



GOVERNO DO
ESTADO DO CEARÁ
Secretaria da Educação

ESCOLA ESTADUAL DE EDUCAÇÃO PROFISSIONAL - EEEP

ENSINO MÉDIO INTEGRADO À EDUCAÇÃO PROFISSIONAL

CURSO TÉCNICO EM INFORMÁTICA

MANUAL DO PROFESSOR

POO/JAVA



**GOVERNO DO
ESTADO DO CEARÁ**
Secretaria da Educação

Governador

Cid Ferreira Gomes

Vice Governador

Domingos Gomes de Aguiar Filho

Secretária da Educação

Maria Izolda Cella de Arruda Coelho

Secretário Adjunto

Maurício Holanda Maia

Secretário Executivo

Antônio Idilvan de Lima Alencar

Assessora Institucional do Gabinete da Seduc

Cristiane Carvalho Holanda

Coordenadora da Educação Profissional – SEDUC

Andréa Araújo Rocha



**GOVERNO DO
ESTADO DO CEARÁ**
Secretaria da Educação

Coordenador Técnico Pedagógico:

Renanh Gonçalves de Araújo

Equipe de Elaboração:

Adriano Gomes da Silva

Cintia Reis de Oliveira

Fernanda Vieira Ribeiro

João Paulo de Oliveira Lima

Liane Coe Girão Cartaxo

Mirna Geyla Lopes Brandão

Moribe Gomes de Alcântara

Paulo Ricardo do Nascimento Lima

Renanh Gonçalves de Araújo

Rogério Lopes Vieira César

Programação Orientada Objetos / Java

MANUAL DO (A) PROFESSOR (A)

Janeiro DE 2013
FORTALEZA/CEARÁ

Apresentação

No intuito de deixar claro à (o) professor (a) o que é esperado do aluno ao final da disciplina, este manual propõe os objetivos de aprendizagem referentes ao tema, acompanhado do conteúdo da disciplina. Disponibiliza uma bibliografia para o (a) professor (a), subsidiando-o (a) para aprofundar os debates, bem como, uma bibliografia de referência.

Elaborado no intuito de qualificar o processo de formação, este Manual é um instrumento pedagógico que se constitui como um mediador para facilitar o processo de ensino-aprendizagem em sala de aula.

A disciplina de POO/Java faz parte do eixo de programação Java para Desktop e segue um fio condutor iniciado pela Lógica de Programação e será seguida pela disciplina de Banco de Dados, e posteriormente por Laboratório de Software.

O principal objetivo deste manual é apresentar para o aluno uma alternativa para o processo de desenvolvimento de software orientado a objetos, embora existam outros processos, outros padrões, em fim, outras alternativas que, certamente, eles irão se deparar no mundo do trabalho. Esta experiência o fará entender a importância do processo de desenvolvimento de software para termos produtos eficiente e eficaz.

Sendo assim, temos a apresentação da UML, na Fase I do manual, ferramenta de documentação de projetos de software mais usada pelo mercado atualmente. Na Fase II, descrevemos alguns itens complementares da linguagem Java que não foram abordados no manual de Lógica de Programação, mas julgamos importantes. Em seguida temos a Fase III, onde teremos os conceitos de Orientação a Objetos. Por fim, na Fase IV, iremos implementar um projeto usando padrão MVC e interface usando Java.swing, como a essa altura os conceitos de banco de dados ainda não foram aprendidos, usaremos uma simulação do banco baseada em vetores de objetos.

É importante que o (a) professor (a) compreenda o propósito do método do curso, e assim, se aproprie do conteúdo e da metodologia proposta por meio das atividades pedagógicas, fazendo um estudo cuidadoso deste Manual e buscando aperfeiçoar sua didática para conduzir com sucesso as atividades propostas. Assim, Esperamos contribuir com a consolidação do compromisso e envolvimento de todos (professores e alunos) na formação desses profissionais.

Objetivos de Aprendizagem

Ao final da disciplina os alunos devem ser capazes de...

