

## CAPÍTULO 2

# REVISÃO DA LITERATURA

### 2.1. Introdução

A literatura pesquisada demonstra que existe uma dificuldade de aprendizagem nas disciplinas de lógica de programação e linguagem de programação, considerando a necessidade de o aluno interpretar o problema proposto, criar uma solução algorítmica, construir a lógica que soluciona o problema. Assim com base em duas teorias da aprendizagem de Ausubel e Feurstein investigam-se quais diferenças e perfil do grupo de estudantes, que pode se situar no mesmo perfil de estudantes de outras instituições de ensino e nacionalidade, que acabam tendo insucesso no aprendizado do algoritmo, problemática comum que não pode ser admitida apenas como fato ou fenômeno, mas como necessidade de pesquisa e melhoria dos esforços docentes e organizacionais melhorando os resultados.

### 2.2. Delimitação da pesquisa bibliográfica

Para o presente trabalho o enquadramento teórico foi consultado obras de duas teorias do aprendizado, conhecidas mundialmente na área de psicologia da aprendizagem. A problemática pesquisada é fato verificado por docentes há mais de vinte anos no Instituto Federal de Sergipe, relatado também nas pesquisas de Ciência da Educação, Psicologia e Educação em Informática apresentadas em congressos científicos, bibliografia impressa e artigos científicos disponíveis nas bases de dados (web) das universidades, fato este que nos levou à presente tarefa de pesquisa nesta dissertação.

### 2.3. A cognição e os subsunçores da disciplina de Lógica de Programação

Considerando que a aprendizagem significativa é o mecanismo humano utilizado para adquirir, agregar, memorizar as diversas ideias e informações do vasto campo do conhecimento humano, conforme a teoria de aprendizagem de David Ausubel fará uma explanação das formas de aprendizagens descritas nesta teoria e utilizaremos em cada tópico de assunto da disciplina de *lógica de programação* uma relação de subsunçores possíveis ou necessários ao aprendizado da mesma, como forma de identificar as “ancoragens” indispensáveis que quando inexistentes comprometam o significado de cada ideia e conceito que sem existirem inviabiliza totalmente a

aprendizagem significativa e concede espaço para a aprendizagem por memorização com todas as desvantagens anteriormente mencionadas (MOREIRA, 1997, p.2-3).

O material didático utilizado na disciplina somente será significativo se o conhecimento prévio da estrutura cognitiva do aprendiz for possuidor de um conjunto de ideias e conceitos claros e suficientes que sirvam de “ancoragem” ao novo conhecimento.

O autor português João Felix Praia (Faculdade de Ciências da Universidade do Porto), publicou artigo durante o acontecimento do ‘Contributos do III Encontro Internacional sobre Aprendizagem Significativa, Peniche, 2000’ que aconteceu em Portugal, artigo denominado *Aprendizagem significativa em D. Ausubel: Contributos para uma adequada visão da sua teoria e incidências no ensino*, p.125, conforme veremos a seguir.

O primeiro tipo de aprendizagem significativa é a aprendizagem do significado de símbolos individuais, principalmente palavras, denominado de **aprendizagem representacional** por Ausubel. (PRAIA, 2000, p.125)

O segundo tipo **aprendizagem de conceito** é um tipo de aprendizagem representacional, sendo que neste caso, se faz a aprendizagem de conceitos, objetos e acontecimentos. O conceito se faz representar por palavras específicas. (PRAIA, 2000, p.125)

O terceiro tipo **aprendizagem proposicional** tem como objetivo aprender o significado das ideias expressadas por grupos de palavras.

Comparativamente, os dois primeiros tipos, aprendizagem de proposição e aprendizagem de conceitos tem a mesma base, a aprendizagem representacional. (PRAIA, 2000, p.126)

As competências necessárias para que ocorra a aprendizagem significativa, segundo o modelo ausubeliano, para viabilizar a aprendizagem significativa exige:

1. Que o aprendiz tenha disposição a aprender, para que efetivamente relacione de forma não-arbitrária, a nova ideia à sua estrutura cognitiva. Caso objetive apenas memorizar sem que haja esta relação o processo e o resultado são apenas mecânicos e sem uma efetiva compreensão (PRAIA, 2000, P.127)
2. O material será potencialmente significativo, caso o material de estudo tenha algum significado para o aprendiz, que deve fazer uma relação com ideias previamente existentes e relevantes pertencentes ao conhecimento prévio do aprendiz. Deve haver para cada aprendiz uma ideia âncora que seja possível relacionar com as novas ideias contidas no material de estudo (PRAIA, 2000, P.127)

Ausubel, propõe quatro princípios, para a facilitação da aprendizagem do conteúdo programático (MOREIRA, CABALLERO e RODRÍGUEZ, 1997, p.37).

