



Curso: Computação II (MAB225)

Nível: Graduação Período: 2014.2

Aluno: \_\_\_\_\_

## Primeira Prova (P1) - 02 Out. 2014

1) [2.5 pt] Desenvolva uma função "obtemDoUsuario", com um funcionamento similar ao da função `input`. Ela recebe como parâmetro o texto que será exibido em tela para fazer a requisição de um valor, e deve retornar o valor digitado. Porém a função `obtemDoUsuario` deve obrigatoriamente retornar um valor válido que tenha sido digitado pelo usuário. Caso o usuário digite um valor que cause uma exceção (qualquer uma que herde *Exception*), o programa deve requisitar que ele tente digitar o valor novamente, até obter um valor válido. Escreva o texto descritivo de cada exceção levantada em um mesmo arquivo texto, cumulativamente (use o modo *append*, parâmetro "a").

2) [7.5 pt] O código abaixo representa uma forma muito simples de "bichinho virtual" (virtual pet), e está incompleto.

```
class VPet(object):
    MAX = 100.0
    MIN = 0.0
    def __init__(self,n,f,i):
        self.setNome(n)
        self.setFome(f)
        self.setIdade(i)

def main():
    n=input("Nome do seu bichinho virtual: ")
    pet = VPet(n,0.0,0)
    continuar = True
    while continuar:
        print pet
        pet.setFome(pet.getFome() + 10)
        pet.setIdade(pet.getIdade() + 1)
        acao = input("b=brincar; a=alimentar")
        if acao=="b":
            pet.setFome(pet.getFome() + 10)
        elif acao=="a":
            pet.setFome(pet.getFome() - 100)
            if pet.getIdade()>=100 or pet.getFome()==100.0:
                continuar = False
        print pet
        print ("Seu bichinho viveu %d anos"%(pet.getIdade()))

if __name__=="__main__": main()
```

a. [1.0 pt] Complete o programa escrevendo os métodos `setNome`, `setFome` e `setIdade`

para a classe `VPet`. Certifique-se de que (i) o atributo de instância referente ao `setFome` nunca possua valor menor que `MIN` ou maior que `MAX` (ambos atributos de `VPet`) e (ii) o atributo de instância referente ao `setIdade` nunca possua valor menor que `MIN`.

- b. [0.5 pt] Escreva os métodos `getNome`, `getFome` e `getIdade` para a classe `VPet`.
- c. [1.0 pt] Quantos anos, no mínimo e no máximo, vive um bichinho virtual nesse programa?
- d. [0.5 pt] Os comandos "print pet" na função `main()` exibem em tela uma string "incompreensível" para o usuário. Sem alterar a função `main()`, proponha uma representação mais adequada da instância de `VPet`, que exiba seus atributos relevantes (sobrescreva `__repr__`).
- e. [0.5 pt] Ainda sobre o item anterior, como ficaria o método `__repr__` se fosse necessário exibir ao usuário, além da representação proposta por você, também, juntamente, as mesmas informações incompreensíveis?
- f. [3.0 pt] Imagine que o usuário precisa desligar seu computador, mas gostaria de salvar o seu bichinho virtual para continuar cuidando dele quando estiver com tempo livre novamente (ligando o computador novamente mais tarde). Crie um método "gravar()" e um método "carregar()" para a classe `VPet`, que atendam a essa necessidade. Trate exceções que possam ser levantadas dentro dessas funções.
- g. [1.0 pt] Sem alterar a classe `VPet` (considerando que ela possui todos os métodos desenvolvidos nos itens anteriores), e nem a função `main()`, como você faria para criar um novo tipo de bichinho virtual (`VPetB`) mais sofisticado (cujas instâncias possuam um atributo a mais como, por exemplo, saúde)? `VPet` deve poder, por exemplo, ser substituído por `VPetB` em toda a função `main()` original, sem alteração na funcionalidade do programa.

---

- Métodos para manipular arquivos: `open()`; `write()`; `read()`; `readlines()`; `readline()`; `close()`; `writelines()`