- Formular programas em linguagem Java.
- Interpretar e programas escritos em linguagem Java no paradigma orientado a objetos.
- Desenvolver aplicações com interfaces gráficas em linguagem Java respeitando o paradigma de orientação a objetos
- Desenvolver aplicações softwares usando padrão MVC.
- Compreender conceitos sobre sistemas de informação.
- Aplicar técnicas para levantamento de requisitos.
- Analisar e conceber requisitos para o desenvolvimento de sistemas.
- Interpretar e elaborar diagramas UML (diagramas de Classes, diagramas de Sequência e diagramas de Casos de Uso)

Conteúdo Programático

Fase I – A UML como Ferramenta de Programação Orientada a Objetos

- 1. Modelagem de Sistemas**
- 2. Introdução a Sistemas**
 - 2.1. Dado, Conhecimento e Informação
 - 2.2. O que é um sistema
 - 2.3. Como construir um Sistema de informação baseado em computador
 - 2.4. O papel do Analista de Sistemas
 - 2.5. Fases do desenvolvimento de sistemas
- 3. Requisitos**
 - 3.1. Introdução
 - 3.2. Levantamento de requisitos
- 4. História e evolução da UML**
- 5. Diagramas Estruturais da UML**
 - 5.1. Diagrama de Classe
 - 5.2. Diagrama de Objeto
 - 5.3. De Componentes
 - 5.4. De Implantação
 - 5.5. De Pacotes
 - 5.6. De Estrutura
- 6. Diagramas Comportamentais**
 - 6.1. De Caso de Uso
 - 6.2. Diagrama de Sequência
 - 6.3. Diagrama de Máquina de Estado
 - 6.4. De Atividades
- 7. Relacionamentos em UML**
 - 7.1. Relacionamento de Dependência
 - 7.2. Relacionamento de Associação
 - 7.3. Relacionamento de Generalização

Fase II – Programação na Linguagem Java

- 8. Visão Geral das Tecnologias Java**
 - 8.1. Introdução
 - 8.2. Tecnologias Java
- 9. Classes Java**
 - 9.1. Classe String
 - 9.2. Classe Math

Fase III – Programação Orientada a Objetos

- 10. Orientação a Objetos**

- 10.1. Introdução
- 10.2. Histórico
- 10.3. Fundamentos de Orientação a Objetos
- 11. Encapsulamento**
 - 11.1. O que é Encapsulamento?
- 12. Herança**
 - 12.1. O que é herança?
 - 12.2. Superclasse e Subclasse
 - 12.3. Herança múltipla
 - 12.4. Classe Object
- 13. Polimorfismo**
 - 13.1. O que é polimorfismo?
 - 13.2. Sobrecarga de métodos
 - 13.3. Classes e métodos abstratos
- Fase IV – Projeto Orientado a Objetos
- 14. Conhecendo e Organizando um Projeto no Padrão MVC**
 - 14.1. Padrão de Software
 - 14.2. Entendo o Padrão Model-View-Controller – MVC
- 15. Implementação do Projeto**
 - 15.1. Identificação do Projeto
 - 15.2. Criando o Projeto
 - 15.3. Criando os Pacotes MVC
 - 15.4. Pacote Model
 - 15.5. Pacote View
 - 15.6. Gerando um arquivo executável do sistema
- 16. Assuntos Complementares**
 - 16.1. Tratamento de Exceções em Java
 - 16.2. Convenções de código Java

Ementa

| Disciplinas | C/H | Ano | Sem. | Pré-Requisito |
|--------------------------------------|-----|-----|------|---------------|
| Programação Orientada a Objetos/Java | 100 | 2º. | 2º. | |

OBJETIVO OPERACIONAL

INTENÇÃO A SER ALCANÇADA

(Apresenta uma competência como sendo o comportamento global esperado)

Construir aplicações em linguagem Java compatíveis com o paradigma orientado a objetos utilizando padrão MVC e modelos descritos em UML.

COMPETÊNCIAS

(Descrevem os elementos essenciais da competência na forma de comportamentos particulares).

