



11.º ANO | ENSINO SECUNDÁRIO

COMPUTAÇÃO

INTRODUÇÃO

A disciplina de Computação integra a componente específica do Curso Científico-Humanísticos de Ciências e Tecnologias do ensino secundário, contribuindo para as diferentes áreas de competências do Perfil dos Alunos à Saída da Escolaridade Obrigatória (PA).

Sendo uma disciplina da componente da formação específica e bienal que visa proporcionar formação científica consistente no domínio do respetivo curso.

Neste documento enunciam-se as Aprendizagens Essenciais (conjunto de conhecimentos, capacidades e atitudes) na área de Computação e Informática. As aprendizagens essenciais têm como base o ciclo de desenvolvimento de *software*, bem como a metodologia de desenvolvimento de projetos de programação, algoritmos, estrutura de dados, fundamentos de programação, análise de sistema, base de dados e desenvolvimento de aplicações para a *web*. Os alunos devem conseguir analisar e

compreender problemas de forma tomar decisões fundamentadas em cada um dos domínios.

As Aprendizagens Essenciais de Computação organizam-se em oito domínios (D1 a D8). Estes oito domínios organizam-se em Aprendizagens Essenciais Transversais (AET) que são integradas e necessárias no processo de tomada de decisão e na concretização das Aprendizagens Essenciais por Domínio Específico (AEDE).

Aprendizagens Essenciais Transversais (AET)

D1 Ciclo de desenvolvimento de *software*

D2 Projeto de programação

Aprendizagens Essenciais por Domínio Específico (AEDE)

D3 Algoritmia

D4 Estruturas de dados

D5 Fundamentos de programação

D6 Análise de sistemas

D7 Base de dados

D8 Desenvolvimentos de aplicações para *web* com base de dados

No domínio sobre o ciclo de desenvolvimento de *software* pretende-se que os alunos o conheçam e o apliquem nas atividades e projetos dos domínios de fundamentos de programação e desenvolvimento de aplicações para a *web*, independentemente, das linguagens de programação que venham a utilizar.

O domínio de projetos de programação tem como objetivo, que os alunos conheçam metodologias e técnicas de gestão e desenvolvimento de projetos de programação e que as utilizem nos diferentes domínios específicos da disciplina.

Relativamente ao domínio de algoritmia pretende-se que os alunos conheçam e criem algoritmos para completar tarefas e no domínio de estrutura de dados que identifiquem os diferentes tipos de estrutura de dados.

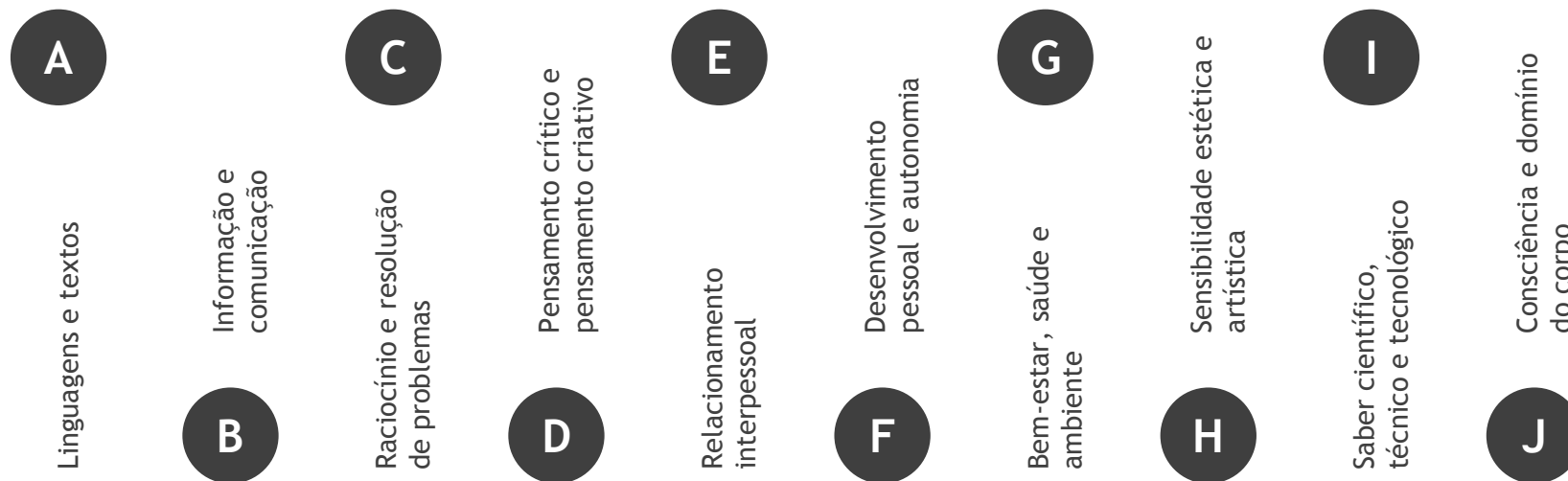
No que se refere ao domínio de fundamentos de programação privilegia-se o paradigma de programação estruturada e programação orientada por objetos (poo).

