

Fundamentos da Programação LEIC/LETI

Epílogo

Projeto 2. Representação gráfica. Tratamento de exceções.

Aula 27

Alberto Abad, Tagus Park, IST, 2021-22

Planeamento sessão

- Projeto 2
- Representação gráfica
- Tratamento de exceções
- Módulos
- Desafios de programação:
- Epílogo

O Prado

- Dúvidas [enunciado](#)
(<https://fenix.tecnico.ulisboa.pt/downloadFile/563568428820899/FP2122P2.20211029.pdf>)
- Dúvidas sobre os testes do Mooshak

O prado

TADs no projeto

TAD posicao

O TAD posicao é usado para representar uma posição (x, y) de um prado arbitrariamente grande, sendo x e y dois valores inteiros não negativos.

TAD animal

O TAD animal é usado para representar os animais que habitam o prado, existindo de dois tipos: predadores e presas. Os predadores são caracterizados pela espécie, idade, frequência de reprodução, fome e frequência de alimentação. As presas são apenas caracterizadas pela espécie, idade e frequência de reprodução.

TAD prado

O TAD prado é usado para representar o mapa do ecossistema e as animais que se encontram dentro.

O prado

geracao

- Seguindo a ordem de leitura do prado, cada animal (**vivo**) realiza o seu turno:
 1. Aumenta idade e fome
 2. Animal tenta movimentarse:
 - a. Presas para posição livre
 - b. Predadores para posição com presa, ou alternativamente, para posição livre
 3. Se conseguiu movimentar e atingiu idade de reprodução
→ Animal reproduz
 4. Se é predador e movimentou para posição de presa:
 - a. O predador se alimenta
 - b. A presa morre
 5. Se é predador e fome igual a frequencia de alimentação
→ Predador morre

Representação gráfica

Existem várias bibliotecas disponíveis para desenvolver GUIs. Entre as mais comuns estão:

- pygame (<http://pygame.org/> (<http://pygame.org/>))
 - Um tutorial em português: <https://old.gustavobarbieri.com.br/jogos/jogo/doc/> (<https://old.gustavobarbieri.com.br/jogos/jogo/doc/>)
- Módulo `graphics` de John Zellee (<http://mcsp.wartburg.edu/zelle/python/graphics.py> (<http://mcsp.wartburg.edu/zelle/python/graphics.py>) and <http://mcsp.wartburg.edu/zelle/python/graphics/graphics/index.html> (<http://mcsp.wartburg.edu/zelle/python/graphics/graphics/index.html>))
 - O livro da UC mostra uma simulação de uma fila de supermercado usando o `graphics`.
- Módulo `tkinter` é uma interface em Python para TCL/TK:
<https://docs.python.org/3/library/tkinter.html> (<https://docs.python.org/3/library/tkinter.html>)

Desafio: Desenvolver (ou melhorar) GUI para o projeto

Representação gráfica

Prado GUI (com tkinter)



- Código disponível no [site da disciplina](https://fenix.tecnico.ulisboa.pt/downloadFile/845043405558124/PradoGUI.zip) (<https://fenix.tecnico.ulisboa.pt/downloadFile/845043405558124/PradoGUI.zip>)
- **ATENÇÃO!!** O projeto deve conter a função publica `parse_config(filename)`

Exceções

Tratamento de exceções

```
try:
    print(x)
except NameError:
    print("WE can handle exceptions")
else:
    print("do something only if no exception happened")
finally:
    print("and do something always at the end")
```

```
In [23]: def func():
    try:
        # x = 0
        print("Primeiro aqui")
        print(x)
        print("Ja nao estou aqui")

    except NameError:
        print("WE can handle exceptions")

    except Exception:
        print ("Fiquei aqui")

    else:
        print("do something only if no exception happened")
    finally:
        print("and do something always at the end")

func()
```

```
Primeiro aqui
WE can handle exceptions
and do something always at the end
```

Exceções

Atenção com a ordem das exceções!!

```
In [2]: def func():
    try:
        # x = 0
        print("Primeiro aqui")
        print(x)
        print("Ja nao estou aqui")

    except Exception:
        print ("Geral exeception")

    except NameError:
        print("Particular exeption")

func()
```

```
Primeiro aqui
Geral exeception
and do something always at the end
```

Exceções

Podemos voltar a levantar exceções (já *apanhadas*)

```
In [25]: try:
          raise NameError('Hello world')
        except NameError:
          print('An exception happened and it will be raised again!')
          raise
```

An exception happened and it will be raised again!

```
-----
-----
NameError                                Traceback (most recent c
all last)
<ipython-input-25-5f99c4721190> in <module>
      1 try:
----> 2     raise NameError('Hello world')
      3 except NameError:
      4     print('An exception happened and it will be raised aga
in!')
      5     raise

NameError: Hello world
```

