

Лекция 15. Класове и обекти, Полета, Методи.

Задача 1:

Създайте клас Computer, който представя компютър.
Класът да има следните полета:

`year` – числова стойност, показваща година на производство на компютъра
`price` – числова стойност (не е задължително да е цяло число), показваща цената на компютъра
`isNotebook` – булева стойност – дали компютъра е преносим или не
`hardDiskMemory` – числова стойност за размера на хардиска
`freeMemory` – числова стойност, показваща размера на свободната памет
`operationSystem` – текстово поле за операционната система на компютъра

Класът да дефинира следните методи:

-метод `changeOperationSystem(newOperationSystem)`, който сменя стойността на полето `operationSystem` със стойността, подадена като параметър.

-метод `useMemory(memory)`, който намалява свободната памет (`freeMemory`) със стойността, подадена като аргумент.
Ако стойността на аргумента е по-голяма от свободната памет, извежда съобщение `"Not enough free memory!"`

Да се създадат 2 обекта от тип Computer.

Да се зададат стойности на всеки от компютрите за `year`, `price`, `hardDiskMemory`, `freeMemory`, `operationSystem`. Нека единият компютър да е лаптоп. На единия от двата компютъра да се задели памет 100 (чрез метода `useMemory`), а на другия, да се смени операционната система (чрез метода `changeOperationSystem`), след което да се изведат на екрана всичките полета на двата компютъра.

Задача 2:

Да се създадат класовете Call и GSM, представлящи съответно телефонно обаждане и телефон.

Класът Call да съдържа следните полета:

`priceForAMinute` – статично поле, което показва цената на разговор за минута

`caller` - показва телефона, от който е направено повикването
`receiver` - показва телефона, към който е направено повикването
`duration` - дължината на разговора (в минути)

Класът GSM да съдържа полетата:

`model` – модел на телефона

`hasSimCard` – дали в телефона е заредена SIM карта

`simMobileNumber` – номер на SIM картата(телефонен номер), ако е заредена такава.

`outgoingCallsDuration` – общото време (в минути) на изходящите повиквания

`lastIncomingCall` – последното входящо повикване

`lastOutgoingCall` – последното изходящо повикване

Да се напишат следните методи за класа GSM:

- `insertSimCard(simMobileNumber)` – метода задава номер(сим карта) на телефона. Да се провери дали подаденият номер е валиден (започва с 08 и се състои от 10 цифри). Ако номерът е валиден, го задава на телефона и задава стойност `true` на полето `hasSimCard`

- `removeSimCard()` - премахва сим картата от телефона (задава `false` на полето `hasSimCard`)

- `call(receiver, duration)` – В тялото му да се направят проверки за:

- дали въведената дължина на разговора е валидна
- дали двата телефона (този за който се извиква метода и този към който се прави обаждането) не са един и същ телефон
- дали и двата телефона имат сим карта

Ако всички проверки преминат успешно, метода прави обаждане с продължителност `duration` към телефона, подаден като параметър. Задава това обаждане като последно изходящо повикване на телефона от който се прави повикването (телефона за който се извиква метода) и задава същото обаждане като последно входящо за телефона към който се прави обаждането.

Осен това увеличава стойността на `outgoingCallsDuration` със дължината на разговора за телефона от който се прави обаждането.

Да се направи и метод `getSumForCall()`, който връща сумата, начислена за изходящите повиквания на телефона (общото време на изходящите повиквания по цената за минута - `priceForAMinute`).

Да се направят два метода

`printInfoForTheLastOutgoingCall()` и

`printInfoForTheLastIncomingCall()`

които извеждат информация за последното изходящо/входящо повикване на телефона (ако има такава)