Ако желаете може да редуцирате ресурси или сгради и мини или кораби.

Създайте три класа: клас **Planet**, клас **Building** и клас **Ship**. Всяка Планета **Planet** има три полета **Name**, **Metal** и **Mineral**, показващи залежите към текущия момент на планетата. Всяка сграда **Building** има три полета **Name**, **Metal** и **Mineral**, показващи необходимите ресурси за построяване на дадената сграда. Всеки кораб **Ship** има три полета **Name**, **Metal** и **Mineral**, показващи необходимите ресурси за построяване на дадения вид кораб.

Нека имената на сградите са: **MetalMine**, **MineralMine**, **YardShip**, а имената на кораби са: **Transporter**, **Battle Ship**, **Fighter**.

Ако ресурсите са достатъчни за построяването на сградата да се изведе съобщение

„On [Име на планета] was builded [Име на сграда]“

Ако ресурсите са достатъчни за построяването на кораб да се изведе съобщение

„On [Име на планета] was builded [Име на кораб]“

Полето **Name** на всички обекти не може да е празно, стойностите на **Metal** и **Mineral** на **Building** и **Ship** са неотрицателни числа. (или поне едно от тях е неотрицателно **) , в противен случай се генерират изключения с подходящи съобщения за възникналите грешки.

Създайте програма, в която всяка команда отговаря на построяване на сграда на дадена планета. Ако Сградата може да се построи, се добавя към наличните на планетата и се извежда **„On [Име а планета] was builded [име на сграда]“**

Ако няма достатъчно ресурси, изведете подходящо съобщение ("**[Име на планета] have not resource [Име на ресурс] to build [име на сграда]**").

Ако Корабът може да се построи се добавя към наличните на планетата. Ако ресурсите от даден вид не достигат за построяване на сграда, да се генерира изключение със подходящо съобщение („**[Име на ресурс] на [Име на Планета] not enough to build a [Име на Сграда]**“). Ако ресурсите от даден вид недостигат за построяване на кораба, да се генерира изключение със подходящо съобщение („**On [Име на Планета] can not build a [Име на Кораб] [брой] units**“). Изведете подходящо съобщение:

- На първия ред се въвеждат всички планети с всички съответни количества ресурси.
- На втори ред се въвеждат всички Сгради с имена и ресурси
- На трети ред се въвеждат всички Кораби с имена и ресурси
- На следващите редове се подават команди за строене на сгради и кораби

След всички построения да се изведат за всяка планета по реда на въвеждане всички сгради и кораби, оито са построени, също в реда на въвеждането им.

Ако не са построени сгради, да се изведе "**On [име на планета] there are not buildings**".

Ако не са построени кораби, да се изведе "**On [име на планета] there are not ships**".

- При въвеждане на невалидни (отрицателна количества ресурс да се създаде изключение със съобщение: "Amount **cannot be negative**") или празно име (празно име с изключение със съобщение : "**Name cannot be empty**")

Команда **END** – край на въвеждането – извежда на екрана състоянието на планетите:

[Име на планета] [Metal] [количество] [Mineral] [количество]

[Име на планета] MetalMine [ниво] MineralMine [ниво] ShipYard [ниво]

[Име на планета] Transporter [брой] BattleShip [брой] Fighter [брой]

Забележка: Възможно е да има неточности във форматирането и изходните данни! Да се провери!

Примери

Вход	Изход
Alpha 3000 5000 Beta 4000 2500 Mars 5000 2000	On Alpha was builded MetalMine On Alpha was builded MineralMine On Bet was builded a MetalMine On Mars was builded MineralMine
MetalMine 1000 200 MineralMine 100 2000	On Mars was builded MetalMine On Beta was builded MineralMine
Transporter 200 300 BattleShip 400 400 Figher 100 100	On Alpha was builded Transporter 2 On Alpha can not to build Fighter 1000 units On Beta can not to build BattleShip 4 On Alpha was builded MineralMine
Alpha MetalMine Alpha MineralMine Beta MetalMine Mars MineralMine Mars MetalMine Beta MineralMine Alpha Transporter 2 Alpha Fihter 1000	Resources: Alpha Metal 1400 Mineral 200 Beta Metal 2900 Mineral 100 Mars Metal 5000 Mineral 2000 Buildings: Alpha MetalMine 1 MineralMine 2 ShipYard 0