- Desenvolver diagramas UML básicos
- Implementar programas em linguagem Java
- Implementar aplicações em linguagem Java compatíveis com o paradigma orientado a objetos
- Aplicar o modelo MVC como padrão de desenvolvimento do software

FUNDAMENTOS TEÓRICO-PRÁTICOS

(Determinam e orientam as aprendizagens, permitindo a aquisição das competências de maneira progressiva e por etapas)

FASE I: A UML como Ferramenta de Programação Orientada a Objetos

- Identificar processos do mundo real que podem ser mapeados para sistemas informatizados
- Elaborar as listas de requisitos necessários a um projeto de software, de acordo com os tipos de requisitos estudados
- Compreender o ciclo de vida do desenvolvimento de software
- Criar diagramas de classes, sequência e casos de uso para documentação de sistemas

FASE II: Programação na Linguagem Java

- Implementar aplicações na linguagem Java **usando** ambiente de programação adequado (IDE NetBeans) no **ambiente** do laboratório de ensino de informática
- Modificar algoritmos e programas formulados por outros programadores
- Identificar e corrigir possíveis erros **a partir** de programas elaborados por outros programadores
- Identificar linguagem de programação apropriada para desenvolvimento de aplicações **a partir** das necessidades do problema

FASE III: Programação Orientada a Objetos

- Determinar as classes necessárias para a solução a partir das necessidades identificadas
- Desenvolver aplicações respeitando os princípios da programação orientada a objetos usando
- Implementar classes encapsuladas em Java
- Identificar e implementar classes que utilizam os conceitos de herança

- Criar aplicações usando o conceito de reaproveitamento de código
- Utilizar e implementar métodos polimórficos

FASE IV: Projeto Orientado a Objetos

- Identificar corretamente as classes que compõem cada pacote (Model-View-Controller) no padrão MVC
- Implementar a integração entre classes de pacotes diferentes

AÇÕES PEDAGÓGICAS

(Os limites a serem respeitados e os meios de aplicação).

- Possibilitar a execução de várias atividades na resolução de problemas;
- Estimular as iniciativas dos alunos respeitando os acordos estabelecidos sobre o trabalho a ser efetuado;
- Assegurar o acompanhamento periódico dos alunos;
- Intervir em casos de dificuldades ou de problemas;
- Guiar o processo de avaliação dos alunos fornecendo instrumentos tais como questionário, grades de análise, estudo de casos, etc;
- Motivar os alunos a executar as atividades propostas;
- Fornecer aos alunos meio de avaliar seu progresso;
- Assegurar a disponibilidade da documentação pertinente: conceitos elementares da lógica;
- Permitir aos alunos a terem trocas entre eles sobre dificuldades e soluções encontradas;
- Procurar integrar os conhecimentos adquiridos na elaboração de algoritmos para resolução de problemas;

QUADRO RESUMO DAS COMPETÊNCIAS:

(Quadro que relaciona aspectos relacionados a cada competência descrita na seção COMPETÊNCIAS e FUNDAMENTOS TEÓRICOS E PRÁTICOS).

| COMPETÊNCIA C1 | |
|---|--|
| Desenvolver e Interpretar os diagramas UML básicos | |
| CONTEXTO DE REALIZAÇÃO | |
| <p>A partir de:</p> <p>Utilizando (elementos de auxílio à aprendizagem): software de desenho de diagramas UML Astah Community</p> <p>Com a ajuda de (material didático): Livros, apostilas, anotações no caderno e orientações do professor.</p> <p>Ambiente (local propício): LEI – Laboratório de Ensino de Informática</p> | |
| ELEMENTOS DA COMPETÊNCIA | CRITÉRIOS DE DESEMPENHO |
| EC11 – Identificar processos do mundo real que podem ser mapeados para sistemas informatizados | <ul style="list-style-type: none"> ✓ Determinar corretamente a sequência de passos de um processo ✓ Relacionar corretamente processos do mundo real que podem ser mapeados no mundo computacional |
| EC12 – Elaborar as listas de requisitos necessários a um projeto de software, de acordo com os tipos de requisitos estudados | <ul style="list-style-type: none"> ✓ Identificar requisitos para sistema proposto |
| EC13 – Compreender o ciclo de vida do desenvolvimento de software | <ul style="list-style-type: none"> ✓ Descrever corretamente as etapas necessárias ao desenvolvimento de um software |
| EC14 – Criar diagramas de classes, sequência e casos de uso para documentação de sistemas | <ul style="list-style-type: none"> ✓ Identificar corretamente elementos em diagramas exibidos ✓ Analisar diagramas elaborados por outras pessoas do grupo ✓ Corrigir erros em diagramas propostos ✓ Elaboração de diagramas de acordo situações problema |

