

ATP 2023/24

Algoritmos e Técnicas de Programação
(Lic. Eng. Biomédica)

Equipe docente

- José Carlos Ramalho (jcr@di.uminho.pt)
 - subject: ATP2023::assunto
- Luís Filipe Cunha (lfc@di.uminho.pt)
 - subject: ATP2023::assunto

Objetivo final

Programar:

- Conhecer as técnicas básicas de programação;
- Saber modelar um problema e implementá-lo na linguagem de programação Python.

Ética

“Ética é a **obediência** ao que não é obrigatório, é bom, é justo, coloca o **coletivo à frente do individual...**”

Avaliação

1. Teste escrito - 45% (nota mínima: 10val)
2. Projeto em grupo (até 3 elementos) - 35% (nota mínima: 10 val, **eliminatório)
3. TPC - 20%

TPC

1. Total de 8: 1 por semana
2. Cada aluno deverá criar um repositório no GitHub ou GitLab;
3. Cada TPC deverá ser colocado em pasta própria no repositório: TPC1, TPC2, ... TPC8;
4. Cada TPC deverá incluir um manifesto, que será definido na 1ª aula e todos os outros ficheiros resultantes do desenvolvimento do trabalho.

Nota: Qualquer tentativa de fraude comprovada levará à anulação de toda esta componente ou, em casos de maior gravidade, a processo disciplinar.

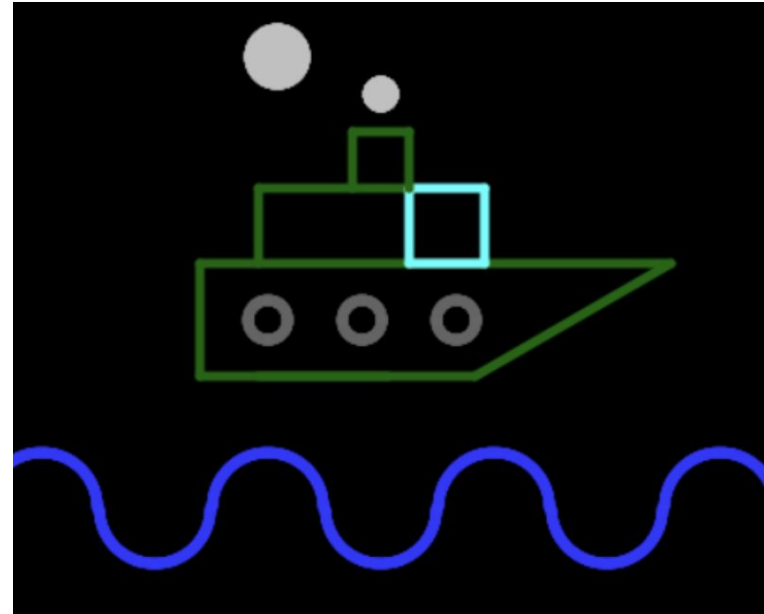
TPC1: de 11-12 a 18-19 de Setembro 2023

1. Criar conta no Github;
2. Criar repositório no Github: ATP2023;
3. Criar pasta TPC1 no repositório;
4. Colocar em TPC1 print-screen do Maze(10);
5. Colocar em TPC1 print-screen da tartaruga do barco;
6. Enviar email ao docentes:

