

Unidade 0 - Nivelamento - Processo AEDs II de Exercícios



PUC Minas

**Instituto de Ciências Exatas e Informática
Departamento de Ciência da Computação**

Agenda

- Introdução
- Comparação de Arquivos com *DIFF*
- Redirecionamento de Entrada e Saída
- Plataforma Verde
- Exercícios

Agenda

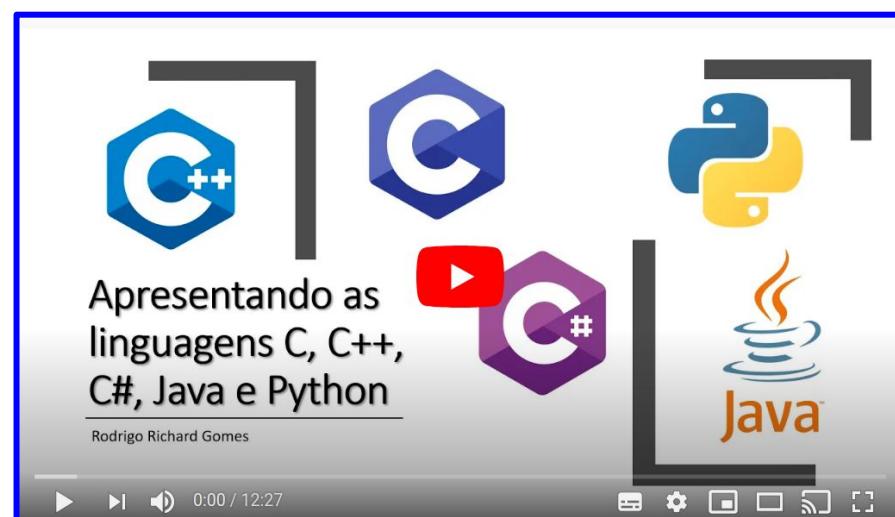
- **Introdução**
 - Comparação
 - Redirecionamento
 - Plataforma
 - Exercícios
- Contextualização: *for AEDs II*
 - Questões iniciais
 - Processo em um Exercício de Programação
 - Processo AEDs II de Exercícios

Agenda

- **Introdução**
 - Comparação
 - Redirecionamento
 - Plataforma
 - Exercícios
- **Contextualização: *for AEDs II***
 - Questões iniciais
 - Processo em um Exercício de Programação
 - Processo AEDs II de Exercícios

Linguagens C, Java, C++, C# e Python

- Veja os vídeos do Prof. Rodrigo Richard
 - <https://youtu.be/-Z7E-TKFeas>
 - <https://youtu.be/hyBXjChj4Pk>
 - <https://youtu.be/MKItMDAGqPA>



Sistema Operacional Linux



Alguns Arquivos Interessantes no Linux

`~/.bash_history` *// Lista de comandos executados*

`~/.vimrc` *// Configurações do VIM*

`~/.profile` *// Configurações do seu Usuário*

Alguns Comandos Interessantes no Modo Shell

man	<i>// manual do linux</i>
exit	<i>// fechar o shell</i>
cd	<i>// mudar de pasta</i>
rm	<i>// apagar</i>
ls (e.g., ls -l -t -h)	<i>// listar pasta</i>
cp	<i>// copiar</i>
mkdir	<i>// criar pasta</i>
clear	<i>// limpar a tela</i>
chmod	<i>// alterar permissão</i>
grep (grep "PALAVRA" ARQUIVO)	<i>// procurar padrão</i>
wget	<i>// efetuar download</i>

Alguns Comandos Interessantes no Modo Shell

diff (diff ARQUIVO1 ARQUIVO2)	<i>// comparar arquivos</i>
ssh	<i>// protocolo ssh</i>
sftp	<i>// protocolo sftp</i>
javac	<i>// compilador Java</i>
java	<i>// interpretador Java</i>
gcc	<i>// compilador C</i>
g++	<i>// compilador C++</i>
sudo	<i>// executar comando como root</i>
apt-get	<i>// manipulação de pacotes apt</i>
tar -zcvf file.tar.gz pasta	<i>// compactar pasta</i>
tar -zxvf file.tar.gz	<i>// descompactar pasta</i>



