

Sobre a Disciplina Programação Imperativa

Prof. Alberto Costa Neto
DComp/UFS



Sobre a Disciplina PI

- **Disciplina:** Programação Imperativa (COMP0334)
- **Equivalentes:**
 - Introdução à Ciência da Computação
 - Programação Imperativa (COMP0197)
- **Carga horária:** 60 horas
- **Créditos:** 4

Ementa

Noções fundamentais sobre algoritmos e sobre a execução de programas. Análise e síntese de problemas. Identificadores, tipos, constantes, variáveis, tipos. Operadores e expressões. Comandos condicionais e de repetição. Variáveis compostas homogêneas e heterogêneas. Procedimentos, funções e passagem de parâmetros. Noções sobre o uso de arquivos em programação. Algoritmos básicos de ordenação. Recursividade. Uma linguagem imperativa. Convenções de código. Boas práticas de programação.

Objetivos

Geral

- Apresentar os conceitos básicos e principais técnicas de desenvolvimento de programas de computador, tornando-o apto a compreendê-los e aplicá-los.

Específicos

- Tornar o aluno capaz de implementar programas básicos usando uma linguagem de programação imperativa.
- Habilitar o aluno a criar programas para executar computação científica na sua área de conhecimento.
- Colocar em prática os conhecimentos aprendidos no curso, desenvolvendo aplicações de pequeno porte em Python.

Conteúdo Programático

1º Unidade

- Motivação para Programar
- Hardware, software e princípios
- Visão Geral da Linguagem Python
- Preparação do Ambiente de Desenvolvimento
- Instruções primitivas: atribuição, entrada e saída
- Expressões
- Tipos
- Comandos Condicionais (if)
- Tratamento de exceções (try / except)
- Funções
- Laço While
- Strings
- Laços For

2º Unidade

- Listas
- Recursividade
- Dicionários
- Tuplas
- Arquivos

Afinal, por que o nome PI?

- Vem da denominação do Paradigma que vamos estudar: Paradigma Imperativo
 - Você escreve explicitamente as ordens e o computador obedece
 - Mais próximo do funcionamento real do computador
 - Existem outros paradigmas, como por exemplo:
 - Funcional
 - Orientado a Objetos

Método de Ensino

Metodologia - Presencial

- Conteúdo teórico estará disponível pelo Youtube
- Sistema que permite programar e tem autoavaliação
- Exercícios podem ser feitos em casa ou laboratório
- Tempo de aula será focado em exercícios e tirar dúvidas

Recursos Didáticos

As aulas serão ministradas presencialmente utilizando as seguintes ferramentas:

- **Youtube**, para exposição das videoaulas.
- **Ferramentas de Videoconferência**: Google Meet.
- **Editores de programas**: VS Code, Repl.it, Notepad++ ou Sublime Text.
- **Interpretador da linguagem Python**, que permite a verificação de erros de sintaxe e execução de programas em Python.
- **Apps** que permitem elaborar, executar e testar programas em smartphones e tablets.
- **Ambientes Virtuais de Aprendizagem (AVA)** SIGAA e Google Classroom
- Questionários com **Problemas de Programação** no site <http://thehuxley.com>

Correção de Questões

- Imagine se seu professor terá como corrigir 100 questões de cada um dos 50 alunos... Façamos as contas:
 - São 5.000 questões!
 - **Supondo que o professor gaste 6 min por questão, seriam necessários 30.000 minutos, ou seja, 500 horas!**
- Seria interessante ter uma ferramenta que ajudasse o professor, concordam?