1. **diferenciação progressiva** - princípio que afirma que os conceitos gerais e inclusivos devem ser apresentados primeiro, por facilitar o estudo do conteúdo visto que o conhecimento existente na estrutura cognitiva possui uma hierarquia.
2. **reconciliação integrativa** – princípio segundo o qual deve haver na instrução uma exploração da relação entre as ideias. Identificar quais são as semelhanças e diferenças mais importantes e atenuar as diferenças aparentes e concretas.
3. **organização seqüencial** - princípio que afirma que o conteúdo apresentado deve estar em uma seqüência de tópicos que permita uma ordem lógica em sua relação de dependência entre as ideias.
4. **consolidação** – princípio que afirma que deve haver um domínio do conteúdo estudado antes que novo material didático seja utilizado, aumenta a probabilidade de aprendizado.

O conteúdo programático da Disciplina de Programação I, que contempla os tópicos de Lógica de Programação (algoritmo) juntamente com Linguagem Pascal do Curso técnico de Informática, modalidade integrado (2º grau e curso técnico profissionalizante) do Instituto Federal

de Sergipe, carga horária total de 160 horas/aula, possui os seguintes tópicos:

1. Lógica Matemática
2. Conceito de Algoritmo;
3. Metodologia de desenvolvimento de Algoritmos;
4. Português;
5. Conceito de Linguagem de Programação;
6. Conhecendo o ambiente do Pascal;
7. Estrutura Básica de um Programa em Pascal;
8. Uso de Constantes, variáveis e expressões;
9. Comando de Atribuição, Entrada (Leitura) e Saída (Impressão);
10. Estrutura de Controle;
11. Variáveis Compostas;
12. Modularização;
13. Rotina de manutenção em arquivos.

O conteúdo programático da Disciplina de Lógica de Programação, do Curso técnico de Informática, modalidade subsequente do Instituto Federal de Sergipe, com carga horária de 06 horas/aulas semanais, carga horária total de 108 horas/aula, possui os seguintes tópicos:

EMENTA:

1. Conceito de algoritmo.
2. Lógica de programação e programação estruturada.
3. Qualidade de um algoritmo.
4. Linguagem de definição de algoritmos.
5. Estrutura de um algoritmo.
6. Identificadores.
7. Variáveis.
8. Declaração de variáveis.
9. Operações Básicas.
10. Comandos de Entrada e Saída.
11. Comandos de Controle de Fluxo.
12. Estruturas de Dados homogêneos.
13. Modularização.

## **Tópico - Conceito de algoritmo**

Cada docente tem autonomia para escolher o material didático, podendo expandir ou reduzir o desenvolvimento de diversos tópicos e trabalhar diversos conceitos âncoras ou subsunçores, contidos em cada tópico.

Por exemplo, para estes dois conceitos de algoritmo.

“Um algoritmo é uma série ordenada de passos não-ambíguos, executáveis.” (BROOKSHEAR, 2003, p.151)

“Algoritmo é um processo de cálculo matemático ou de resolução de um grupo de problemas semelhantes”. (MANZANO e OLIVEIRA, 2000, p.6)

Conceitos bases são necessários para que o novo conceito seja significativo.

Primeiro subsunçor–“série ordenada de passos”

Segundo subsunçor–“Não-ambíguo” Terceiro subsunçor–“executável”.

Conceitos bases necessários para que o segundo conceito seja significativo.

Primeiro subsunçor – “Processo de cálculo matemático” ou Segundo subsunçor – “Resolução de problemas”

Para cada um destes subsunçores, pode ocorrer a seguinte situação, exclusivamente ou associado:

- 1 - Inexistência de Subsunçor.
- 2 - Existir um subsunçor que permita a aprendizagem significativa representacional.
- 3 - Existir um subsunçor que permita a aprendizagem significativa subordinada.
- 4 - Existir um subsunçor que permita a aprendizagem significativa superordenada.
- 5 - Existir um subsunçor que permita a aprendizagem significativa combinatória.

Cada aprendiz terá o seu exclusivo perfil definidor de existência de ideias em sua estrutura cognitiva, base informacional e ideológica para permitir ou não a aprendizagem significativa. A simbologia linguística utilizada no material didático e docente terá representações diferenciadas para cada aprendiz.

## **Tópico – Lógica de programação e programação estruturada**

Exemplo de ideias âncoras ou subsunçores deste tópico. Percebamos que o tópico tem a mesma denominação da disciplina, englobando, portanto diversos conceitos e ideias, que muitas vezes possuem diversas denominações que representam a mesma ideia.

