

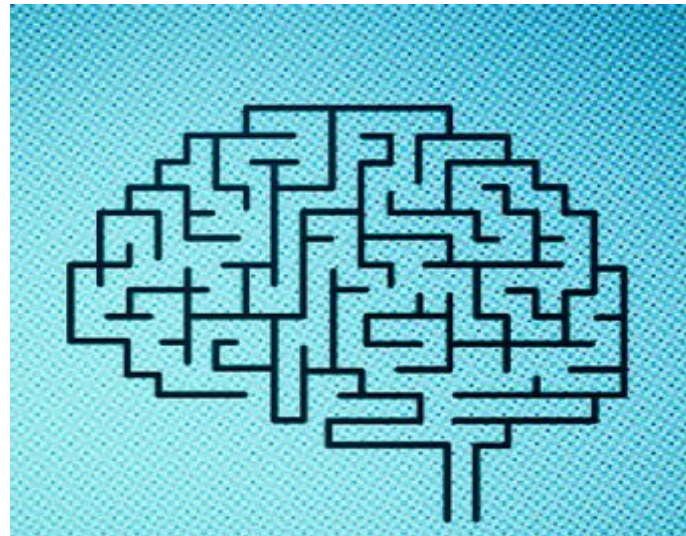
# Guia do Projeto 01

Inteligência Artificial

Licenciatura em Engenharia de Sistemas Informáticos, 2024-25

# Introdução

- O objetivo do projeto é desenvolver um agente inteligente, documentando as etapas de conceção e implementação
- Cada grupo de trabalho deve ser constituído por **2 ou 3 alunos**
- O projeto deve estar disponível numa pasta do Google Drive ou num repositório do Github
  - O código e a documentação devem ser integrados num caderno de notas Jupyter
  - A pasta ou o repositório deve conter todos os ficheiros necessários para executar a solução
  - A pasta ou o repositório deve ser partilhado com o professor
- Um ficheiro ZIP do repositório deve ser submetido no Moodle antes da data limite



# Plataformas colaborativas

- Cada estudante deve ter uma conta na plataforma **GitHub**
  - Um dos membros do grupo deve criar um repositório privado
  - Utilizar a nomenclatura “**IA24\_###**”, em que ### corresponde ao grupo
  - Adicionar os restantes membros do grupo como colaboradores
- O docente deve também ser adicionado ao repositório e ao projeto
- Se o grupo de trabalho decidir utilizar o **Google Colab**
  - Um dos membros do grupo deve criar uma pasta no Google Drive com a nomenclatura “**IA24\_###**”, em que ### corresponde ao grupo
  - A pasta deve ser partilhada com todos os membros do grupo e com o professor.



## Tema — Planeamento de tarefas

# Planeamento de tarefas

- Pretende-se programar todos os trabalhos em tempo e modo de execução para minimizar os atrasos do projeto.
- Cada tarefa faz parte de um projeto
- Um tarefa pode ser executada de diferentes formas
- Cada forma é um **modo de execução** que implica uma duração diferente, mas também a utilização de recursos diferentes.