No âmbito do domínio de análise de sistemas, pretende-se que os alunos usem modelos de entidades e relações e modelos físicos de dados, antes de iniciar o domínio das bases de dados, no âmbito do qual se pretende que construam bases de dados relacionais.

O domínio sobre desenvolvimento de aplicações para a *web* com bases de dados tem como objetivo programar aplicações para a *web* integrando as bases de dados.

A lógica que deve prevalecer na implementação da disciplina, será a do desenvolvimento de desafios, problemas ou projetos, recomendando-se um trabalho conjunto e em simultâneo para as aprendizagens de diferentes domínios, bem como a articulação com outras áreas disciplinares e a colaboração com serviços e projetos da escola, com a família e com instituições regionais, nacionais ou internacionais.

ÁREAS DE
COMPETÊNCIAS
DO PERFIL DOS
ALUNOS (ACPA)



OPERACIONALIZAÇÃO DAS APRENDIZAGENS ESSENCIAIS (AE)

ORGANIZADOR Domínio	AE: CONHECIMENTOS, CAPACIDADES E ATITUDES O aluno deve ficar capaz de:	AE: AÇÕES ESTRATÉGICAS DE ENSINO ORIENTADAS PARA O PERFIL DOS ALUNOS (Exemplos de ações a desenvolver na disciplina)	DESCRITORES DO PERFIL DOS ALUNOS
------------------------	---	---	----------------------------------

APRENDIZAGENS ESSENCIAIS TRANSVERSAIS (AET)

Ciclo de desenvolvimento de software

- Implementar o ciclo de desenvolvimento de *software*:
- Fase de requisitos: Fazer o levantamento dos requisitos mínimos, estudar a viabilidade e definir o modelo a ser usado;
 - Fase de projeto: Envolve atividades de conceção, especificação, *design* da interface, prototipagem e *design* da arquitetura;
 - Fase de implementação: Tradução para uma linguagem de programação das funcionalidades definidas durante as fases anteriores;
 - Fase de testes: realização de testes no que foi desenvolvido de acordo com os

- Promover estratégias que envolvam aquisição de conhecimento, informação e outros saberes, relativos aos conteúdos das AE, que impliquem:**
- ser rigoroso, articular e usar de forma consistente conhecimentos para criar algoritmos a fim de resolver problemas complexos;
 - seleccionar informação pertinente e ajustada ao problema a resolver e/ou à tarefa ou ao projeto a desenvolver;
 - organizar de modo sistemático algoritmos, representando-os através de fluxogramas e/ou pseudocódigo;
 - analisar problemas complexos, factos, teorias ou situações reais, identificando os seus elementos ou dados com vista à posterior modelação em computador;
 - estabelecer relações intra e interdisciplinares.

- Promover estratégias que envolvam a criatividade dos alunos:**
- desenvolver novos programas ou modificar programas existentes para adicionar novos recursos e comportamentos usando diferentes formas de entradas e saídas (por exemplo, entradas como sensores, cliques do mouse e conjuntos de dados e saídas como texto, gráficos e sons);
 - criar modelos computacionais que simulem sistemas do mundo real (por exemplo, ecossistemas, epidemias, disseminação de ideias);
 - projetar, desenvolver e implementar um artefacto de computação

Conhecedor/ sabedor/ culto/ informado
(A, B, I)

Criativo
(A, C, D, H)