Módulos Python (STD Library)

<https://docs.python.org/3/library/> (<https://docs.python.org/3/library/>)

- **functools**: Higher-order functions and operations on callable objects
- **itertools**: Functions creating iterators for efficient looping
- **random**: Generate pseudo-random numbers
- **pickle**: Python object serialization
- **csv**: CSV File Reading and Writing
- **logging**: Logging facility for Python
- **re**: Regular expression operations
- **copy**: Shallow and deep copy operations
- **doctest**: Test interactive Python examples
- **unittest**: Unit testing framework

Módulos Python (third-party)

- **NumPy**: operações matemáticas, nomeadamente sobre estruturas multidimensionais (eg, matrizes)
- **Pillow**: funções para manipulação de imagens
- **Matplotlib**: geração de gráficos e *plots* 2D
- **Requests**: "HTTP for Humans"
- **Pandas**: funções para processamento de conjuntos de dados complexos
- **Pywin32**: funções de acesso à API do Windows
- **Keras, Tensorflow, PyTorch, Scikit**: machine learning
- **NLTK, OpenCV** - NLP e *processamento de imagem*
- ...

Desafios

- Projetos para treino de habilidades de programação:
 - [Project Euler \(https://projecteuler.net\)](https://projecteuler.net)
 - [Advent of Code \(https://adventofcode.com\)](https://adventofcode.com)
 - [Codewars \(https://www.codewars.com\)](https://www.codewars.com)
 - [MIUP \(https://miup20.tecnico.ulisboa.pt\)](https://miup20.tecnico.ulisboa.pt) / [SWERC \(https://swerc.eu\)](https://swerc.eu)

Project Euler.net



Perspetiva do Curso LEIC

- **Fundamentos da Programação** (1º ano, 1º período)
 - bases programação (programação imperativa, programação funcional - Python)
- **Lógica para Programação** (1º ano, 2º período)
 - programação declarativa (Prolog)
- **Introdução aos Algoritmos e Estruturas de Dados** (1º ano, 3º período)
 - algoritmos, eficiência (programação imperativa - C)
- **Introdução à Arquitetura de Computadores** (1º ano, 4º período)
 - cheirinho de programação Assembly
- **Programação com Objectos** (2º ano, 1º período)
 - programação baseada em objetos (Java, C++)
- **Sistemas Operativos** (2º ano, 2º período)
 - programação paralela com *threads* (C)
- **Bases de Dados** (2º ano, 4º período)
 - *domain specific language* (DSL) (SQL)
- **Compiladores** (3º ano, 4º período)
 - processadores de linguagens (C++)

Perspetiva do Curso LETI

- **Fundamentos da Programação** (1º ano, 1º período)
 - bases programação (programação imperativa, programação funcional - Python)
- **Introdução aos Algoritmos e Estruturas de Dados** (1º ano, 3º período)
 - algoritmos, eficiência (programação imperativa - C)
- **Arquitetura de Computadores** (1º ano, 4º período)
 - cheirinho de programação Assembly
- **Programação com Objectos** (2º ano, 1º período)
 - programação baseada em objetos (Java, C++)
- **Sistemas Operativos** (2º ano, 2º período)
 - programação paralela com *threads* (C)
- **Bases de Dados** (2º ano, 4º período)
 - *domain specific language* (DSL) (SQL)

A seguir em FP...

- Publicação de notas e **revisão de provas da ficha 6 e ficha de repescagem**
 - data e hora a definir durante a próxima semana
 - nota de componente teórica fica definida esta semana
- Publicação de notas e **revisão de provas do projeto 1**
 - data e hora a definir, *possivelmente* antes do exame
- Entrega do projeto 2 até **19 Novembro às 17h**
 - publicação de notas e **revisão de provas do projeto 2** (data e hora a definir, após do exame).
- Exame de repescagem **24 de Novembro às 13h** (opcional):
 - Conta a melhor nota entre exame e fichas
 - Inscrição obrigatória
 - Publicação de notas e **revisão de provas do exame** (data e hora a definir, semanas a seguir).

Horários dúvidas FP

- Esta semana terminam os horários de dúvidas do período letivo, mas:
 - Horários de dúvidas nas **próximas semanas**:
 - Prof. Alberto Abad (Sala 2.N11.21): 16, 19 e 23 de Novembro das 9 às 12h
 - Potencialmente híbrido on-line/presencial
 -
 - O Slack continua a funcionar: <http://fp-tagus.slack.com> (<http://fp-tagus.slack.com>)

Obrigado!