Alguns Comandos Interessantes no VIM

i	<i>// abrir modo de inserção</i>
ESC	<i>// fechar modo de inserção</i>
:w	<i>// salvar</i>
:q	<i>// sair</i>
:q!	<i>// forçar saída</i>
:wq!	<i>// salvar e forçar a saída</i>
? <u>PALAVRA</u>	<i>// procurar as ocorrências de <u>PALAVRA</u></i>
:%s/ <u>OLD</u> / <u>NEW</u> /gc	<i>// substituir as ocorrências de <u>OLD</u> por <u>NEW</u></i>

Alguns Comandos Interessantes no VIM

yy	<i>// copiar uma linha</i>
<u>NÚMERO</u> yy	<i>// copiar <u>NÚMERO</u> linhas</i>
dd	<i>// excluir uma linha</i>
<u>NÚMERO</u> dd	<i>// excluir <u>NÚMERO</u> linhas</i>
dw	<i>// excluir palavra</i>
<u>NÚMERO</u> dw	<i>// excluir <u>NÚMERO</u> palavras</i>
p	<i>// colar</i>
:sp <u>ARQUIVO</u>	<i>// abrir o arquivo <u>ARQUIVO</u></i>
ww	<i>// alternar entre os arquivos abertos</i>
CTRL+V	<i>// abrir modo de visualização</i>

Agenda

- **Introdução**
 - Comparação
 - Redirecionamento
 - Plataforma
 - Exercícios
- Contextualização: *for AEDs II*
 - **Questões iniciais**
 - Processo em um Exercício de Programação
 - Processo AEDs II de Exercícios

Questões Iniciais

- Quando temos um exercício de programação:
 - Como sabemos que nosso programa está correto?
 - Quanto tempo esperamos pela correção do programa?
 - Alguém já “bateu o olho” no seu programa e identificou um erro?

Agenda

- **Introdução**
- Comparação
- Redirecionamento
- Plataforma
- Exercícios

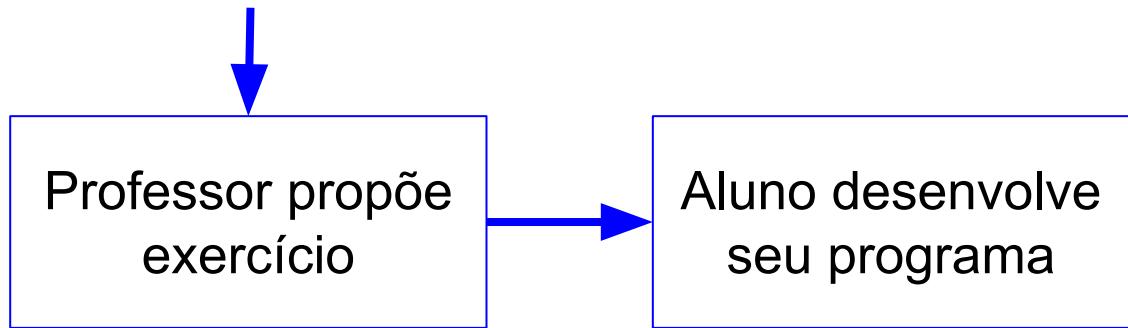
- Contextualização: *for AEDs II*
- Questões iniciais
- **Processo em um Exercício de Programação**
- Processo AEDs II de Exercícios