<p>requisitos;</p> <p>Fase de produção: Desenvolver o produto final.</p>	<p>que responda a um evento (por exemplo, um <i>robot</i> que responde a um sensor, uma aplicação móvel que responde a uma mensagem de texto);</p> <ul style="list-style-type: none"> - usar técnicas de pesquisa e <i>design</i> centradas no utilizador (por exemplo, pesquisas, entrevistas) para criar soluções de <i>software</i>; - usar técnicas da área de investigação <i>user-centered design</i> (UCD) para criar soluções de <i>software</i> ajustados aos potenciais utilizadores; - integrar técnicas, métodos e processos matemáticos apropriados à criação de artefactos de computação. 	
<p>Projeto de Programação</p> <p>Identificar e selecionar as linguagens de programação adequadas.</p> <p>Utilizar as principais metodologias e técnicas de desenvolvimento de <i>software</i>.</p> <p>Selecionar as técnicas de programação informática em função dos produtos a desenvolver.</p> <p>Criar testes aplicativos utilizando diferentes metodologias.</p> <p>Aplicar os procedimentos de qualidade e segurança no desenvolvimento de <i>software</i>.</p> <p>Conhecer e utilizar as recomendações relativas à acessibilidade no desenho de <i>software</i>.</p>	<p>Promover estratégias que desenvolvam o pensamento crítico e analítico dos alunos, incidindo em:</p> <ul style="list-style-type: none"> - mobilizar o discurso argumentativo (expressar uma tomada de posição, pensar e apresentar argumentos e contra-argumentos, rebater os contra-argumentos); - organizar debates que requeiram sustentação de afirmações, elaboração de opiniões ou análises de factos ou dados; - discutir conceitos ou factos numa perspetiva disciplinar e interdisciplinar, incluindo conhecimento disciplinar específico; - analisar textos com diferentes pontos de vista, confrontando argumentos para encontrar semelhanças, diferenças, consistência interna; - problematizar situações; - analisar factos, teorias, situações, identificando os seus elementos ou dados, em particular numa perspetiva disciplinar e interdisciplinar. 	<p>Crítico/Analítico (A, B, C, D, E, I)</p>
<p>APRENDIZAGENS ESSENCIAIS POR DOMÍNIO ESPECÍFICO (AEDE)</p>		
<p>Algoritmia</p> <p>Reconhecer o conceito de algoritmo.</p> <p>Modelar processos diários criando e seguindo algoritmos para completar tarefas.</p>	<p>Promover estratégias que envolvam por parte do aluno:</p> <ul style="list-style-type: none"> - executar tarefas de pesquisa sustentada por critérios, com autonomia progressiva; - incentivar a procura e aprofundamento de informação; - recolher dados e opiniões para análise e modelação de temáticas em estudo. 	<p>Indagador/ Investigador (B, C, D, F, H, I)</p>

	<p>Usar fluxogramas e/ou pseudocódigo para tratar de problemas complexos como algoritmos.</p>	<p>Promover estratégias que requeiram/induzam por parte do aluno:</p> <ul style="list-style-type: none"> - aceitar e/ou argumentar pontos de vista diferentes; - promover estratégias que induzam respeito por diferenças de características, crenças ou opiniões; - confrontar ideias e perspetivas distintas na abordagem a um dado problema e ou forma de o resolver, tendo em conta, diferentes perspetivas de cada contexto ou sejam de incidência local, nacional ou global. 	<p>Respeitador da diferença/ do outro (A, B, D, E, F, H)</p>
<p>Estrutura de Dados</p>	<p>Reconhecer os diferentes tipos de estruturas de dados.</p> <p>Identificar algoritmos de manipulação de estruturas de listas lineares e matriciais uni e multidimensional.</p>	<p>Promover estratégias que envolvam por parte do aluno:</p> <ul style="list-style-type: none"> - realizar tarefas de síntese; - realizar tarefas de planeamento, de revisão e de monitorização; - ser organizado (por exemplo, criar planos com as etapas de determinado projeto e respetiva calendarização, gerir uma agenda da turma, fazer registos individuais do trabalho realizado); - criar, estruturar e manter atualizado um <i>eportefolio</i> da equipa e/ou individual de acordo com critérios e objetivos definidos; - realizar trabalho autónomo com o apoio do professor à sua concretização, identificando quais os obstáculos e formas de os ultrapassar. 	<p>Sistematizador/ organizador (A, B, C, D, F, I)</p>
<p>Fundamentos de programação</p>	<p>Paradigma: programação estruturada e programação orientada por objetos (POO).</p> <p>Identificar classes, subclasses, objetos e hierarquias e princípios subjacentes de abstração, encapsulação, herança e polimorfismo.</p> <p>Identificar e realizar operações com diferentes tipos de dados, variáveis e operadores.</p> <p>Utilizar estruturas de controlo de seleção e execução.</p> <p>Utilizar funções/módulos em programas.</p> <p>Executar operações com estruturas de dados.</p> <p>Utilizar escrita e leitura de ficheiros.</p>	<p>Promover estratégias que impliquem por parte do aluno:</p> <ul style="list-style-type: none"> - questionar uma situação; - organizar questões para terceiros, sobre temáticas abordadas ou a abordar; - interrogar o seu próprio conhecimento prévio. <p>Promover estratégias que impliquem por parte do aluno:</p> <ul style="list-style-type: none"> - desencadear ações de comunicação uni e bidirecional; - desencadear ações de resposta, apresentação, iniciativa; - desencadear ações de questionamento organizado. <p>Promover estratégias envolvendo tarefas em que, com base em critérios, se oriente o aluno para:</p> <ul style="list-style-type: none"> - identificar pontos fracos e fortes das suas aprendizagens; 	<p>Questionador (A, B, C, D, E, F, I)</p> <p>Comunicador (A, B, D, E, H, I)</p> <p>Autoavaliador (transversal às áreas)</p>

