

Exame de época normal: rpcw2025-normal

UC: Representação e Processamento de Conhecimento na Web

24 de Maio de 2025, 9h, CP1: sala 1.05

Mestrado em Engenharia Informática (1º ano)

Sinopsis

O objectivo principal deste teste é avaliar os conhecimentos obtidos durante as aulas na especificação e desenvolvimento de ontologias e de aplicações que tiram partido destas.

Antes de começares, lê atentamente até ao fim para ficares com uma percepção do todo que se pretende. Vais ver que tomarás decisões mais acertadas depois de uma leitura completa.

Os resultados finais deverão ser enviados ao docente da seguinte forma:

- Enviar por email para: jcr@di.uminho.pt;
 - Colocar no subject/assunto: **RPCW2025::Normal::Axxxxx**, em que **Axxxxx** corresponde ao número do aluno (nalguns casos **PGxxxxxx**);
 - Enviar ao docente um link do github para um repositório novo criado especificamente para o exame (poderá não ser um repositório, pode ser uma pasta no repositório de RPCW2025) com o seguinte conteúdo (esta preparação poderá valer 1 valor do exame):
 - O repositório ou a pasta no GitHub deverá chamar-se **RPCW2025-Normal**;
 - Dentro do repositório deverá haver um ficheiro, **PR.md**, contendo uma descrição do que fez e como fez, contendo instruções se necessário, para o docente poder replicar os cenários na correção.
-

Vamos extrair conhecimento de uma história

Neste exercício, deverás especificar uma ontologia OWL, podes usar o **Protégé**, para a história abaixo e os requisitos que lhe seguem.

História

Num futuro não muito distante, a cidade de Sapiencia abriga uma das mais avançadas academias de conhecimento do mundo. Lá, os Mentores do Saber preservam séculos de descobertas nas áreas de ciência, arte, tecnologia e filosofia. No entanto, os arquivos estão espalhados em sistemas antigos e incompatíveis — um verdadeiro labirinto digital. Sempre que um novo aprendiz chega, é desafiado a reestruturar parte deste legado.

Certo dia, o Conselho da Academia convoca os estudantes para uma missão especial: construir uma Ontologia do Conhecimento Universal que permita integrar, consultar e raciocinar sobre tudo o que

se sabe na cidade — desde os tipos de conhecimento (como "Filosofia Grega" ou "Programação Funcional") até os seus mestres, obras, conceitos-chave, aplicações práticas e inter-relações. O objetivo é simples: tornar o saber verdadeiramente navegável.

Cabe agora aos novos aprendizes — vocês — definir as classes, propriedades, indivíduos e restrições que tornarão este sonho possível. Que conceitos são fundamentais? Como se relacionam? Quem ensinou o quê? Em que época? E, acima de tudo, como fazer com que uma máquina compreenda o que é o conhecimento?

Orientações/sugestões

Fica aqui uma sugestão para os conceitos iniciais e alguns requisitos para vos orientar na construção da ontologia OWL com base na história da Academia Sapiientia:

Conceitos Iniciais

Classes possíveis:

- TipoDeConhecimento (ex: Filosofia, Matemática, Linguística);
- Disciplina (ex: Filosofia Grega, Álgebra Linear, Programação Funcional);
- Obra (ex: Ética a Nicômaco, Principia Mathematica);
- Mestre (ex: Aristóteles, Ada Lovelace);
- Aprendiz (pode ser um indivíduo atual);
- Conceito (ex: Lógica Formal, Recursividade, Silogismo);
- PeríodoHistorico (ex: Idade Antiga, Século XIX);
- Aplicação (ex: Robótica, Educação, Medicina).

Propriedades sugeridas:

- ensina (Mestre → Disciplina)
- aprende (Aprendiz → Disciplina)
- estáRelacionadoCom (Conceito → Conceito)
- pertenceA (Disciplina → TipoDeConhecimento)
- explica (Obra → Conceito)
- foiEscritoPor (Obra → Mestre)
- éEstudadoEm (Conceito → Disciplina)
- surgeEm (Conceito → PeríodoHistorico)
- temAplicaçãoEm (Conceito → Aplicação)

Requisitos

Estas são perguntas que a ontologia deve ser capaz de responder se bem modelada:

- Quais são os conceitos ensinados por um determinado mestre?
- Que obras explicam o conceito de "Recursividade"?
- Em que disciplinas se estuda "Silogismo"?
- Qual é a aplicação prática dos conhecimentos em "Lógica Formal"?
- Quais disciplinas pertencem ao tipo de conhecimento "Matemática"?
- Que mestres ensinaram no século XIX?

- Que aprendizes estudam "Programação Funcional"?
- Como se relacionam os conceitos de "Função" e "Composição"?

Especificação e implementação

Vais agora, depois de ler a história, as sugestões e os requisitos, especificar uma ontologia OWL.

De início, irás especificar a estrutura da tua ontologia: classes e propriedades. A seguir, irás povoar esta estrutura com a informação de um conjunto de datasets em JSON disponibilizados juntamente com este enunciado.

A Ontologia: estrutura base (2 val = 0.5 + 0.5 + 0.5 + 0.5)

- Especifica agora a ontologia que dê suporte ao conhecimento nesta história:
 1. Define as Classes;
 2. Define os atributos de cada Classe (**data properties**);
 3. Define as relações que se podem estabelecer entre indivíduos da mesma classe ou de classes diferentes (**object properties**);
 4. Grava o teu trabalho num ficheiro TTL: **sapientia_base.ttl**.