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

“Instruções”

“Fluxo de execução”

“código”

“seqüencia de execução”

“condições”

“interrupção de execução”

”programação”

“programação estruturada”

”subprogramas”

”subrotinas”

”modularização”

”estrutura de controle”

”formatação de programa”

”comentário”

“refinamento sucessivo”

“fluxograma, diagrama de blocos”

“português estruturado”

A aplicação prática do tópico é o objetivo proposto pela disciplina.

### **Tópico – Qualidade de Software**

Exemplos de ideias bases ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “características de software”
- “manutenção de software”
- “medida de qualidade”
- “processo de desenvolvimento”
- “requisitos de qualidade”
- “teste de software”
- “métodos de métrica”
- “padronização de qualidade”
- “padrão ISO/IEC”
- “pesquisa e planejamento”

### **Tópico – Linguagem de definição de Algoritmo**

Exemplos ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “Português estruturado”
- “portugol”
- “fluxograma”
- “diagrama de blocos” “diagrama de chapin”
- “pseudo-código”

### **Tópico – Estrutura de um algoritmo**

Exemplos de ideias âncoras possíveis e necessários para aprendizagem conceitual significativa do tópico

- “tarefa e solução algorítmica”
- “problemas de algoritmo (computacional)”
- “Seqüência de passos (Instruções)”
- “repetição de passos”
- “decisões ou seleções ou desvio (passos)”

### **Tópico – Identificadores**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “conceito de identificador”
- “nome de variáveis”
- “tipos definidos”
- “nomes de procedimentos”
- “nomes de funções”
- “nomes de constantes”
- “caracteres ascii, ebcdic”
- “conjunto alfabético, alfanumérico, numérico”

### **Tópico – Variáveis**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “função das variáveis”
- “Valores das variáveis”
- “tipos de variáveis”
- “declaração de variáveis”

### **Tópico – Declaração de Variáveis**

Neste caso específico os tópicos “Variáveis” e “declaração de variáveis” é um exemplo típico de aprendizagem significativa derivativa, onde os subsunçores são complementares.

### **Tópico – Operações Básicas**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “operações aritméticas”
- “operações lógicas”
- “tabela-verdade”
- “operação lógica composta”

### **Tópico – Comandos de Entrada e Saída**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “operações aritméticas”
- “expressões booleanas - operações lógicas”
- “tabela-verdade”
- “expressões booleanas composta”
- “instruções de entrada - leia”
- “sintaxe da instrução de entrada”
- “execução da instrução de entrada”
- “instruções de saída - imprima”
- “sintaxe da instrução de saída”
- “execução da instrução de saída”

### **Tópico – Comandos de Controle de Fluxo**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “estrutura de controle sequencial”
- “estrutura de controle condicional”
- “estrutura de controle condicional simples”
- “estrutura de controle condicional composta”

### **Tópico – Estrutura de Dados Homogênea**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “Vetores”
- “Matrizes”
- “dimensão de uma matriz”
- “elementos de uma matriz”

### **Tópico – Modularização**

Exemplos de ideias âncoras ou subsunçores possíveis e necessários para aprendizagem conceitual significativa do tópico

- “conceito de modularização”
- “Função”
- “Procedimento”
- “parâmetro”
- “passagem de parâmetro”
- “passagem de parâmetro por valor e referência”

## **2.4. O processo ensino-aprendizagem da disciplina de lógica de programação no Instituto Federal de Sergipe**

### **2.4.1. Recursos didáticos e audiovisuais**

A partir do projeto do curso definido pelo Instituto Federal de Sergipe - IFS, onde cada disciplina traz seus objetivos e conteúdo programático, o docente elabora o plano de curso, plano da unidade e o plano de aula, distintos entre si somente pela abrangência, sendo que o plano de curso da disciplina aborda as atividades do semestre e bimestre, o plano da unidade, divide o curso da disciplina em duas unidades, o plano de aula é o tópico da disciplina ministrado por dia de aula.

Os recursos audiovisuais mais utilizados na prática docente do Instituto Federal de Sergipe - IFS - são: quadro branco, retroprojetor, projetores de slides, computadores e televisores, sendo que existe a predominância no uso do quadro branco, por ser o recurso disponível em todas as salas de aula, enquanto os retroprojetores, projetores computadores e televisores não estão disponíveis para todas as salas, dependendo de reserva para uso do referido recurso tecnológico, exceto nos cursos da área de informática que possuem oito laboratórios com microcomputadores conectados em rede local com acesso a internet.

Conforme explica a autora Débora Bonat em sua obra *Didática do Ensino Superior*, p.79.

Hoje, os professores esbarram em toda tecnologia que está à disposição dos alunos em todos os momentos de sua vida. O professor que se mantém estagnado em uma aula tradicional, meramente expositiva, acaba perdendo lugar para outras questões mais interessantes.