Processo em um Exercício de Programação

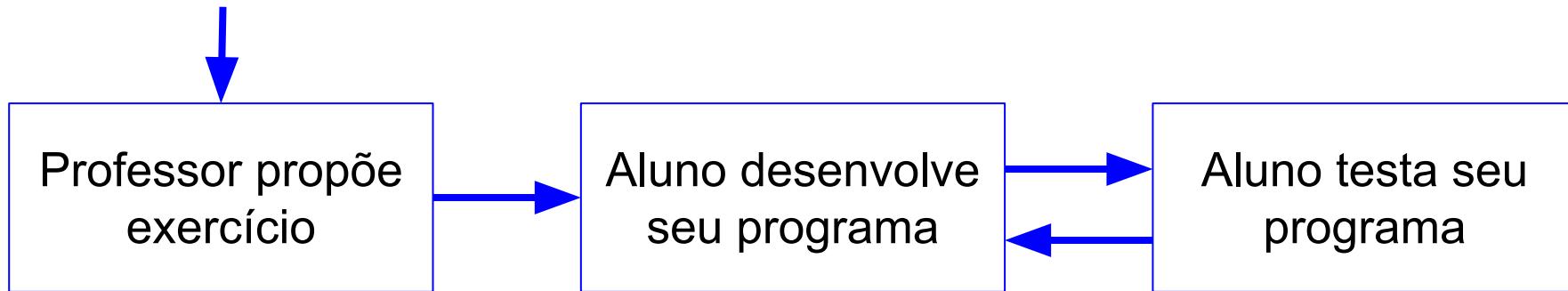


Professor propõe
exercício

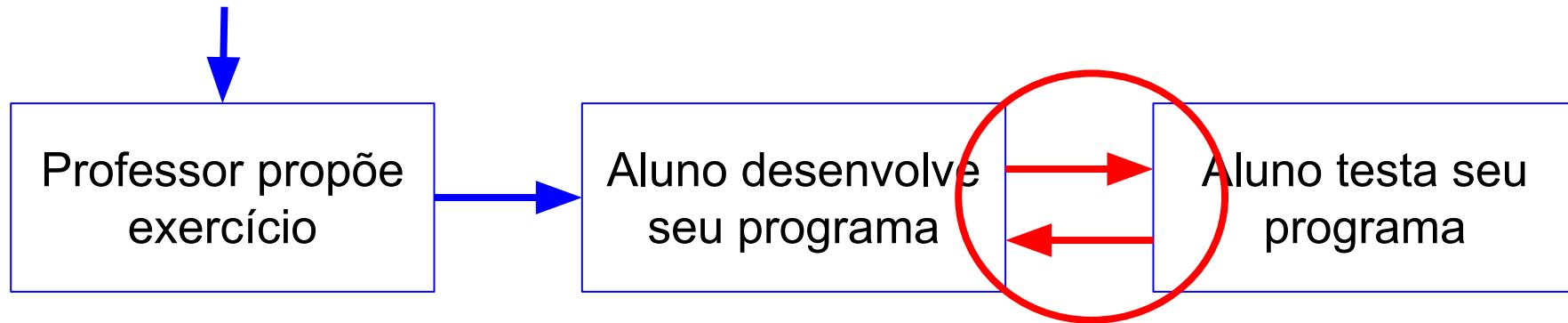
Processo em um Exercício de Programação