Beta BattleShip 4 Alpha MineralMine END	Beta MetalMine 1 MineralMine 1 ShipYard 0 On Mars there are not buildings Ships: Alpha Transporter 2 BatleShip 0 Fighter 0 On Beta there are not ships On Mars there are not ships
---	---

Problem 3. Междузвездни войни* (Империята отвърща на удара)

Добавете към програмата следната функционалност:

- Изминало време – TIME [time] т.е. време, което е изминало и наличните сгради са повлияли на :
 - добива на ресурс по формулата: Ресурс+=брой сгради*100*[time]
 - времето за строенето на кораби, като то намалява при увеличаване на нивото на корабостроителницата (това условие може да се реализира на последващ етап)

Примери

Примерът е както предишния с изключение, че може да се даде команда TIME [време], която по формулата с величини време и ниво на сградите ще повиши ресурсите на планетата.

Ако точно преди END дадем команда TIME 20, за планета Alpha ресурсите ще се повишат.

Вход	Изход
TIME 20	

Problem 4. В най-тъмните подземия (Dungeonest Dark)

Като млад авантюрист търсиш злато и слава в най-тъмните подземия.

Имате първоначално **здраве 100** и първоначални **монети 0**. Ще ви бъде даден низ, представляващ стаите в подземие, където сте изпратен на мисия. Информацията за всяка стая е разделена от останалите с '|' (вертикална черта): "стая1 | стая2 | стая3 ..."

Всяка стая съдържа име на намерен предмет или чудовище и цяло число, разделени с интервал.

- Ако първата част е **"potion"**, то сте попаднали на отвара, която ви лекува. Увеличете здравето на героя ви с числото във втората част. Но вашето здраве не може да надвишава първоначалното (100). Освен това, ако дадена отвара ви дава възможност да се излекувате над 100, то вие не може да се възползвате напълно от цялата ѝ сила, а само от тази нейна част, която довежда здравето ви до 100. Отпечатайте: **"You healed for {0} hp."**, където {0} е частта от отварата, от която сте се възползвали. След това, отпечатайте текущото си здраве: **"Current health: {0} hp."**
- Ако първата част е **"chest"**, то вие сте намерили сандък с монети, колкото е числото на втора позиция. Отпечатайте **"You found {0} coins."** и ги прибавете към вашите.
- **Във всеки друг случай** сте изправени пред чудовище, ще трябва да се биете. Втората част на информацията на стаята съдържа атаката на чудовището. Трябва да извадите силата на атаката на чудовището от вашето здраве. После:
 - Ако още сте жив (здраве > 0), то вие сте убили чудовището и трябва да се изведе на конзолата **"You slayed {monster}."**
 - Ако сте умрели, изведете **"You died! Killed by {monster}."** и най-далечната стая до която сте успели да достигнете: **"Best room: {room}"**. С това вашата мисия приключва.

Ако сте успели да преминете през всички стаи в подземие, изведете на конзолата следващите три реда:

"You've made it!", "Coins: {coins}", "Health: {health}".

Вход

Получавате низ, представляващ стаите в подземие, разделение с '|': **"room1|room2|room3..."**.

Изход

Отпечатайте съответните съобщения, описани по-горе.

Примери

Вход	Изход
rat 10 bat 20 potion 10 rat 10 chest 100 boss 70 chest 1000	You slayed rat. You slayed bat. You healed for 10 hp. Current health: 80 hp. You slayed rat. You found 100 coins. You died! Killed by boss. Best room: 6

Вход	Изход
cat 10 potion 30 orc 10 chest 10 snake 25 chest 110	You slayed cat. You healed for 10 hp. Current health: 100 hp. You slayed orc. You found 10 coins. You slayed snake. You found 110 coins. You've made it! Coins: 120 Health: 65

... игра, в която всеки герой печели деня с блестяща броня и усмивка ...

Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма **"Обучение за ИТ кариера"** на МОН за подготовка по професия "Приложен програмист".



Министерство
на образованието
и науката



Национална
програма
„Обучение за
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от **фондация "Софтуерен университет"** и се разпространява под **свободен лиценз CC-BY-NC-SA** (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).



SoftUni
Foundation

