**Etapas da Pesquisa Científica**

As **etapas da pesquisa científica** são um conjunto de passos organizados que orientam a construção do conhecimento de forma lógica, crítica e sistemática. Essas etapas ajudam o pesquisador a transformar uma dúvida ou problema em conhecimento validado.

**Etapas da Pesquisa Científica**

Aqui estão as principais etapas, adaptadas para uso em **Ciência da Computação** (mas válidas em qualquer área):

**1. Escolha do Tema**

* **O que é:** Definir um assunto de interesse pessoal ou acadêmico.
* **Exemplo:** “Segurança em aplicações web”, “Chatbots com IA”, “Gamificação no ensino de lógica”.

**2. Delimitação do Problema**

* **O que é:** Focar em um aspecto específico do tema.
* **Exemplo:** “Como evitar ataques de injeção SQL em formulários PHP?”

**3. Formulação de Hipóteses (se aplicável)**

* **O que é:** Suposição inicial a ser testada com dados.
* **Exemplo:** “A implementação de CAPTCHA reduz em 90% o envio de spam em formulários.”

**4. Objetivos da Pesquisa**

* **Geral:** Descreve o que se deseja alcançar.
* **Específicos:** Etapas para alcançar o objetivo geral.
* **Exemplo:**
  + Geral: “Investigar soluções contra ataques SQL.”
  + Específicos:
    - Levantar as vulnerabilidades mais comuns
    - Testar ferramentas de proteção
    - Comparar desempenho com e sem proteção

**5. Justificativa**

* **O que é:** Explicar a importância e relevância do estudo.
* **Exemplo:** “Falhas de segurança afetam a integridade de sistemas e dados de usuários.”

**6. Revisão da Literatura**

* **O que é:** Levantamento de autores, teorias e pesquisas anteriores.
* **Exemplo:** Artigos sobre OWASP Top 10, estudos de caso sobre segurança web.

**7. Definição da Metodologia**

* **O que é:** Escolher o método científico e os procedimentos de coleta e análise de dados.
* **Exemplo:** Pesquisa aplicada, com método hipotético-dedutivo, testes de vulnerabilidade em três sistemas diferentes.

**8. Coleta de Dados**

* **O que é:** Aplicar questionários, testes, simulações, entrevistas etc.
* **Exemplo:** Rodar scripts de teste em sistemas web, coletar logs de acesso, medir desempenho com e sem proteção.

**9. Análise e Interpretação dos Dados**

* **O que é:** Organizar os dados e verificar se confirmam ou não as hipóteses.
* **Exemplo:** Mostrar gráficos de redução de ataques após aplicar medidas.

**10. Conclusões e Resultados**

* **O que é:** Retomar os objetivos e apresentar os resultados alcançados, com ou sem confirmação da hipótese.

**11. Referências**

* **O que é:** Lista completa das fontes utilizadas, conforme normas acadêmicas (ABNT, APA etc).

**12. Anexos e Apêndices (opcional)**

* **O que é:** Códigos, tabelas, formulários ou materiais complementares.

**Atividade de Fixação – Etapas da Pesquisa Científica**

**Objetivo:**

Fixar o entendimento das etapas do processo de pesquisa científica por meio de aplicação prática em temas de Ciência da Computação.

**Parte 1 – Perguntas Objetivas (escolha a alternativa correta)**

**1. Qual das alternativas representa a primeira etapa de uma pesquisa científica?**

A) Aplicação de questionário  
B) Coleta de dados  
C) Escolha do tema  
D) Teste de hipóteses

**2. A etapa de “Revisão da Literatura” tem como objetivo:**

A) Coletar os dados dos usuários do sistema  
B) Verificar se o tema escolhido está na moda  
C) Estudar trabalhos anteriores e identificar lacunas de pesquisa  
D) Criar a interface do sistema

**3. A “Justificativa” de uma pesquisa responde a qual das perguntas abaixo?**

A) O que será feito?  
B) Como será feito?  
C) Por que este estudo é importante?  
D) Quem vai responder o formulário?