### **2.4.2. Metodologias docentes**

Prevalece a aula expositiva explicando os fundamentos e os elementos da disciplina de lógica de programação. Acompanhada do material didático impresso, livro ou apostila, que fica a critério de cada docente para adotar os textos escolhidos pelo mesmo. Geralmente são complementados por

trabalhos em grupo dos alunos, seminários, palestras, conferências e relatórios que complementam o conteúdo da disciplina.

A pedagogia tradicional ainda prevalece nas instituições de ensino brasileiras, entretanto a crítica que se faça a elas depende da prática docente, conforme bem explica os autores Antônio Feltran Filho e Ilma Passos Alencastro Veiga, em sua obra *Técnicas de ensino: por que não?*, p.46:

A aula expositiva, estruturada a partir da Pedagogia tradicional, tem sido criticada pela forma como vem sendo adotada pela grande maioria dos professores, a saber, de modo mecânico e desvinculado da prática social, produzindo uma postura autoritária do professor e inibindo a participação do aluno. Entretanto, nem todas as aulas expositivas podem ser consideradas com essas características. Adotando uma atitude dialógica um professor poderá ser muito dinâmico e transformador por intermédio de suas aulas expositivas. A questão não está em se rotular essa técnica como tradicional e rejeitá-la como meio de ensino. Ocorre que professores com atitudes tradicionais tornarão uma aula autoritária, monótona e desinteressante, seja ela expositiva ou não, enquanto que professores com atitude crítica mostram-se capazes de levar seus alunos a reelaborar ou produzir conhecimentos por meio de aulas expositivas.

Necessário ressaltar que a maior parte destas ideias âncora ou subsunções são de natureza abstrata, exigindo maior esforço mental para relacionar ideais âncoras com novas ideias, para que possa viabilizar a compreensão e aprendizagem significativa de cada aprendiz.

Segundo a obra *Ordenadores en las aulas: La clave es la metodología*, coordenada por Carme Barba, Sebastiá Capella, Equipo de la Comunitat Catalana de WebQuest, Editora Graó, 2010, p. 85, são características da inteligência lógico-matemática.

## La inteligencia lógico-matemática

Capacidad de utilizar los números con eficacia (matemáticos, contables, estadísticos) y de razonar bien (científicos, programadores informáticos, especialistas en lógica). Esta inteligencia incluye la sensibilidad a patrones y relaciones lógicas, afirmaciones y proposiciones (si..., entonces; causa-efecto), funciones y otras abstracciones relacionadas. Los procesos utilizados en la inteligencia lógico-matemática incluyen categorización, clasificación, deducción, generalización, cálculos y prueba de hipótesis. El pensamiento lógico-matemático comienza desde las primeras edades, siendo la adolescencia y los primeros años de la vida adulta las etapas en las que se consolida y alcanza el máximo desarrollo. Las capacidades matemáticas superiores comienzan a declinar después de los cuarenta años (Armstrong,2006).

Gardner (2001) sostiene que la inteligencia lógico-matemática no es necesariamente superior a las otras inteligencias, y que no se le debe otorgar universalmente el mismo prestigio. Existen otros procesos lógicos y métodos de solución de problemas inherentes a cada una de las inteligencias. Cada inteligencia posee su propio mecanismo ordenador, sus principios, sus operaciones fundamentales y sus recursos.

## Características de la inteligencia lógico-matemática

Según Gardner, la inteligencia lógico-matemática incluye numerosas clases de pensamientos. Nos dice que esta inteligencia comprende tres campos muy amplios e interrelacionados: la matemática, la ciencia y la lógica.

Es probable que una persona con una inteligencia lógico-matemática muy desarrollada presente algunas de las siguientes características:

- . Percibe los objetos y su función en el entorno.
- . Utiliza símbolos abstractos para representar objetos y conceptos concretos.
- . Participa activamente en las clases de matemáticas.
- . Domina los conceptos de cantidad, tiempo y causa-efecto.



- . Demuestra habilidad para encontrar soluciones lógicas a los problemas.
- . Percibe modelos e relaciones.
- . Plantea e pone a prueba hipótesis.
- . Utiliza diversas habilidades matemáticas, como estimación, cálculo de algoritmos, interpretación de estadísticas y representación visual de información en forma gráfica.
- . Se entusiasma con operaciones complejas, como ecuaciones, fórmulas físicas, programas de computación o métodos de investigación.
- . Piensa de forma matemática mediante la recopilación de pruebas, la enumeración de hipótesis, la formulación de modelos, el desarrollo de contraejemplos y la construcción de argumentos sólidos.
- . Usa la tecnología para resolver problemas matemáticos.
- . Muestra interés por carreras como: ciencias económicas, tecnologías, informática, derecho, ingeniería y química.
- . Crear nuevos modelos o percibe nuevas facetas en ciencia o matemáticas.