| COMPETÊNCIA C2 | |
|---|---|
| Implementar programas em linguagem Java | |
| CONTEXTO DE REALIZAÇÃO | |
| <p>A partir de:</p> <p>Utilizando (elementos de auxílio à aprendizagem): IDE NetBeans</p> <p>Com a ajuda de (material didático): Livros, apostilas, anotações no caderno e orientações do professor.</p> <p>Ambiente (local propício): LEI – Laboratório de Ensino de Informática</p> | |
| ELEMENTOS DA COMPETÊNCIA | CRITÉRIOS DE DESEMPENHO |
| EC21 – Implementar aplicações na linguagem Java usando ambiente de programação adequado (IDE NetBeans) no ambiente do laboratório de ensino de informática | ✓ Elaborar programas usando conceitos básicos de programação |
| EC22 – Modificar algoritmos e programas formulados por outros programadores | ✓ Elaborar modificações a partir de um programa já implementado |
| EC23 – Identificar e corrigir possíveis erros a partir de programas elaborados por outros programadores | ✓ Corrigir erros de sintaxe da linguagem Java. |

| COMPETÊNCIA C3 | |
|---|---|
| Implementar aplicações em linguagem Java compatíveis com o paradigma orientado a objetos | |
| CONTEXTO DE REALIZAÇÃO | |
| <p>A partir de:</p> <p>Utilizando (elementos de auxílio à aprendizagem): IDE NetBeans</p> <p>Com a ajuda de (material didático): Livros, apostilas, anotações no caderno e orientações do professor.</p> <p>Ambiente (local propício): LEI – Laboratório de Ensino de Informática</p> | |
| ELEMENTOS DA COMPETÊNCIA | CRITÉRIOS DE DESEMPENHO |
| EC31 – Determinar as classes necessárias para a solução a partir das necessidades identificadas | ✓ Mapear as classes necessárias a um software a partir de seus requisitos |
| EC32 – Desenvolver aplicações respeitando os princípios da programação orientada a objetos usando | ✓ Identificar a necessidade de usar herança |
| | ✓ Implementar o conceito de encapsulamento na implementação das classes |
| | ✓ Identificar a necessidade e implementar métodos polimórficos |

| COMPETÊNCIA C4 | |
|---|--|
| Aplicar o modelo MVC como padrão de desenvolvimento do software | |
| CONTEXTO DE REALIZAÇÃO | |
| <p>A partir de:</p> <p>Utilizando (elementos de auxílio à aprendizagem): IDE NetBeans</p> <p>Com a ajuda de (material didático): Livros, apostilas, anotações no caderno e orientações do professor.</p> <p>Ambiente (local propício): LEI – Laboratório de Ensino de Informática</p> | |
| ELEMENTOS DA COMPETÊNCIA | CRITÉRIOS DE DESEMPENHO |
| EC41 – Identificar corretamente as classes que compõem cada pacote (Model-View-Controller) no padrão MVC | <ul style="list-style-type: none">✓ Implementar sistema simples nas camadas propostas pelo padrão MVC✓ Implementar a integração entre classes de pacotes diferentes |