**4. Qual etapa da pesquisa consiste em transformar uma dúvida em algo que possa ser testado?**

A) Delimitação do problema  
B) Formulação da hipótese  
C) Objetivos específicos  
D) Metodologia

**Parte 2 – Perguntas Abertas (respostas escritas)**

**5. Dê um exemplo de tema de pesquisa na área de Ciência da Computação.**

**6. A partir do tema acima, escreva uma possível delimitação do problema.**

**7. Dê um exemplo de objetivo geral e dois objetivos específicos com base no problema anterior.**

* Geral:
* Específicos:

**8. Escreva uma justificativa curta para a pesquisa anterior.**

**9. Qual método científico seria mais adequado para essa pesquisa e por quê?**

**Parte 3 – Resumo Final (fixação em grupo ou individual)**

**10. Liste as principais etapas da pesquisa científica na ordem correta.**

**Modelo de Plano de Pesquisa Científica – Exemplo Preenchido**

**Título do Projeto:**

Detecção Automatizada de Plágio em Códigos Fonte Utilizando Algoritmos de Aprendizado de Máquina

**1. Tema:**

Plágio acadêmico em programação e detecção automática com Inteligência Artificial.

**2. Problema de Pesquisa:**

Como identificar automaticamente cópias de códigos fonte escritos em Python e Java, utilizando algoritmos de aprendizado de máquina?

**3. Hipótese (se aplicável):**

A aplicação de algoritmos de machine learning pode identificar plágios em códigos com mais de 85% de precisão.

**4. Objetivos:**

**Geral:**

Desenvolver uma ferramenta capaz de detectar plágio em códigos de programação por meio de técnicas de inteligência artificial.

**Específicos:**

* Levantar os principais métodos de detecção de plágio em códigos-fonte.
* Coletar um banco de dados com códigos plagiados e originais.
* Treinar um algoritmo de machine learning para detectar similaridades.
* Avaliar a precisão e desempenho do sistema desenvolvido.

**5. Justificativa:**

O plágio em trabalhos de programação compromete a avaliação e a aprendizagem dos alunos. Métodos manuais de verificação são ineficientes. Uma solução baseada em IA pode auxiliar professores na análise e promover uma educação mais ética e eficiente.

**6. Fundamentação Teórica:**

* Estudos sobre plágio e ética na programação.
* Algoritmos de detecção de similaridade (Jaccard, Levenshtein, etc).
* Modelos de aprendizado supervisionado.
* Ferramentas como MOSS, JPlag, Turnitin.

**7. Metodologia:**

* **Tipo de Pesquisa:** Aplicada, exploratória e quantitativa.
* **Método Científico:** Hipotético-dedutivo.
* **Procedimentos Técnicos:** Estudo de caso, simulação, teste de software.
* **Coleta de Dados:** Conjunto de códigos originais e plagiados coletados de exercícios acadêmicos.
* **Ferramentas:** Python, Jupyter Notebook, scikit-learn.

**8. Cronograma (resumo por mês – pode ser detalhado em anexo):**

| **Etapa** | **Mês 1** | **Mês 2** | **Mês 3** | **Mês 4** |
| --- | --- | --- | --- | --- |
| Revisão bibliográfica | ✅ |  |  |  |
| Coleta de dados |  | ✅ |  |  |
| Desenvolvimento do modelo |  | ✅ | ✅ |  |
| Testes e análises |  |  | ✅ | ✅ |
| Escrita e apresentação final |  |  |  | ✅ |

**9. Referências (exemplos):**

* WATSON, R.; LEE, A. *Detecting Software Plagiarism Using Neural Networks*. IEEE Transactions, 2021.
* GUSMAO, J. *Ética e Códigos na Computação*. Ed. Campus, 2020.
* Documentação oficial do scikit-learn e Python.

**Modelo de Plano de Pesquisa Científica – Exemplo 2**

**Título do Projeto:**

Análise da Eficiência de Algoritmos de Ordenação em Grandes Volumes de Dados Utilizando Python

**1. Tema:**

Desempenho de algoritmos de ordenação em diferentes cenários de volume de dados.

**2. Problema de Pesquisa:**

Qual algoritmo de ordenação apresenta melhor desempenho ao lidar com conjuntos de dados grandes (acima de 1 milhão de registros) em Python?

**3. Hipótese:**

Algoritmos híbridos como o Timsort (padrão do Python) são mais eficientes para conjuntos de dados grandes e variados do que algoritmos simples como Bubble Sort ou Insertion Sort.

**4. Objetivos:**

**Geral:**

Avaliar o desempenho de diferentes algoritmos de ordenação em grandes volumes de dados utilizando a linguagem Python.

