Работа с класове и обекти

**Задача 1:**

Напишете клас **SwiftDate**, в който да бъдат дефинирани полета година (**year**), месец (**month**), ден (**day**). Класът да има следните методи:

|  |  |
| --- | --- |
| **boolean isLeapYear()** | проверява дали датата, съдържана в обекта, е във високосна година |
| **int getCentury()** | връща века, в който е датата, съдържана в обекта |
| **int getDaysDifference**  (SwiftDate other) | връща разликата в дни между датата, съдържана в обекта, и подадената като параметър дата |
| **void printInfo()** | принтира на екрана датата, последвана от века, в който е. Ако годината е високосна, добавя едно изречение, казващо “It is LEAP year.” Изхода изглежда така:  *yyyy MM dd - <XX> century.[ It is LEAP year.]* |

Напишете програма, която чете 2 дати от стандартния вход на 2 отделни реда, във формата “yyyy MM dd”. На стандартния изход отпечатва 3 реда:

a. разликата в брой дни, между двете дати (вкл. двата края на интервала)

b. резултата от **printInfo()** на първата дата

c. резултата от **printInfo()** на втората дата

Пример:

|  |  |
| --- | --- |
| Вход: | Изход: |
| 2016 09 26  1878 03 03 | 50612  2016 09 26 – 21 century. It is LEAP year.  1878 03 03 – 19 century. |

|  |  |
| --- | --- |
| Вход: | Изход: |
| 2016 09 26  2017 01 01 | 98  2016 09 26 – 21 century. It is LEAP year.  2017 01 01 – 21 century. |

**Задача 2:**

Създайте клас **Employee**, който съдържа задължителните характеристики име (**name**), заплата (**salary**), позиция (**position**), отдел (**department**) и незадължителните години (**age**) и електронна поща (**email**). Напишете подходящите полета, свойства и конструктори. Напишете програма, която чете от стандартния вход число N и след това N на брой реда във формата:

*<ИМЕ>,<ЗАПЛАТА>,<ПОЗИЦИЯ>,<ОТДЕЛ>[,<ГОДИНИ>][,<E-MAIL>]*

Програмата да изведе тримата работници с най-високи заплати, подредени в низходящ ред, като отпечата характеристиките име, отдел, позиция и ако работника има електронна поща, също. Формата да бъде следния:

*<ИМЕ>, <ОТДЕЛ>, <ПОЗИЦИЯ>[, <E-MAIL>]*

*Пример*:

|  |  |
| --- | --- |
| Вход: | Изход: |
| 5  Pesho, 2000, Developer, DevOps, 26, [pesho@gmail.com](mailto:pesho@gmail.com)  Gosho, 2800, Leader, DevOps  Svetla, 1970.11, Manager, Sales, 31, [svetla@gmail.com](mailto:svetla@gmail.com)  Iva, 999, Janitor, Cleaning  Tina, 3333.33, Director, Sales, 42, tina@sales.my | Tina, Sales, Director, [tina@sales.my](mailto:tina@sales.my)  Gosho, DevOps, Leader  Pesho, DevOps, Developer, pesho@gmail.com |

**Задача 3:**

Създайте клас **Vehicle**, който съдържа задължителните характеристики вид (**type**), модел (**model**), сила на двигателя (**power**), разход в литри на 100 км (**fuelConsumption**), година на производство (**yearProduced**), регистрационен номер (**licenseNo**) и незадължителните тегло (**weight**) и цвят (**color**). Напишете подходящите полета, свойства и конструктори. Напишете следните методи:

|  |  |
| --- | --- |
| **double calculateTripPrice (double fuelPrice, double distance)** | изчислява цената на пътуване |
| **double getInsurancePrice()** | Изчислява цената на застраховка по следната формула: ***( 0.16 \* power ) \* (1.25 \* carAge ) \****  ***( 0.05 \* fuelConsumption ) \* typeCoefficient***  ***carAge –*** възрастта на колата  ***typeCoefficient:***  • 1.00, ако **type** e автомобил (“car”)  • 1.12, ако **type** e ван (“suv”)  • 1.20, ако **type** e камион (“truck”)  • 1.50, ако **type** e мотоциклет (“motorcycle”) |

Напишете програма, която чете от стандартния вход числа N, F и N на брой последващи описания на превозни средства. F е цената на горивото, необходима за пресмятането на цената на пътуване, а описващите редове са в следния вид:

*<ТИП>, <МОДЕЛ>, <СИЛА>, <РАЗХОД>, <ГОДИНА>, <РАЗСТОЯНИЕ>[,<ТЕГЛО>][, <ЦВЯТ>]*

Програмата да генерира регистрационен номер за всяко въведено МПС (неповтарящ се идентификационен номер от 0000 до 9999) и да отпечатва следната информация:

*<РЕГ. НОМЕР> - <МОДЕЛ>, <ГОДИНА>, <ЦВЯТ> Insurance cost: <ЦЕНА ЗА ЗАСТРАХОВКА> - Тravel cost: <ЦЕНА НА ПЪТУВАНЕТО>*

Ако за дадено превозно средство няма информация за цвят, да се извежда N/A на негово място.

Пример:

|  |  |
| --- | --- |
| Вход: | Изход: |
| 4  1.89  car, Mercedes SLK 208, 185, 13.5, 2002, 300, 1850, green  car, Opel Astra, 115, 9.5, 2003, 255, 1440, gray  suv, Mercedes Vaneo, 177, 16, 2008, 1100, 3120, white  truck, Chavdar, 355, 18, 1986, 46 | 0000 - Mercedes SLK 208, 2002, green Insurance cost: 349.65 - Travel cost: 76.55  0001 - Opel Astra, 2003, gray Insurance cost: 142.03 - Travel cost: 45.79  0002 - Mercedes Vaneo, 2008, white Insurance cost: 253.75 - Travel cost: 332.64  0003 - Chavdar, 1986, N/A Insurance cost: 2300.40 - Travel cost: 15.65 |