jcr@di.uminho.pt; lfc@di.uminho.pt

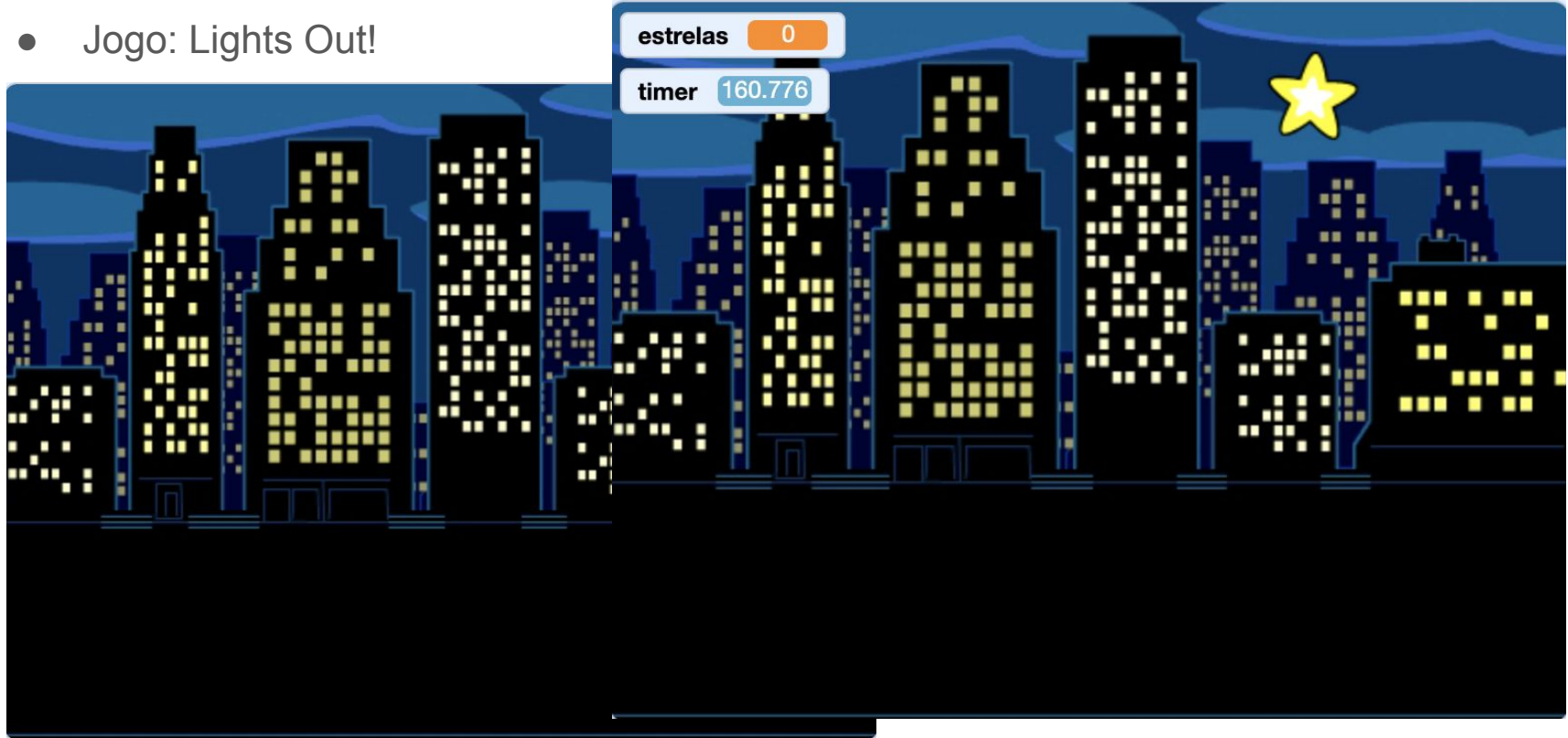
subject: ATP2023::Axxxxx::TPC1

mensagem: link para o repositório no Github



TPC2: de 18-19 a 25-26 de Setembro 2023

- Jogo: Lights Out!



Projeto

1. Equipe formada livremente pelos alunos e que poderá ter dois ou três elementos;
2. O enunciado será apresentado entre a quinta e a sexta semana de aulas;
3. A sua entrega será feita em data a combinar com os alunos;
4. Após a entrega, cada equipe agendará uma sessão de 20 minutos com o docente para apresentar e defender o seu trabalho;
5. O trabalho terá dois resultados: um relatório técnico (20%), e uma aplicação informática (80%).

Teste escrito

1. Consistirá num pequeno conjunto de pequenos problemas para os quais o aluno deverá apresentar soluções em Python;

Programa

1. Arquitetura simplificada de um computador;
2. Máquina virtual: conceito;
3. Variáveis de tipos simples: inteiros e reais;
4. Estruturas condicionais: if-else, if-elif-else;
5. Estruturas cíclicas: while, for;
6. Definição de funções;
7. Scope de variáveis;
8. Tipos de dados estruturados: strings, listas, tuplos, dicionários, ...
9. Canais de entrada e saída: stdin, stdout, ficheiros;
10. Definição de módulos;
11. Vários tipos de aplicações: processamento de dados, aplicações web, ...