Fonte:
http://2.bp.blogspot.com/_Q4jxieZF5Hk/TNbebADQ2FI/AAAAAAAAABM/gnjeS8-S2I0/s1600/estres-laboral-y-enfermedad-periodontal.jpg

- Uma ferramenta Web que oferece um **banco de problemas de programação** (juiz *on-line*).
- Os **alunos podem enviar soluções** (programas em várias linguagens de programação).
- O **The Huxley executa a solução** com entradas presentes em casos de teste e compara com o resultado esperado.
- Com esta ferramenta o aluno tem um **feedback imediato**

The Huxley



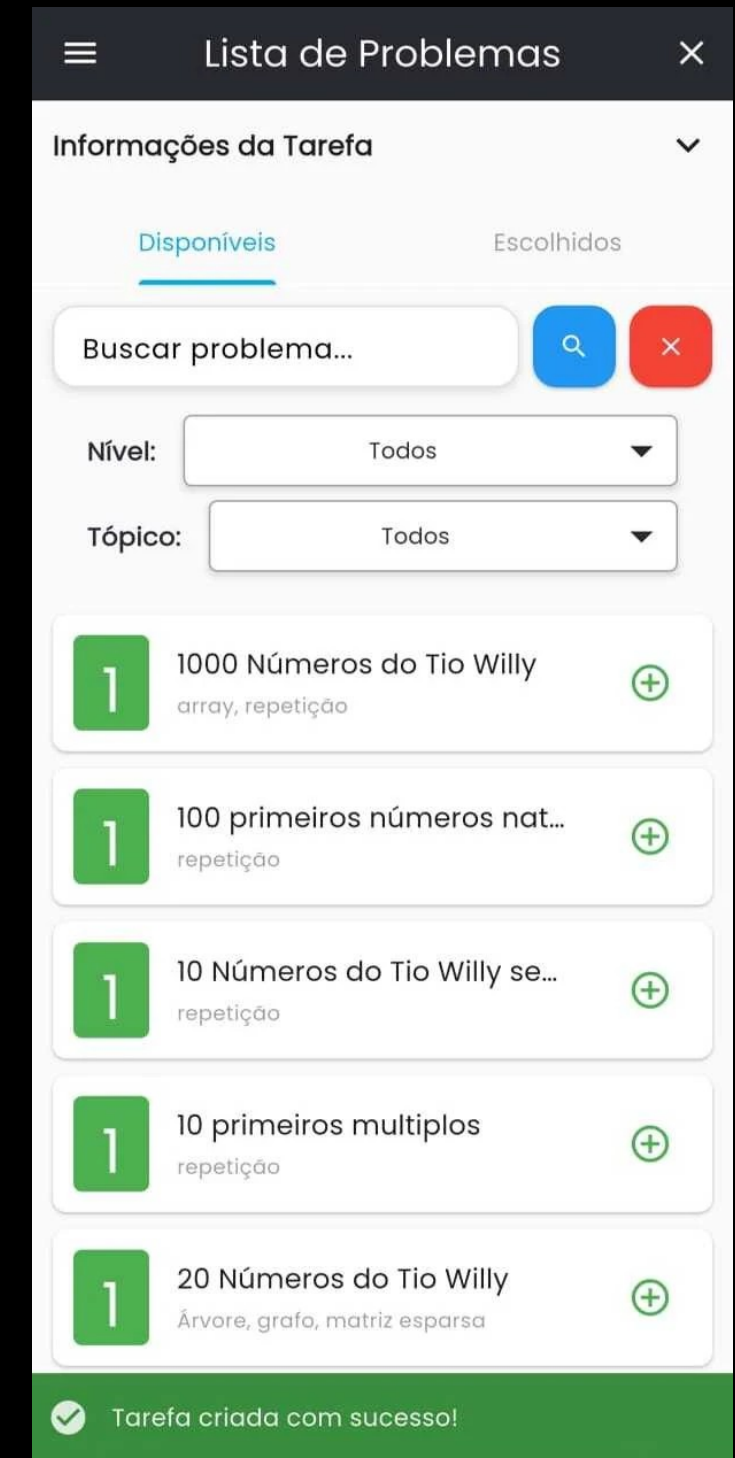
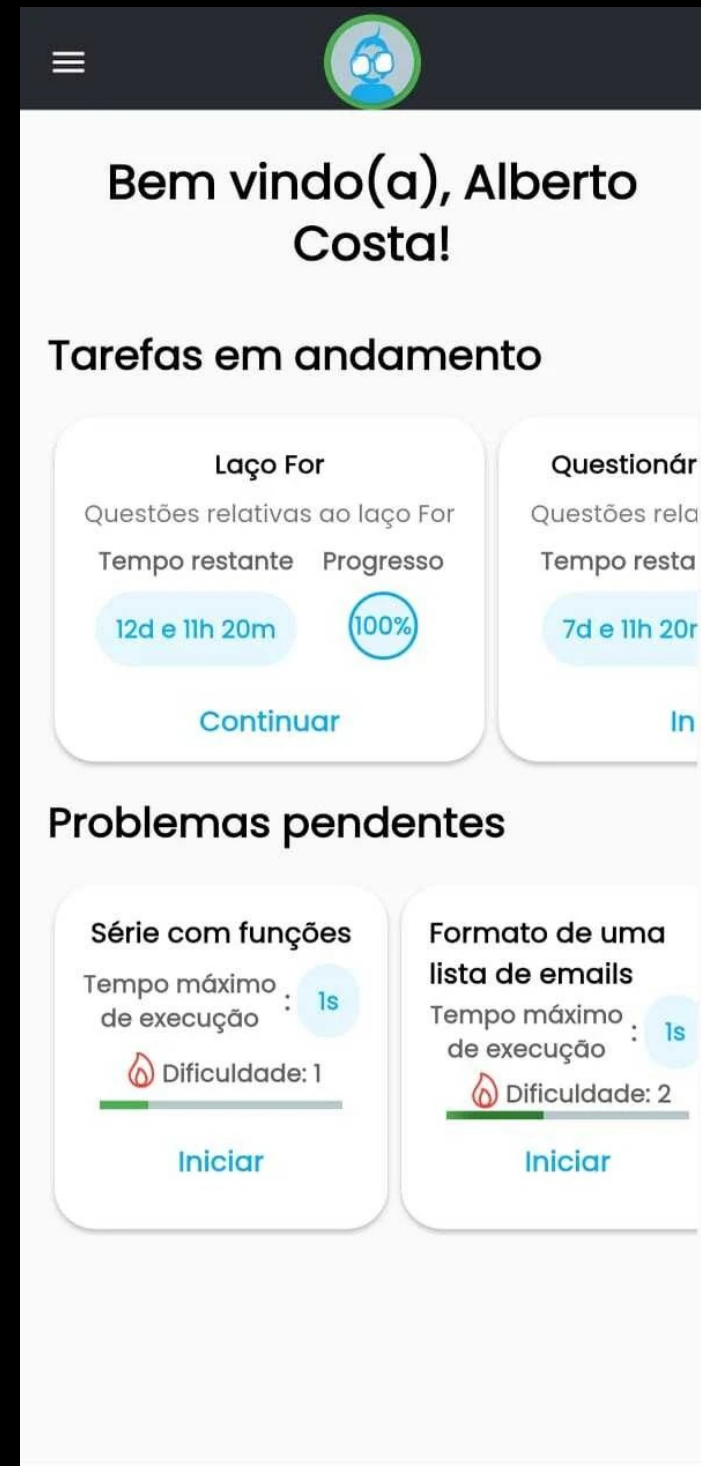
The screenshot displays the The Huxley web application interface. At the top, there is a navigation bar with the site logo, a search bar, and user options like 'Entrar' and 'Cadastrar'. Below this is a secondary menu with categories: Problemas, Submissões, Questionários, Grupos, and TopCoder. The main content area is split into two sections. On the left, a code editor shows a C program named 'helloworld.c' with the following code:

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("Hello world!");
5     return(0);
6 }
```

On the right, there is a 'CADASTRE-SE!' registration form with fields for 'Nome Completo', 'Login', 'Email', 'Confirme seu email', 'Senha', and 'Confirme sua senha', followed by a 'CONFIRMAR' button. A small advertisement for a book 'INTRODUÇÃO A PROGRAMAÇÃO COM A LINGUAGEM C' is also visible. The footer contains logos for UnB, Unit, UFAL, and FAT.

The Huxley Mobile

- Uma interface para o The Huxley para dispositivos móveis
- Desenvolvido e mantido por alunos da UFS
- Disponível na Play Store



Avaliação

Critério de Avaliação

Através de testes presenciais, obedecendo à fórmula:

$$\text{Nota Final} = (\text{NT1} + \text{NT2}) / 2$$

Onde:

- **NT1** = Nota do 1º Teste (peso 8) + Exercícios (peso 2)
- **NT2** = Nota do 2º Teste (peso 8) + Exercícios (peso 2)

Observação: Haverá um teste de reposição no final do semestre apenas para os alunos com falta justificada em algum teste, desde que a justificativa esteja prevista nas normas acadêmicas.

Pontuação dos Exercícios

- Haverá uma série de **questionários no The Huxley**, cada um com vários problemas de programação.
- A nota dos exercícios será calculada de acordo com a **pontuação obtida em cada questionário, variando de 0 a 10.**
- A nota dos exercícios da unidade será a **média das notas dos questionários.**
- Durante as aulas, o professor selecionará alunos para **apresentarem suas soluções para os exercícios e responder questões sobre o assunto.**

Apresentação dos Exercícios

- Caso o aluno se **negue a apresentar** ou **não o faça de forma que fique claro seu entendimento da solução e conhecimento sobre o assunto**, o aluno **perderá os pontos referentes aos Exercícios, passando a sua prova a ter peso 10.**
- Presume-se que, se o aluno resolveu de fato o exercício, o mesmo deve ser capaz de explicar seu raciocínio lógico e as construções de linguagem utilizadas.
- Caso o aluno apresente todas as vezes seguintes com sucesso, **pode recuperar a contagem de pontos dos exercícios.**

Calendário de Provas

As provas serão realizadas **presencialmente**:

- No horário da aula
- Segundo **calendário** e **orientações** divulgados no Plano de Ensino e no SIGAA

Controle de Frequência

Controle de Frequência (Turmas Presenciais)

- O aluno é obrigado a estar presencialmente nas aulas.
- Assim, a frequência dos alunos será computada através da **Lista de presença**.
- Caso o aluno assine a lista de presença e se ausente da sala de aula sem autorização do professor, terá sua presença computada parcialmente.

Bibliografia

Referências Bibliográficas (Básica)

- **Fundamentos da Programação de Computadores.** Ana Fernanda Gomes Ascencio / Edilene Aparecida Veneruchi De Campos. 3^o edição; 2012, Pearson; ISBN 978-8564574168
- **Algoritmos e Lógica de Programação.** Marco A. Furlan de Souza, Marcelo M. Gomes, Marcio V. Soares, Ricardo Concilio. Editora Cengage Learning, 2^a edição, 2011.
- **Algoritmos: Lógica para Desenvolvimento de Programação de Computadores.** José Augusto N. G. Manzano, Jayr Figueiredo de Oliveira. Editora Érica, 17^a edição, 2005.
- **Python Para Todos: Explorando Dados com Python 3.** Charles R. Severance. Publicação independentes; 1^o edição, 2020; ISBN: 979-8635191408

Referências Bibliográficas (Complementares)

- **Como pensar como um Cientista da Computação usando Python (traduzido)**. Allen Downey, Jeffrey Elkner, and Chris Meyers. 2002.
- **Introdução à Programação com Python**. Nilo Ney Coutinho, 2^o edição, 2014, ISBN: 978-85-7522-408-3.
- **Python para Desenvolvedores**. Luiz Eduardo Borges. Rio de Janeiro; 2010
- **Learning to Program Using Python**. Cody Jackson. CreateSpace Independent Publishing Platform

Contatos dos Professores

- Alberto Costa Neto – T03
 - alberto@dcomp.ufs.br
 - albertocn@academico.ufs.br

Como proceder em caso de dificuldade?

- Sempre que identificar alguma dificuldade, dúvida sobre conceitos das videoaulas ou problemas, **entre em contato com o(s) professor(es)** responsáveis pela sua turma.
- Caso não consiga **acessar os AVAs ou sites**, também entre em contato com o(s) professor(es).

Não deixe de tirar suas dúvidas!

E sejam bem-vindos ao curso de PI!!!