Industrial Software Development (ISDe) – February 15, 2021

Name: ..... Surname: ..... Student ID: .....

EXERCISE 2 (10 points) – 30 minutes

We consider a game with gnomes and orcs. Both **Gnome** and **Orc** are subclasses of **Character**. At creation, each gnome has an initial energy of 100, while each orc has an initial energy of 200. Both increase their energy by the amount **x** using the **.eat(x)** method.

(a) Implement the classes, trying to concentrate the common code in the superclass.

Write your code into a 'main_2a.py' file

```
# Use exactly this 'MAIN'
# This code must run correctly, producing the showed output.
# Write your code into a 'main_1.py' file

orc1 = Orc()
gnome1 = Gnome()

print('orc1 energy:', orc1.energy)
print('gnome1 energy:', gnome1.energy)

print('gnome1 eats 10')
gnome1.eat(10)
print('gnome1 energy:', gnome1.energy)
```

The output is:

```
orc1 energy: 200
gnome1 energy: 100
gnome1 eats 10
gnome1 energy: 110
```

(b) When a character attacks another, they both lose energy. If the attack occurs during the day, the attacker loses 20% energy while the victim loses 30% energy. If the attack occurs at night, the attack is more effective thanks to the surprise effect. The attacker loses only 10% of energy while the victim loses 40% of energy.

Use the **strategy desing pattern** to implement this behavior.
Write your code into a '**main_2b.py**' file

```
# Use this 'MAIN'
# complete where necessary, but without deleting or altering the instructions
already present
# This code must run correctly, producing the showed output.
# Write your code into a 'main_2.py' file

orc1 = Orc()
gnome1 = Gnome()

## ATTACK - DAY
print('\n \n')
print('\nATTACK - DAY.')
print('orc1 energy:', orc1.energy, '-  gnome1 energy:', gnome1.energy)
print('orc1 attacks gnome1!')
orc1.attack(gnome1)
print('orc1 energy:', orc1.energy, '-  gnome1 energy:', gnome1.energy)


## ATTACK - NIGHT


print('\n \n')
print('\nATTACK - NIGHT.')
print('orc1 energy:', orc1.energy, '-  gnome1 energy:', gnome1.energy)
print('orc1 attacks gnome1!')
orc1.attack(gnome1)
print('orc1 energy:', orc1.energy, '-  gnome1 energy:', gnome1.energy)
```

The output is:

```
ATTACK - DAY.
orc1 energy: 200 -  gnome1 energy: 100
orc1 attacks gnome1!
orc1 energy: 160.0 -  gnome1 energy: 70.0



ATTACK - NIGHT.
orc1 energy: 160.0 -  gnome1 energy: 70.0
orc1 attacks gnome1!
orc1 energy: 144.0 -  gnome1 energy: 42.0
```