Processo em um Exercício de Programação

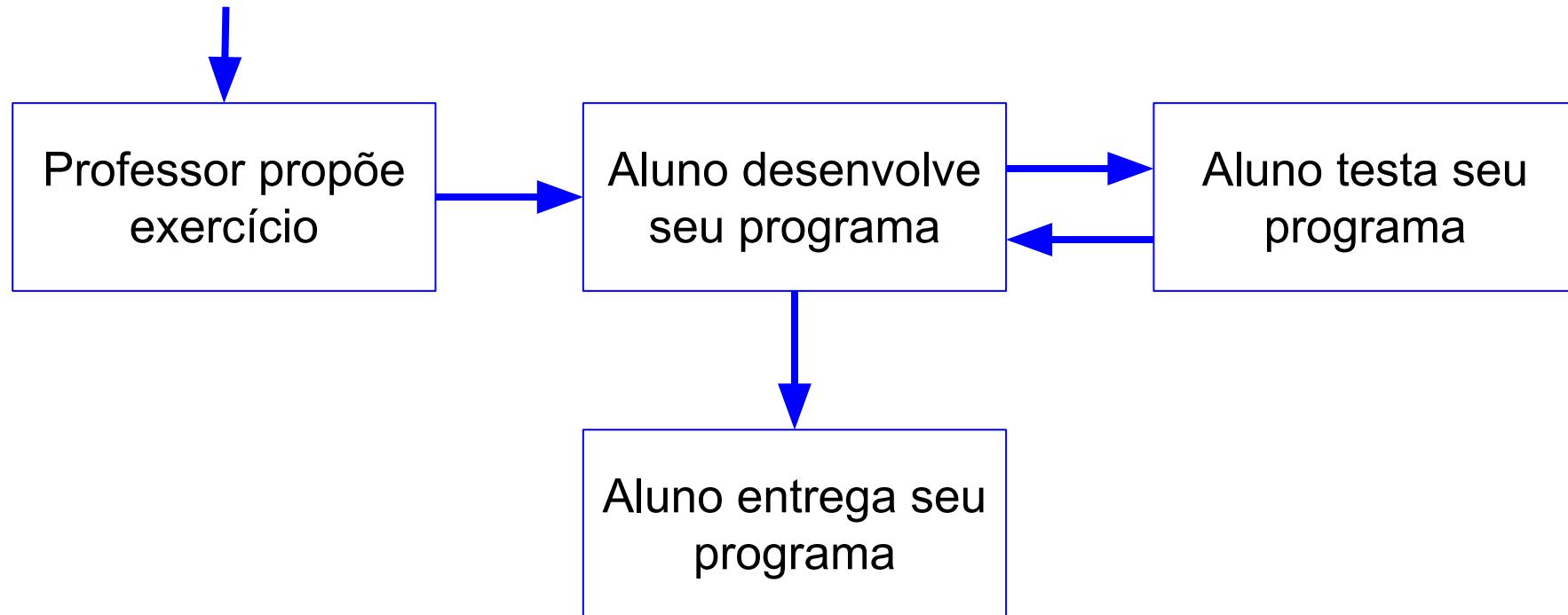


Processo em um Exercício de Programação

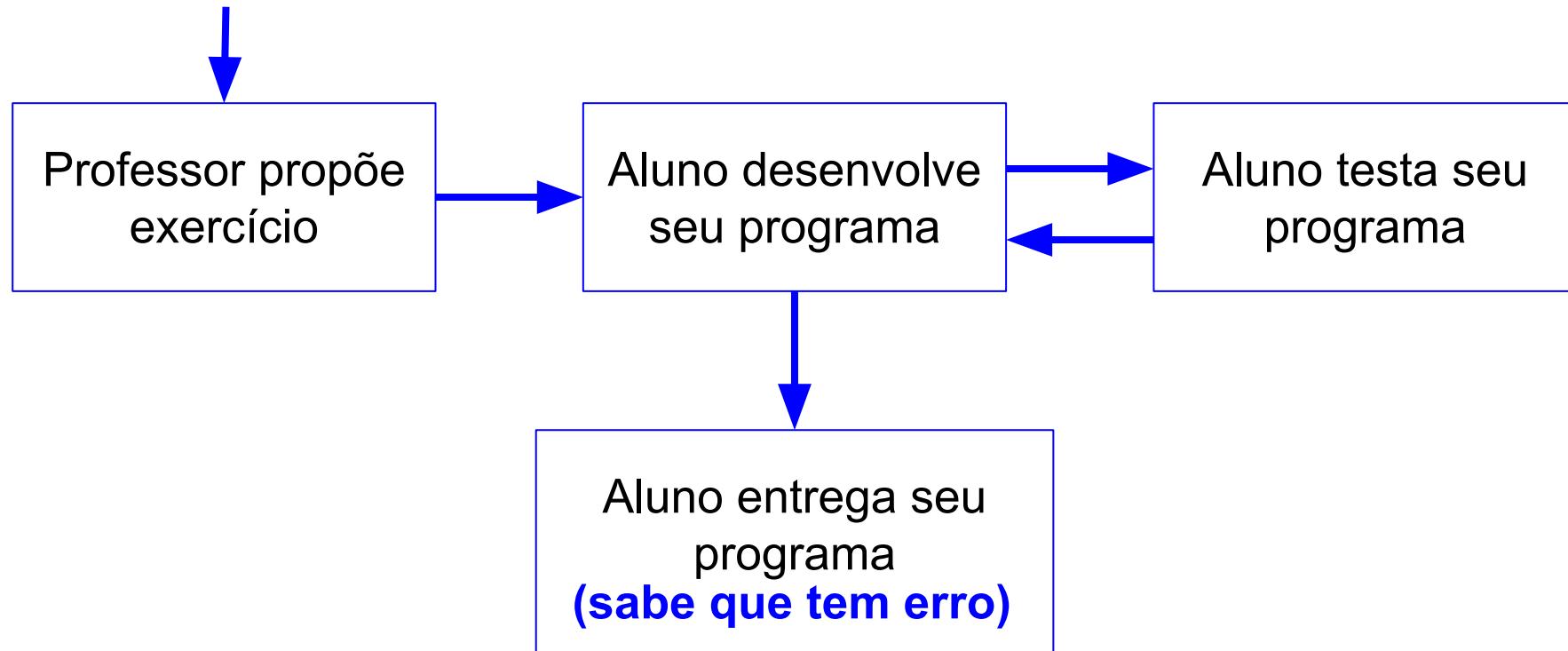


