

# Projeto de Inteligência Artificial

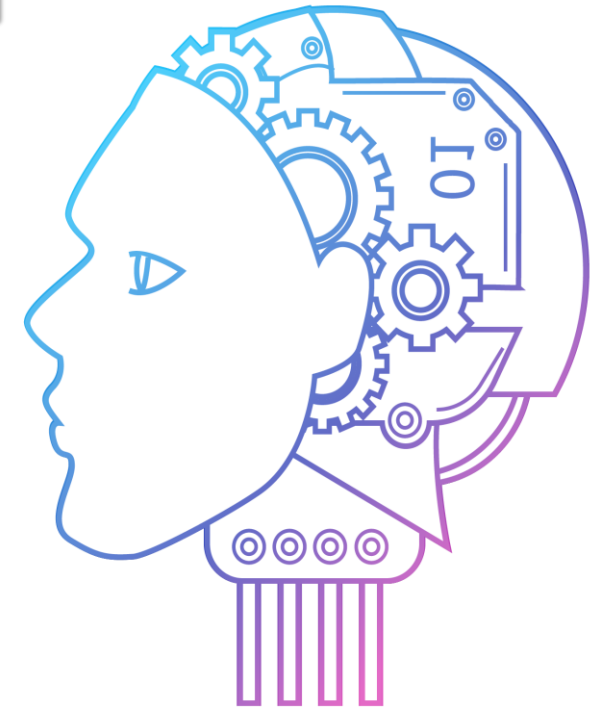
Apresentação do Projeto da Unidade Curricular

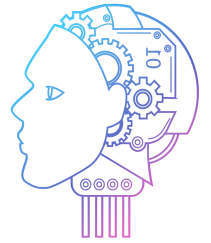
**Orientador:** Professor Doutor Luís Alexandre

**Elaborado por:**

Carlos Martins – Engenharia Informática – 41968

Rui Ferreira – Engenharia Informática - 41064



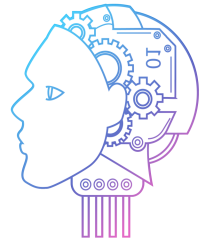


# Pergunta 1: Qual foi a penúltima pessoa que viste?

---

```
for i in range(len(objetos)):
    aux_objetos = objetos[i].split(",")
    if any(s for s in aux_objetos if "enfermeiro" in s or "doente" in s or "medico" in s):
        if list_people[0] != aux_objetos:
            list_people[1] = list_people[0]
            list_people[0] = aux_objetos
```

**Figura 1** – Representação da função que permite visualizar a penúltima pessoa



# Pergunta 2: Em que tipo de sala estás agora?

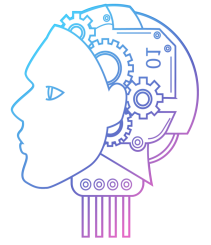
```
def room_type(room):  
    list_aux = []  
    if room in [1, 2, 3, 4]:  
        return "Corredor"  
  
    for i in items_list:  
        if i[0] == room:  
            list_aux.append(i)  
  
    mesa = 0  
    cadeira = 0  
    for i in list_aux:  
        if i[1] == "cama":  
            list_aux.clear()  
            return "Quarto"  
        if i[1] == "mesa":  
            mesa += 1  
        if i[1] == "cadeira":  
            cadeira += 1  
  
    if mesa >= 1 and cadeira >= 1:  
        list_aux.clear()  
        return "Sala Enfermeiro"  
    elif cadeira >= 2:  
        list_aux.clear()  
        return "Sala de espera"  
    else:  
        list_aux.clear()  
        return "Divisao"
```



```
room_name[0] = room_type(current_position(posicao[0], posicao[1]))
```

Figura 3 – Representação da utilização da função *room\_type(room)* na função *work(posicao,bateria,objetos)*


Figura 2 – Representação da função auxiliar que permite classificar o tipo de sala



# Pergunta 6: Quanto tempo achas que falta até ficares sem bateria?

```
if int(battery_aux) != int(bateria):  
    time_pass = time.time() - start_time  
    battery_lost = 100 - int(bateria)  
    battery_remaining = time_remaining(battery_lost, time_pass)  
    battery_aux = bateria  
    start_time = time.time()
```

Figura 4 – Representação Função calcular perdas bateria



```
def time_remaining(bateria, time_used):  
    global times_list  
    times_list.append(time_used)  
    len_times_list = len(times_list)  
    if len_times_list > 3:  
        times_list.pop(0)  
  
    average_times = sum(times_list) / len_times_list  
    result = average_times * (101 - bateria + 1)  
  
    return result
```

Figura 5 – Representação da função auxiliar *time\_remaining(bateria, time\_used)*



# Pergunta 7: Qual a probabilidade de encontrar um livro numa divisão, se já encontraste uma cadeira?

---

```
def resp7():
    room_with_chair = 0
    room_with_chair_and_book = 0
    rooms_visited = 0

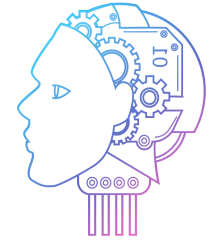
    for room in range(5, 15):
        aux_visited = False
        aux_chair = 0
        aux_book = 0

        for item in items_list:
            if item[0] == room:
                if item[1] == "cadeira":
                    aux_chair += 1
                if item[1] == "livro":
                    aux_book += 1
                aux_visited = True

        if aux_visited == True:
            rooms_visited += 1
        if aux_chair >= 1:
            room_with_chair += 1
        if aux_chair >= 1 and aux_book >= 1:
            room_with_chair_and_book += 1

    if(rooms_visited == 0 or room_with_chair == 0):
        print("Nenhuma divisao com cadeira encontrada")
    else:
        pb_chair = room_with_chair / rooms_visited
        pb_chair_and_book = room_with_chair_and_book / rooms_visited
        res = pb_chair_and_book / pb_chair
        print("A probabilidade é %.2f." % res)
```

Figura 6 – Representação da função que permite calcular a probabilidade



# Pergunta 8:

Se encontrases um enfermeiro numa divisão, qual é a probabilidade de estar lá um doente?

```
def resp8():
    room_with_enf = 0
    room_with_enf_and_doente = 0
    rooms_visited = 0

    for room in range(5, 15):
        aux_visited = False
        aux_enf = 0
        aux_doente = 0

        for item in items_list:
            if item[0] == room:
                if item[1] == "enfermeiro":
                    aux_enf += 1
                if item[1] == "doente":
                    aux_doente += 1
                aux_visited = True

        if aux_visited == True:
            rooms_visited += 1
        if aux_enf >= 1:
            room_with_enf += 1
        if aux_enf >= 1 and aux_doente >= 1:
            room_with_enf_and_doente += 1

    if(rooms_visited == 0 or room_with_enf == 0):
        print("Nenhuma divisao com enfermeiro encontrada")
    else:
        pb_enf = room_with_enf / rooms_visited
        pb_enf_and_doente = room_with_enf_and_doente / rooms_visited
        res = pb_enf_and_doente / pb_enf
        print("A probabilidade é %.2f." % res)
```

Figura 7 – Representação da função que permite calcular a probabilidade