Source: Roadmunk product roadmap templates

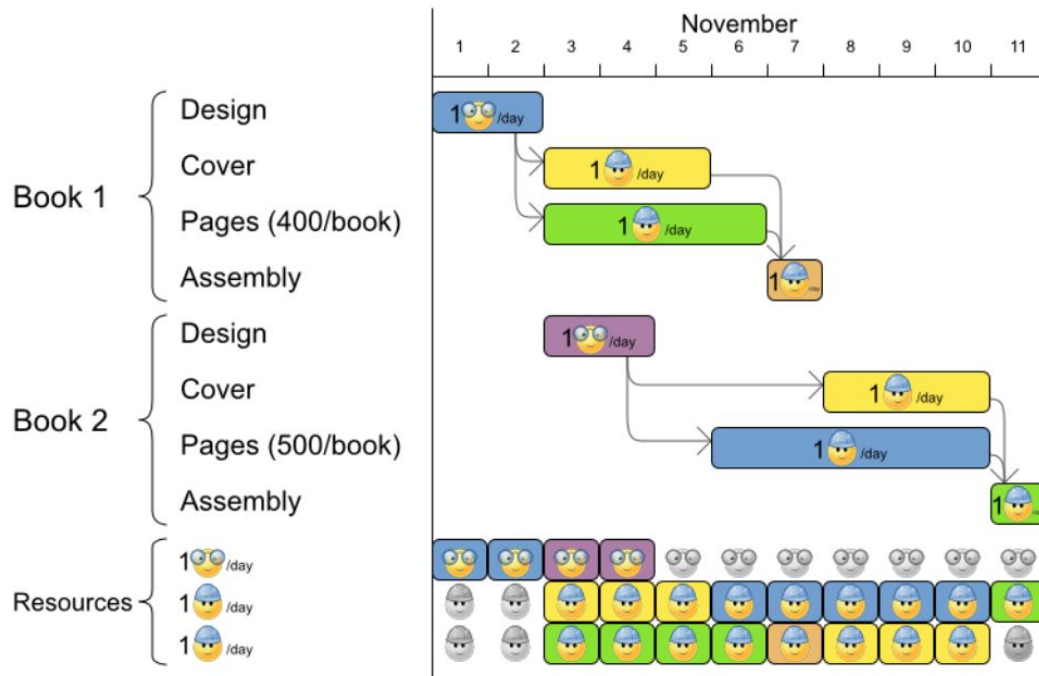
# Tipos de soluções

Um problema de planeamento pode ter muitas soluções de diferentes categorias:

- Uma **solução possível** é qualquer solução, uma atribuição completa, quer quebre ou não um determinado número de restrições. Os problemas de planeamento tendem a ter um número incrivelmente elevado de soluções possíveis, muitas das quais não têm qualquer valor.
- Uma **solução viável** é uma solução que não quebra nenhuma restrição forte. O número de soluções viáveis tende a ser relativo ao número de soluções possíveis. Por vezes, não existem soluções viáveis. Cada solução viável é uma solução possível.
- Uma **solução ótima** é a solução com a pontuação mais elevada. Os problemas de planeamento tendem a ter uma ou poucas soluções ótimas. Existe sempre pelo menos uma solução ótima, mesmo no caso de não existirem soluções viáveis e de a solução ótima não ser viável.
- A **melhor solução encontrada** é a solução com a pontuação mais elevada encontrada por uma implementação num determinado período de tempo. É provável que a melhor solução encontrada seja viável e, com tempo suficiente, é uma solução ótima.

# Exemplo

- Trata-se de uma forma de programação flexível de trabalho
- Este problema pode incluir diferentes tipos de restrições
  - **Restrições fortes** ou rígidas — obrigatórias para obter uma solução válida.
  - **Restrições leves** ou moderadas— preferidas mas não estritamente necessárias, pelo que podem ser violadas em determinadas circunstâncias.
- Estarão disponíveis no Moodle 2 conjuntos de dados
  - um pequeno e simples
  - um maior e mais complexo



Source: Optaplanner User Guide

# Restrições

## Restrições fortes

- Precedência de tarefas — uma tarefa só pode ser iniciada quando todas as suas tarefas predecessoras estiverem concluídas.
- Capacidade de recursos — não utilizar mais recursos do que os disponíveis
  - Os recursos são locais (partilhados entre trabalhos do mesmo projeto) ou globais (partilhados entre todos os trabalhos)
  - Os recursos são renováveis (capacidade disponível por dia) ou não renováveis (capacidade disponível no total)
- Podem ser acrescentadas mais restrições para tornar este problema mais realista
  - Limitar a duração de todo o calendário multi-projeto (forte)
  - Alguns projectos podem ter um prazo (forte)
  - Minimizar a duração (makespan) de cada projeto (leve)





# Jupyter Notebook

# Estrutura do Notebook (1)

- Introdução
  - Estabelecer aqui o contexto e o objetivo do projeto
  - Identificar os colegas de equipa: nome e número do aluno
- Formulação do objetivo
  - Definição do objetivo, possíveis limitações e acções a realizar
- Planear e conceber um agente adequado
  - Formular o problema como um problema de pesquisa
  - Apresentar o algoritmo utilizado (pseudo-código)
  - Destacar as heurísticas aplicadas

# Estrutura do Notebook (2)

- Funcionamento do Agente
  - Fornecer uma solução para um ou mais estados iniciais, i.e., diferentes problemas
  - Utilizar diferentes algoritmos ou definições
  - Efetuar uma análise crítica dos resultados e identificar algumas melhorias futuras para o agente
- Conclusão
  - Elaborar uma conclusão sobre os resultados obtidos, o processo de desenvolvimento e as ferramentas utilizadas.
- A estrutura do caderno deve ser adaptada de acordo com as características de cada projeto.



Obrigado!