A Ontologia: povoamento/criação de indivíduos (3 val = 0.5 cada alínea)

- Povoas agora a tua ontologia base com a informação presente nos datasets fornecidos atendendo aos seguintes pressupostos:
 - A informação poderá estar incompleta: por exemplo, um conceito referido numa disciplina poderá não constar da lista de conceitos; Poderá haver casos semelhantes, referentes a outros elementos de informação; Nestes casos, os indivíduos devem ser criados na mesma com a informação disponível. Exemplo: uma disciplina refere os conceitos A e B; A está no dataset de conceitos e B não está; na ontologia devem ser criados dois indivíduos A e B, A com a informação constante no dataset e B apenas com o nome;
 - No fim, todas as relações entre indivíduos, presentes nos datasets, terão de estar materializadas em relações entre indivíduos na ontologia, sendo necessário criar vários indivíduos com informação mínima ou incompleta na ontologia.

5. **conceitos.json** - estrutura:

```
{
  "conceitos": [
    {
      "nome": "Lógica Formal",
      "aplicações": ["Inteligência Artificial", "Programação",
"Matemática"],
      "períodoHistórico": "Idade Antiga",
      "conceitosRelacionados": ["Lógica Matemática", "Teoria da
Computação", "Inteligência Artificial"]
    }, ...
  ]
}
```

6. `disciplinas.json` - estrutura:

```
{
  "disciplinas": [
    {
      "nome": "Filosofia",
      "tiposDeConhecimento": ["Humanidades", "Filosofia"],
      "conceitos": ["Ética", "Metafísica", "Epistemologia", "Filosofia da Mente"]
    }, ...
  ]
}
```

7. `mestres.json` - estrutura:

```
{
  "mestres": [
    {
      "nome": "Aristóteles",
      "períodoHistórico": "Idade Antiga",
      "disciplinas": ["Filosofia", "Lógica", "Ética"]
    }, ...
  ]
}
```

8. `obras.json` - estrutura:

```
{
  "obras": [
    {
      "titulo": "Ética a Nicômaco",
      "autor": "Aristóteles",
      "conceitos": ["Ética", "Virtude", "Felicidade"]
    }, ...
  ]
}
```

9. `pgxxxxx.json` ou `exxxxxx.json`:

Este dataset correspondente aos aprendizes está numa pasta designada `specific_datasets`. Deverás utilizar o dataset correspondente ao teu número de identificação.

```
[
  {
```

```

    "nome": "Sávio Arinto",
    "idade": 63,
    "disciplinas": [
        "Pedagogia",
        "Biotecnologia",
        "Física Aplicada",
        "Genética",
        "Literatura",
        "Línguas",
        "Engenharia Biomédica"
    ]
}, ...
]

```

10. Grava o teu trabalho num ficheiro TTL: `sapientia_ind.ttl`.

Queries simples (4 val.: 0.5 cada alínea)

- É importante que saibas que as queries que vais desenvolver a partir daqui serão testadas no GraphDB com a ontologia que criaste previamente e pela ordem em que aqui são pedidas.
11. Cria um repositório no teu GraphDB de nome "sapientia" e importa a ontologia criada;
- Especifica as seguintes queries SPARQL no GraphDB e verifica os resultados (coloca estas queries num ficheiro "sparql.txt", cada query deverá estar completa tendo os prefixos necessários e deverá estar identificada com a alínea a que responde):
12. Quais são os conceitos ensinados por um determinado mestre (arbitra um)?
13. Que obras explicam o conceito de "Relatividade"?
14. Em que disciplinas se estudam "Ciências Sociais"?
15. Qual é a aplicação prática dos conhecimentos em "Lógica Formal"?
16. Quais disciplinas pertencem ao tipo de conhecimento "Ciências Exatas"?
17. Que mestres ensinaram no século XIX?
18. Que aprendizes estudam "Programação"?

Cálculo de distribuições (7 val.)

- Vais continuar a especificar queries, agora um pouco mais complexas:
19. (1 val) Qual a distribuição de aprendizes por disciplina? (Resultado será uma lista de pares: Disciplina, número de alunos)
20. (1 val) Qual a distribuição de conceitos por período histórico? (Resultado será uma lista de pares: Período, número de conceitos)
21. (1.5 val) Quais os títulos das obras associadas a cada conceito? (Resultado será uma lista de pares: Conceito, Lista de títulos de obras)
22. (1.5 val) Quais os aprendizes que frequentam cada uma das disciplinas? (Resultado será uma lista de pares: Disciplina, Lista dos nomes dos aprendizes que frequentam a disciplina)

23. (2 val) Qual a distribuição de aprendizes por escalões etários? (Resultado será uma lista de pares: Escalão etário, número aprendizes nesse escalão); Considera os seguintes escalões: [17, 34], [35, 44], [45, 54], [55, 64], [65, 74], [75, 100].

Inferência de novo conhecimento (4 val.)

- Com a utilização de queries CONSTRUCT e queries INSERT, vais calcular novo conhecimento e nalguns casos acrescentar esse conhecimento à ontologia:
24. (0.5 val) Cria uma nova relação na ontologia, **estudaCom**, que relaciona um aluno com um mestre;
25. (1.5 val) Cria uma query CONSTRUCT para calcular os novos triplos da nova relação **estudaCom**; Altera a query para uma query INSERT e acrescenta esses triplos à ontologia;
26. (0.5 val) Cria uma nova relação na ontologia, **dáBasesPara**, que relaciona uma disciplina com uma aplicação;
27. (1.5 val) Cria uma query CONSTRUCT para calcular os novos triplos da nova relação **dáBasesPara**; Altera a query para uma query INSERT e acrescenta esses triplos à ontologia.
-

Para terminar:

28. Coloca na pasta do Git criada, a ontologia base, a ontologia povoada, o ficheiro com as queries e as scripts de migração (caso não tenhas usado scripts, explica como fizeste a migração).
-

Bom trabalho e boa sorte jcr