**Específicos:**

* Implementar algoritmos clássicos de ordenação: Bubble Sort, Merge Sort, Quick Sort, Timsort.
* Gerar conjuntos de dados com diferentes tamanhos e características.
* Medir e comparar o tempo de execução e uso de memória.
* Apresentar os resultados por meio de gráficos e análises.

**5. Justificativa:**

A ordenação de dados é uma etapa essencial em muitos sistemas e aplicações. Com a crescente demanda por manipulação de grandes volumes de dados, conhecer o desempenho dos algoritmos ajuda no desenvolvimento de sistemas mais eficientes e escaláveis.

**6. Fundamentação Teórica:**

* Conceitos de algoritmos e estruturas de dados.
* Complexidade de algoritmos (notação Big O).
* Estudos de caso sobre ordenação em bancos de dados e aplicações web.
* Comparativos empíricos de performance entre algoritmos.

**7. Metodologia:**

* **Tipo de Pesquisa:** Aplicada, experimental, quantitativa.
* **Método Científico:** Hipotético-dedutivo.
* **Procedimentos Técnicos:** Simulações computacionais com geração e análise de dados.
* **Ferramentas:** Python (NumPy, time, matplotlib), Visual Studio Code.

**8. Cronograma (4 meses):**

| **Etapa** | **Mês 1** | **Mês 2** | **Mês 3** | **Mês 4** |
| --- | --- | --- | --- | --- |
| Pesquisa teórica | ✅ |  |  |  |
| Implementação dos algoritmos |  | ✅ |  |  |
| Testes com dados simulados |  | ✅ | ✅ |  |
| Análise e apresentação |  |  |  | ✅ |

**9. Referências:**

* CORMEN, T. H. et al. *Algoritmos: Teoria e Prática*. Campus, 2002.
* SEDGEWICK, R. *Algorithms in Python*. Addison-Wesley, 2019.
* Documentação oficial do Python e NumPy.

**Modelo de Plano de Pesquisa Científica – Exemplo 3**

**Título do Projeto:**

Reconhecimento de Dígitos Manuscritos Usando Redes Neurais Artificiais com TensorFlow

**1. Tema:**

Aplicação de redes neurais artificiais para reconhecimento de padrões visuais.

**2. Problema de Pesquisa:**

Como uma rede neural artificial treinada com o banco de dados MNIST pode ser usada para reconhecer com precisão imagens de dígitos manuscritos?

**3. Hipótese:**

Uma rede neural simples (feedforward) com uma camada oculta é capaz de atingir pelo menos 90% de acurácia no reconhecimento de dígitos manuscritos no conjunto de dados MNIST.

**4. Objetivos:**

**Geral:**

Investigar o uso de redes neurais artificiais para reconhecimento de imagens com o auxílio da biblioteca TensorFlow.

**Específicos:**

* Compreender o funcionamento das redes neurais artificiais.
* Preparar o conjunto de dados MNIST para treinamento e teste.
* Implementar uma rede neural básica em Python usando TensorFlow.
* Avaliar a acurácia e desempenho da rede com métricas adequadas.

**5. Justificativa:**

O reconhecimento de padrões visuais é fundamental em diversas aplicações modernas (OCR, carros autônomos, biometria). Essa pesquisa introduz conceitos de IA de forma prática e acessível, conectando teoria e prática em um problema clássico e didático.

**6. Fundamentação Teórica:**

* Conceitos básicos de inteligência artificial e machine learning.
* Estrutura de redes neurais artificiais (camadas, neurônios, pesos, função de ativação).
* TensorFlow e Keras como ferramentas de implementação.
* Estudos sobre reconhecimento óptico de caracteres (OCR).

**7. Metodologia:**

* **Tipo de Pesquisa:** Aplicada, quantitativa, experimental.
* **Método Científico:** Indutivo, com testes empíricos.
* **Procedimentos Técnicos:** Implementação prática em Python, análise de desempenho.
* **Ferramentas:** Python, Jupyter Notebook, TensorFlow, Google Colab.
* **Dados Utilizados:** Conjunto de dados MNIST (imagens de dígitos manuscritos de 0 a 9).