	Identificar erros, validações e correções.	- descrever processos de pensamento usados durante a realização de uma tarefa ou abordagem de um problema; - considerar o <i>feedback</i> dos pares para melhoria ou aprofundamento de saberes; - reorientar o seu trabalho a partir da explicitação de <i>feedback</i> do professor e/ou especialistas da área, individualmente ou em equipa.	
Análise de Sistemas	Conhecer o conceito de análise e de sistema de informação. Usar modelos de entidades e relações. Usar modelos físicos de dados.	Promover estratégias que criem oportunidades para o aluno: - colaborar com outros colegas (preferencialmente em equipa) e apoiar terceiros em tarefas; - fornecer <i>feedback</i> para melhoria ou aperfeiçoamento de um produto de <i>software</i> ;	Participativo/ colaborador (B, C, D, E, F, H, I)
Base de Dados	Identificar bases de dados relacionais. Consultar dados armazenados dentro da base de dados. Criar tabelas, inserir, alterar e apagar dados. Analisar e gerir base de dados, de acordo com as técnicas de análise de sistemas de informação e utilizando a linguagem de programação adequada.	- obter <i>feedback</i> de especialistas para melhoria ou aprofundamento de um produto de <i>software</i> ; - demonstrar como a colaboração diversificada afeta o <i>design</i> e o desenvolvimento de produtos de <i>software</i> (por exemplo, discutir exemplos reais de produtos que foram aperfeiçoados por meio de uma equipa de projeto diversificada e/ou refletindo sobre a experiência de desenvolvimento levada a cabo pela sua própria equipa); - projetar e desenvolver um artefacto de <i>software</i> trabalhando em equipa.	
Desenvolvimentos de aplicações para web com base de dados	Identificar princípios básicos associados à programação para a <i>Web</i> . Programar aplicações cliente-servidor e <i>Web</i> . Estruturar conteúdos na <i>Web</i> . Aplicar estilos a conteúdos <i>Web</i> .	Promover estratégias e modos de organização das tarefas que impliquem por parte do aluno: - assumir de responsabilidades adequadas ao que lhe for pedido; - organizar e realizar autonomamente tarefas; - assumir e cumprir compromissos, contratualizar tarefas; - apresentar trabalhos com auto e heteroavaliação; - dar conta a outros do cumprimento de tarefas e funções que assumiu.	Responsável/ autónomo (D, E, F, G)
		Promover estratégias que induzam: - desencadear ações solidárias para com outros nas tarefas de	Cuidador de si e do outro

Técnicas de armazenamento de dados na *web*
Aplicar regras e protocolos de segurança e proteção dos dados na *web*.

aprendizagem ou na sua organização/atividades de entreajuda;
- analisar e refletir sobre o seu posicionamento perante situações dilemáticas de ajuda a outros e de proteção de si;
- estar disponível para o autoaperfeiçoamento.

(D, E, F, G)