

# INF01202 - Algoritmos e Programação

Semana 15 - Aula Prática

Prof. Vinícius Garcia Pinto

21-06-2019

## Avaliação das Atividades Práticas

A avaliação das atividades práticas do segundo bimestre (a partir da semana 09) levará em conta os seguintes critérios:

- **corretude** do programa, ou seja, o programa mostra o resultado correto para uma dada entrada;
- **correspondência ao enunciado**, o programa atende a todos os requisitos presentes no enunciado da questão;
- **validação das entradas** quando solicitado no enunciado. Exemplo: programa só deve aceitar valores maiores que 1;
- programas que **não compilarem** receberão **nota 0**. Caso seja necessário enviar uma resposta parcial ou incompleta, comente (preferível) ou retire eventuais linhas de código que estejam impedindo a compilação;
- **identação** do código;

A avaliação das aulas práticas pode ser consultada na planilha disponibilizada em <http://bit.ly/praticas-inf01202-20191>. Eventualmente as notas incluem **comentários** para o aluno com sugestões, correções, justificativas ou dicas. Em caso de dúvidas quanto a correção, entre contato com o professor pelo e-mail [vinicius.pinto@inf.ufrgs.br](mailto:vinicius.pinto@inf.ufrgs.br) informando a questão e a aula a que se refere.

## Instruções de envio

- Uma resposta por arquivo, com nome correspondente ao número da questão.
- Nome do arquivo no formato `problemaX.c`. Exemplo: `problema1.c`, `problema2.c`, `problema3.c`.

- Não colocar o nome do aluno ou o cartão UFRGS no nome do arquivo.
- Comentário no cabeçalho de cada arquivo informando o nome completo do aluno e o número do cartão UFRGS.

```
// Nome do Aluno: Meu nome completo
// Cartao UFRGS: 00XXXXXX

/* Breve descrição sobre o problema e sobre o que faz
o código. */

#include<stdio.h>

int main(){

    // Solução do problema

    return 0;
}
```

Exemplo de arquivo a ser enviado.

- Enviar na entrada correspondente no Moodle Acadêmico.
  - Atenção, o horário limite para envio é o horário de término da aula:
    - \* Turma I: 15h10;
    - \* Turma J: 17h10.
  - Envie cada exercício tão logo quanto possível. São permitidas correções, reenvios, e adições até o horário de término da aula.

### Verificação anti-plágio

A detecção de plágio em qualquer atividade implicará penalidades (nota zero) a todos os envolvidos!

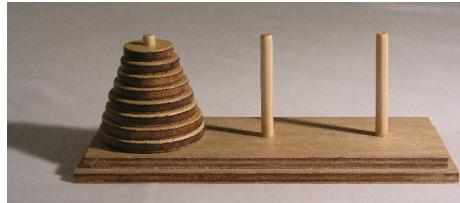
- **todos** os materiais entregues (práticas e trabalho) são submetidos a verificação anti-plágio

## Algumas dicas & erros recorrentes

A lista de dicas e erros recorrentes está disponível em: <https://github.com/viniciusvgp/intro-prog-c>

## Problema 1

Em um dos filmes da série *Planeta dos Macacos*, inspirada no livro *La planète des singes* de Pierre Boulle, um experimento baseado no quebra-cabeça Torre de Hanói é utilizado para testar a inteligência dos macacos (<https://youtu.be/aMEbboWmVCo>). Segundo a Wikipédia, "*Torre de Hanói é um quebra-cabeça que consiste em uma base contendo três pinos, em um dos quais são dispostos alguns discos uns sobre os outros, em ordem crescente de diâmetro, de cima para baixo. O problema consiste em passar todos os discos de um pino para outro qualquer, usando um dos pinos como auxiliar, de maneira que um disco maior nunca fique em cima de outro menor em nenhuma situação. O número de discos pode variar sendo que o mais simples contém apenas três.*"



Fonte: <https://w.wiki/56A>

O número mínimo de movimentos necessários para passar todos os discos de um pino para outro qualquer é dado pela expressão:

$$hanoiMin(n) = \begin{cases} 1, & \text{se } n = 1 \\ 2 \times hanoiMin(n-1) + 1, & \text{se } n > 1 \end{cases}$$

- **Atenção:** a implementação deste problema deve ser implementada, obrigatoriamente, com uso de recursividade.

### Exemplos de execução:

```
Informe o número de discos presentes na torre: -5
Informe o número de discos presentes na torre: 0
Informe o número de discos presentes na torre: -9
Informe o número de discos presentes na torre: 4
0 número mínimo de movimentos necessários é: 15
```

```
Informe o número de discos presentes na torre: 15
0 número mínimo de movimentos necessários é: 32767
```

```
Informe o número de discos presentes na torre: 7
O número mínimo de movimentos necessários é: 127
```

## Problema 2

Dado o protótipo de função recursiva: `void imprev(char txt[ ], int tam)`, escreva o corpo dessa função. Sua função deverá como receber como parâmetros um texto e o tamanho deste texto. Em seguida, escreva um programa C que leia do teclado um texto de até 5000 caracteres úteis e utilize a função `imprev` para mostrar este texto de trás para frente.

- **Atenção:** a função `imprev` deve ser implementada, obrigatoriamente, com uma estrutura recursiva.

## Exemplo de execução:

```
Informe um texto:
Esta disciplina tem por objetivo introduzir conhecimento e
tecnicas necessarios para projeto e desenvolvimento de solucoes
de problemas, atraves da construcao de algoritmos e programas
que utilizam os principios da programacao estruturada. O aluno
que cursou esta disciplina deve ser capaz de analisar problemas
e elaborar programas que os solucionem, utilizando para isto a
linguagem de programacao C. Deve dominar os comando basicos,
estruturar os dados em tipos simples e estruturados, utilizar
conceitos de subprogramacao e recursao, alem de manipular
arquivos. A disciplina abrange os seguintes topicos: nocao de
algoritmo, dado, variavel, instrucao e programa; construcoes
basicas: atribuicao, leitura e escrita; estruturas de controle:
sequencia, selecao e iteracao; tipos de dados escalares:
inteiros, reais, caracteres, strings; tipos estruturados
basicos: vetores, matrizes, registros e strings; subprogramas:
funcoes com e sem retorno; recursao; arquivos.

Inverso:
.soviuqra ;oasruce ;onroter mes e moc seocnuf :samargorpbu
;sgnirts e sortsiger ,sezirtam ,serotev :socisab sodaruturtse
sopit; sgnirts ,seretcarac ,siaer ,sorietni :seralacse sodad ed
sopit; oacareti e oaceles ,aicneuqes :elortnoc ed saruturtse
;atircse e arutiel ,oaciubirta :sacisab seocurtsnoc ;amargorp e
oacurtsni ,levairav ,odad ,omtirogla ed oacon :socipot setniuges
so egnarba anilpicsid A .soviuqra ralupinam ed mela ,oasruce e
oacamargorpbu ed sotiecnoc razilitu ,sodaruturtse e selpmis
sopit me sodad so raruturtse ,socisab odnamoc so ranimod eveD
.C oacamargorp ed megaugnil a otsi arap odnazilitu ,menoiculus
so euq samargorp rarobale e samelborp rasilana ed zapac res eved
anilpicsid atse uosruc euq onula 0 .adaruturtse oacamargorp ad
soipicnirp so mazilitu euq samargorp e somtirogla ed oacurtsnoc
ad sevarta ,samelborp ed seoculus ed otnemivlovnesed e otejorp
arap soirassecen sacincet e otnemichecnoc rizudortni ovitejbo rop
met anilpicsid atsE
```