**8. Cronograma (resumo de 4 meses):**

| **Etapa** | **Mês 1** | **Mês 2** | **Mês 3** | **Mês 4** |
| --- | --- | --- | --- | --- |
| Revisão teórica sobre IA e redes | ✅ |  |  |  |
| Estudo da base de dados e do TensorFlow |  | ✅ |  |  |
| Implementação e testes |  | ✅ | ✅ |  |
| Análise de resultados e relatório final |  |  |  | ✅ |

**9. Referências:**

* GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. MIT Press, 2016.
* Documentation TensorFlow: <https://www.tensorflow.org>
* LE CUN, Y. *The MNIST Database of handwritten digits* (http://yann.lecun.com/exdb/mnist/)

**Modelo de Plano de Pesquisa Científica – Para Preenchimento**

**Título do Projeto:**

*(Defina um título claro e objetivo para sua pesquisa)*

**1. Tema:**

*(Qual o tema geral do seu projeto?)*

**2. Problema de Pesquisa:**

*(Qual a pergunta que sua pesquisa tentará responder?)*

**3. Hipótese (se houver):**

*(Qual sua suposição inicial sobre a resposta ao problema?)*

**4. Objetivos:**

**Geral:**

*(Objetivo principal do seu trabalho)*

**Específicos:**

**5. Justificativa:**

*(Por que essa pesquisa é importante? Qual a relevância acadêmica, científica ou social?)*

**6. Fundamentação Teórica:**

*(Conceitos, autores, teorias e obras que embasam seu trabalho. Cite pelo menos 2 ou 3 referências.)*

**7. Metodologia:**

* **Tipo de Pesquisa:** ( ) Básica ( ) Aplicada
* **Abordagem:** ( ) Qualitativa ( ) Quantitativa ( ) Mista
* **Método Científico:** ( ) Dedutivo ( ) Indutivo ( ) Hipotético-dedutivo
* **Procedimentos Técnicos:**

*(Como você vai realizar a pesquisa? Quais ferramentas, linguagens ou ambientes serão usados?)*

**8. Cronograma (exemplo por mês – pode ser adaptado):**

| **Etapa/Atividade** | **Mês 1** | **Mês 2** | **Mês 3** | **Mês 4** |
| --- | --- | --- | --- | --- |
| Levantamento bibliográfico |  |  |  |  |
| Desenvolvimento da proposta |  |  |  |  |
| Coleta/análise de dados ou testes |  |  |  |  |
| Redação e apresentação final |  |  |  |  |

**9. Referências:**

*(Liste autores, artigos, livros, sites ou documentações utilizadas até agora ou previstas)*

**Atividade Prática: Elaboração de um Plano de Pesquisa Científica Aplicada à Computação**

**Objetivo:**

Permitir que os alunos elaborem um plano de pesquisa científica realista, aplicando os conhecimentos de metodologia científica a um tema de interesse na área de Ciência da Computação.

**Modalidade:**

Trabalho em grupos de 2 alunos.

**Descrição da Atividade:**

Cada grupo deverá:

1. **Escolher um tema relevante** da área de Ciência da Computação (ex: segurança da informação, redes neurais, banco de dados, programação, algoritmos, IoT, robótica, jogos, etc.).
2. **Elaborar um plano de pesquisa científica**, seguindo o modelo fornecido em branco.
3. **Apresentar oralmente** sua proposta para a turma (5 a 10 minutos).
4. **Entregar o plano preenchido** (impresso ou digital) como produto final da atividade.

**Etapas e Prazos Sugeridos (adaptável):**

| **Etapa** | **Tempo estimado** |
| --- | --- |
| Formação dos grupos e escolha do tema | 1 aula |
| Pesquisa teórica e rascunho inicial | 2 aulas |
| Finalização do plano e ensaio da apresentação | 1 aula |
| Apresentações e entrega final | 1 aula |

**Entrega Esperada:**

* 1 plano de pesquisa preenchido por grupo (impresso ou PDF).
* Apresentação em sala com exposição das principais partes do plano.

**Critérios de Avaliação (sugestão):**

| **Critério** | **Pontuação** |
| --- | --- |
| Clareza e coerência do tema | 2,0 pts |
| Definição do problema e hipótese | 2,0 pts |
| Justificativa e objetivos | 2,0 pts |
| Metodologia e cronograma | 2,0 pts |
| Apresentação oral | 2,0 pts |
| **Total** | **10,0 pts** |

**Dicas de Temas para Escolha:**

* Como a inteligência artificial pode prever falhas em sistemas computacionais?
* Ferramentas de detecção de plágio em código-fonte.
* Comparativo de linguagens para ensino de programação no Ensino Médio.
* Otimização de consumo de energia em redes sem fio.
* Reconhecimento facial com Python e OpenCV.
* Aplicativos móveis para apoio ao ensino de lógica de programação.