Cronograma de Atividades

| FASE I: Ferramentas de Programação Orientada a Objetos | | |
|--|--|----------------------------|
| Apresentação da disciplina. | Plano de ensino, método de trabalho, bibliografia e formas de avaliação. | Aula 01 |
| Núcleo de Conteúdos | Descrição | Aula |
| Introdução a Sistemas | Modelagem de Sistemas | Aula 02 |
| | Dado, Conhecimento e Informação | |
| | O que é um sistema | Aula 03 |
| | Como construir um Sistema de informação baseado em computador | |
| | O papel do Analista de Sistemas | Aula 04 |
| | Fases do desenvolvimento de sistemas | |
| Requisitos | Introdução | Aula 05 e Aula 06 |
| | Levantamento de requisitos | |
| | Exercício prático | Aula 07 e Aula 08 |
| História e evolução da UML | | Aula 09 |
| Diagramas Estruturais da UML | Diagrama de Classe | Aula 10, Aula 11 e Aula 12 |
| | Diagrama de Objeto | |
| | De Componentes | Aula 13 e Aula 14 |
| | De Implantação | |
| | De Pacotes | |
| | De Estrutura | |
| | Exercício prático | Aula 15 e Aula 16 |
| Diagramas Comportamentais | De Caso de Uso | Aula 17 e Aula 18 |
| | Diagrama de Sequência | Aula 19 e Aula 20 |

| | | |
|---|-------------------------------------|---------------------------|
| Relacionamentos em UML | Diagrama de Máquina de Estado | |
| | De Atividades | |
| | Relacionamento de Dependência | Aula 21 |
| | Relacionamento de Associação | |
| | Relacionamento de Generalização | |
| | Exercício prático | Aula 22 |
| Intervenção Pedagógica | | Aula 23 e Aula 24 |
| Avaliação da Fase | | Aula 25 e Aula 26 |
| Fase II – Programação na Linguagem Java | | |
| Núcleo de Conteúdos | Descrição | Aula |
| Visão Geral das Tecnologias Java | Introdução | Aula 27 e Aula 28 |
| | Tecnologias Java | |
| Classes Java | Classe String | Aula 29 e Aula 30 |
| | Classe Math | Aula 31 e Aula 32 |
| | Exercício prático | |
| Intervenção Pedagógica | | Aula 33 e Aula 34 |
| Avaliação da Fase | | Aula 35 e Aula 36 |
| Fase III – Programação Orientada a Objetos | | |
| Núcleo de Conteúdos | Descrição | Aula |
| Orientação a Objetos | Introdução | Aula 37 e Aula 38 |
| | Histórico | |
| | Fundamentos de Orientação a Objetos | |
| Encapsulamento | O que é Encapsulamento? | Aula 39 |
| | Exercício prático | Aula 40, Aula 41e Aula 42 |
| Herança | O que é herança? | Aula 43 |
| | Superclasse e Subclasse | Aula 44 |

| | | |
|---|--|--|
| | Herança múltipla | |
| | Classe Object | Aula 45 e Aula 46 |
| | Exercício prático | Aula 47, Aula 48, Aula 49 e Aula 50 |
| | | |
| Polimorfismo | O que é polimorfismo? | Aula 51 |
| | Sobrecarga de métodos | Aula 52, Aula 53 e Aula 54 |
| | Classes e métodos abstratos | Aula 55 e Aula 56 |
| | Exercício prático | Aula 57 e Aula 58 |
| Intervenção Pedagógica | | Aula 59 e Aula 60 |
| Avaliação da Fase | | Aula 61 e Aula 62 |
| Fase IV – Projeto Orientada a Objetos | | |
| Núcleo de Conteúdos | Descrição | Aula |
| Conhecendo e Organizando um Projeto no Padrão MVC | Entendo o Padrão Model-View-Controller – MVC | Aula 63 e Aula 64 |
| Implementação do Projeto | Identificação do Projeto | Aula 65 e Aula 66 |
| | Criando o Projeto | Aula 67 e Aula 68 |
| | Criando os Pacotes MVC | Aula 69 |
| | Pacote Model | Aula 70, Aula 71, Aula 72, Aula 73, Aula 74e Aula 75 |
| | Pacote View | Aula 76, Aula 77, Aula 78, Aula 79, Aula 80, Aula 81, Aula 82e Aula 83 |
| | Gerando um arquivo executável do sistema | Aula 84, Aula 85 e Aula 86 |
| Assuntos Complementares | Tratamento de Exceções em Java | Aula 87 e Aula 88 |
| | Exercício prático | Aula 89 e Aula 90 |
| | Convenções de código Java | Aula 91, Aula 92 |
| | Exercício prático | Aula 93 e Aula 94 |
| Intervenção Pedagógica | | Aula 95, Aula 96 |

| | |
|--------------------------|---|
| Avaliação da Fase | Aula 97, Aula 98, Aula 99 e Aula 100 |
|--------------------------|---|