Ferramentas e outros

1. Github ou Gitlab: TPC, Projeto, exercícios das aulas;
2. Python 3;
3. Jupyter Notebook;

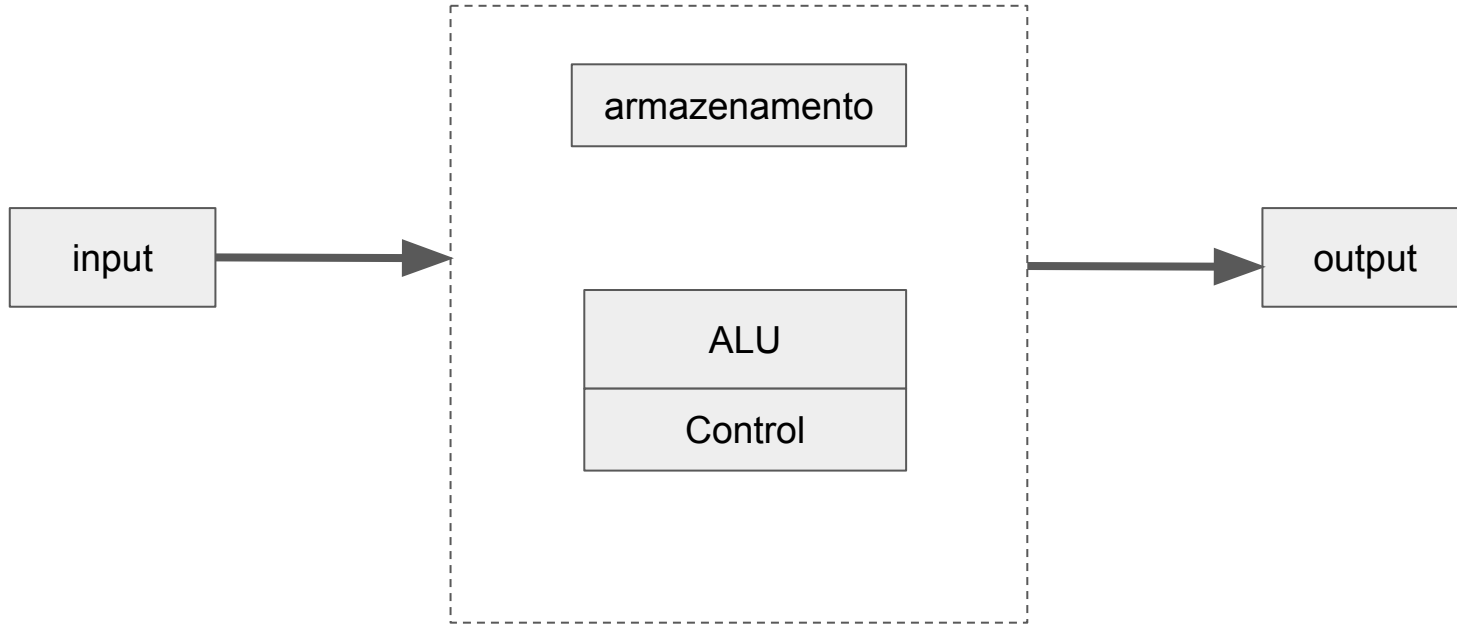
4. Formatos textuais: CSV, JSON e XML;
5. Markdown;
6. HTML e LaTeX.

... Vamos ver!

Aula 1

- <https://blockly.games>
- Edição 2021/2022: <https://epl.di.uminho.pt/~jcr/AULAS/ATP2021/>
- Edição 2022/2023: <https://epl.di.uminho.pt/~jcr/AULAS/ATP2022/>
- Edição corrente: <https://epl.di.uminho.pt/~jcr/AULAS/ATP2023/>

Arquitetura base de um computador



Componentes de hardware de input

- Rato;
- Teclado;
- Caneta;
- Leitor de código de barras;
- Leitor do Cartão do Cidadão;
- ...

Canal: `stdin`

Componentes de hardware de output

- Monitor;
- Impressora;
- Rede;
- ...

Canal: `stdout`, `stderr`

Armazenamento

- Memória principal (RAM);
 - volátil
- Memória secundária: pen-drive, disco, tape, fita magnética, ...
 - persistente

Sistema Operativo

- O que é? Para que serve?
- Como interagimos com ele?
- Exemplos: Windows, Linux, Mac OS, DOS, S36, ...

Representação de informação: bits e bytes

- ASCII
- Unicode: UTF-8 (mostrar no VS Code)

Conceito de variável

- Em Python, uma variável é um nome que é usado para referenciar um valor. As variáveis são usadas para armazenar informações que podem ser usadas posteriormente no programa;
- As variáveis podem ser usadas para armazenar uma variedade de valores, incluindo números, strings, listas, tuplas e dicionários;
- Em Python, as variáveis são tipadas dinamicamente, o que significa que o tipo de uma variável é determinado pelo valor atribuído a ela. Por exemplo, o seguinte código irá declarar duas variáveis, `nome` e `idade`, e atribuir a elas os valores `Pedro` e `25` respectivamente:

```
nome = "Pedro"

idade = 25
```

Tipos de dados





- Carateres
 - Operação de leitura em Python: `input("texto introdutório")`
 - Lê sempre carateres
 - Conversões para outros tipos de dados:
 - `int()`
 -
- Inteiros
- Reais
- Complexos
- Strings*

AulaT1: Conditionals - if

Syntax:

```
if condition:  
    # Statements to execute  
    # if condition is true
```

Python3


```
 # python program to illustrate If statement  
 i = 10  
 if (i > 15):  
    print("10 is less than 15")  
 print("I am Not in if")
```

Aula1: Loops - while

Syntax :

```
while expression:  
    statement(s)
```

Python



```
1  # Python program to illustrate  
2  # while loop  
3  count = 0  
4  while (count < 3):  
5      count = count + 1  
6      print("Hello Geek")  
7
```

Aula Teórica 1:

1. Ambientes e ferramentas:
 - a. Jupiter notebook;
 - b. Visual Studio Code;
 - c. w3Schools tutorial and exercises;
 - d. Python Tutor.
2. Manifesto e sua definição;
3. Tipos de dados em Python:
 - a. inteiros
 - b. reais
 - c. complexos
 - d. strings

Manifesto

- Definir o manifesto em VS Code, explicando o Markdown