Conteúdos Interdisciplinares

BASE TÉCNICA:

- Operadores relacionais
- Proposições lógicas
- Conceitos de estrutura lógica
- Conceitos de variáveis
- Técnicas para resolução de problemas (Top down, dividir pra conquistar etc)

DISCIPLINAS CONTRIBUTIVAS:

FILOSOFIA: Lógica aristotélica, abstração, eficiência e eficácia

MATEMÁTICA: Lógica matemática (proposições lógicas, tabelas verdade e álgebra de Boole), situações, expressões matemáticas (equações e inequações), conjuntos, funções (domínio, contra-domínio e imagem), substituição de variáveis em equações, constantes, sequência, finitude e matrizes

PORTUGUÊS: Leitura e interpretação de texto

Mapa de Análise de Avaliação

| | | | | |
|--------------------|-----------------|---------------|---------------|--|
| ESCOLA: | | | | |
| DISCIPLINA: | POO/JAVA | SÉRIE: | 2º ANO | |

| N | NOME | FASE I | | | FASE II | | | FASE III | | FASE IV | | |
|----|------|--------|------|------|---------|------|------|----------|------|---------|------|--|
| | | EC11 | EC12 | EC13 | EC14 | EC21 | EC22 | EC23 | EC31 | EC32 | EC41 | |
| 1 | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | |
| 3 | | | | | | | | | | | | |
| 4 | | | | | | | | | | | | |
| 5 | | | | | | | | | | | | |
| 6 | | | | | | | | | | | | |
| 7 | | | | | | | | | | | | |
| 8 | | | | | | | | | | | | |
| 9 | | | | | | | | | | | | |
| 10 | | | | | | | | | | | | |
| 11 | | | | | | | | | | | | |
| 12 | | | | | | | | | | | | |
| 13 | | | | | | | | | | | | |
| 14 | | | | | | | | | | | | |
| 15 | | | | | | | | | | | | |
| 16 | | | | | | | | | | | | |
| 17 | | | | | | | | | | | | |
| 18 | | | | | | | | | | | | |
| 19 | | | | | | | | | | | | |
| 20 | | | | | | | | | | | | |
| 21 | | | | | | | | | | | | |
| 22 | | | | | | | | | | | | |
| 23 | | | | | | | | | | | | |
| 24 | | | | | | | | | | | | |
| 25 | | | | | | | | | | | | |
| 26 | | | | | | | | | | | | |
| 27 | | | | | | | | | | | | |
| 28 | | | | | | | | | | | | |
| 29 | | | | | | | | | | | | |
| 30 | | | | | | | | | | | | |
| 31 | | | | | | | | | | | | |
| 32 | | | | | | | | | | | | |
| 33 | | | | | | | | | | | | |
| 34 | | | | | | | | | | | | |

| | | | | | | | | | | | | |
|----|--|--|--|--|--|--|--|--|--|--|--|--|
| 35 | | | | | | | | | | | | |
| 36 | | | | | | | | | | | | |
| 37 | | | | | | | | | | | | |
| 38 | | | | | | | | | | | | |
| 39 | | | | | | | | | | | | |
| 40 | | | | | | | | | | | | |
| 41 | | | | | | | | | | | | |
| 42 | | | | | | | | | | | | |
| 43 | | | | | | | | | | | | |
| 44 | | | | | | | | | | | | |
| 45 | | | | | | | | | | | | |

Legendas para o Mapa de Análise de Avaliação

| | |
|--------------------------------|--|
| ED – Em desenvolvimento | O Aluno já passou por uma avaliação de competência, e ainda não foi satisfatório em alguma das habilidades. |
| MB – Muito Bom | O Aluno passou pela a avaliação de competência e teve um resultado muito bom, este esta no nível intermediário a Avançado. |
| B – Básico | O Aluno passou pela a avaliação de competência e teve um resultado bom, este esta no nível básico. |
| MC – Muito Critico | Quando a aluno não esta conseguindo adquirir os elementos da competência. |
| Campo Vazio | Significa que o aluno não passou por uma avaliação de competência. |

Hino Nacional

Ouviram do Ipiranga as margens plácidas
De um povo heróico o brado retumbante,
E o sol da liberdade, em raios fúlgidos,
Brilhou no céu da pátria nesse instante.

Se o penhor dessa igualdade
Conseguimos conquistar com braço forte,
Em teu seio, ó liberdade,
Desafia o nosso peito a própria morte!

Ó Pátria amada,
Idolatrada,
Salve! Salve!

Brasil, um sonho intenso, um raio vívido
De amor e de esperança à terra desce,
Se em teu formoso céu, risonho e límpido,
A imagem do Cruzeiro resplandece.

Gigante pela própria natureza,
És belo, és forte, impávido colosso,
E o teu futuro espelha essa grandeza.

Terra adorada,
Entre outras mil,
És tu, Brasil,
Ó Pátria amada!
Dos filhos deste solo és mãe gentil,
Pátria amada, Brasil!

Deitado eternamente em berço esplêndido,
Ao som do mar e à luz do céu profundo,
Fulguras, ó Brasil, florão da América,
Iluminado ao sol do Novo Mundo!

Do que a terra, mais garrida,
Teus risonhos, lindos campos têm mais flores;
"Nossos bosques têm mais vida",
"Nossa vida" no teu seio "mais amores."

Ó Pátria amada,
Idolatrada,
Salve! Salve!

Brasil, de amor eterno seja símbolo
O lábaro que ostentas estrelado,
E diga o verde-louro dessa flâmula
- "Paz no futuro e glória no passado."

Mas, se ergues da justiça a clava forte,
Verás que um filho teu não foge à luta,
Nem teme, quem te adora, a própria morte.

Terra adorada,
Entre outras mil,
És tu, Brasil,
Ó Pátria amada!
Dos filhos deste solo és mãe gentil,
Pátria amada, Brasil!

Hino do Estado do Ceará

Poesia de Thomaz Lopes
Música de Alberto Nepomuceno
Terra do sol, do amor, terra da luz!
Soa o clarim que tua glória conta!
Terra, o teu nome a fama aos céus remonta
Em clarão que seduz!
Nome que brilha esplêndido luzeiro
Nos fulvos braços de ouro do cruzeiro!

Mudem-se em flor as pedras dos caminhos!
Chuvas de prata rolem das estrelas...
E despertando, deslumbrada, ao vê-las
Ressoa a voz dos ninhos...
Há de florar nas rosas e nos cravos
Rubros o sangue ardente dos escravos.
Seja teu verbo a voz do coração,
Verbo de paz e amor do Sul ao Norte!
Ruja teu peito em luta contra a morte,
Acordando a amplidão.
Peito que deu alívio a quem sofria
E foi o sol iluminando o dia!

Tua jangada afoita enfune o pano!
Vento feliz conduza a vela ousada!
Que importa que no seu barco seja um nada
Na vastidão do oceano,
Se à proa vão heróis e marinheiros
E vão no peito corações guerreiros?

Se, nós te amamos, em aventuras e mágoas!
Porque esse chão que embebe a água dos rios
Há de florar em meses, nos estios
E bosques, pelas águas!
Selvas e rios, serras e florestas
Brotem no solo em rumorosas festas!
Abra-se ao vento o teu pendão natal
Sobre as revoltas águas dos teus mares!
E desfraldado diga aos céus e aos mares
A vitória imortal!
Que foi de sangue, em guerras leais e francas,
E foi na paz da cor das hóstias brancas!



GOVERNO DO
ESTADO DO CEARÁ
Secretaria da Educação