Efeito Lulu Santos: A vida vem em ondas, como um mar / Num indo e vindo infinito

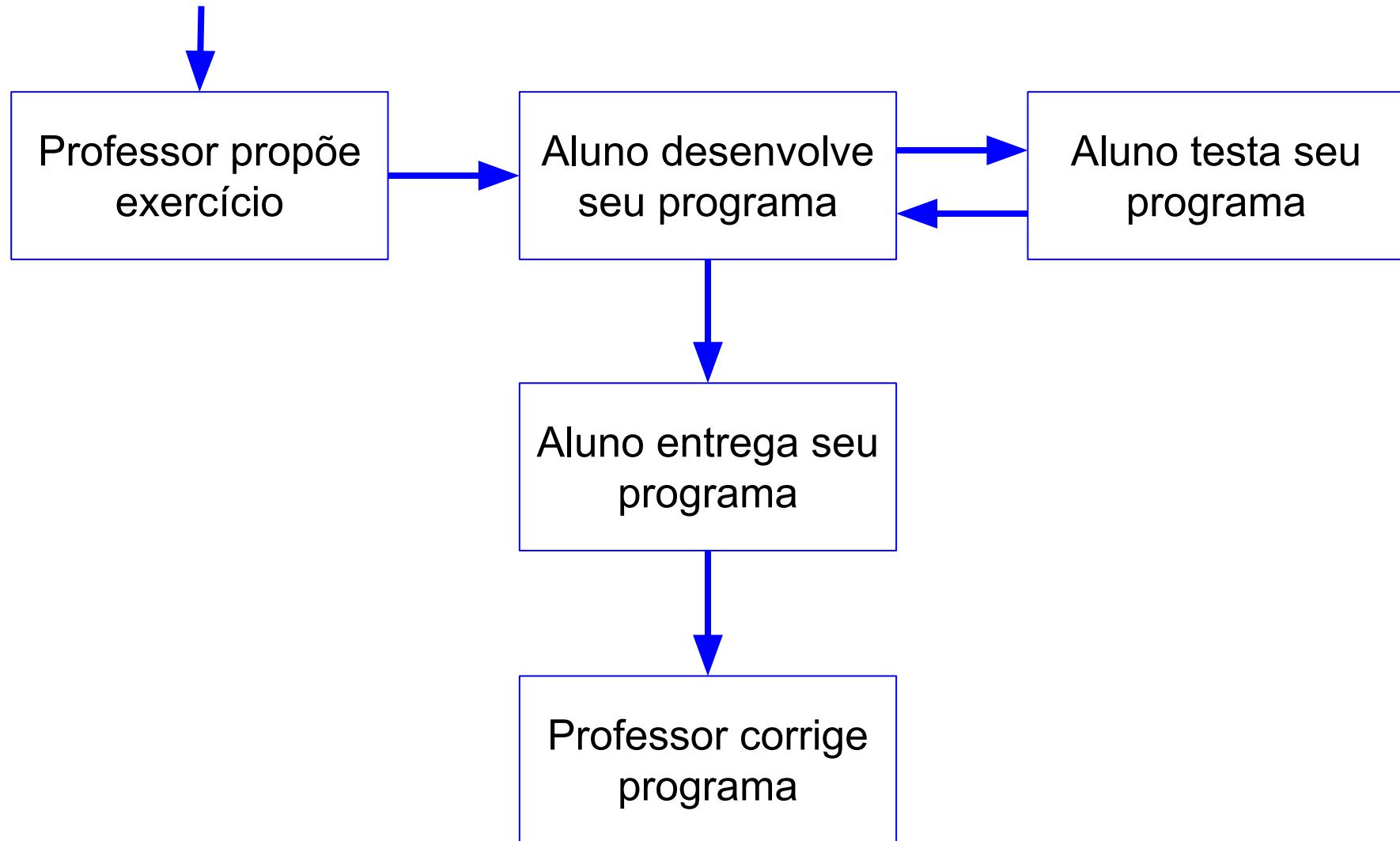
Processo em um Exercício de Programação



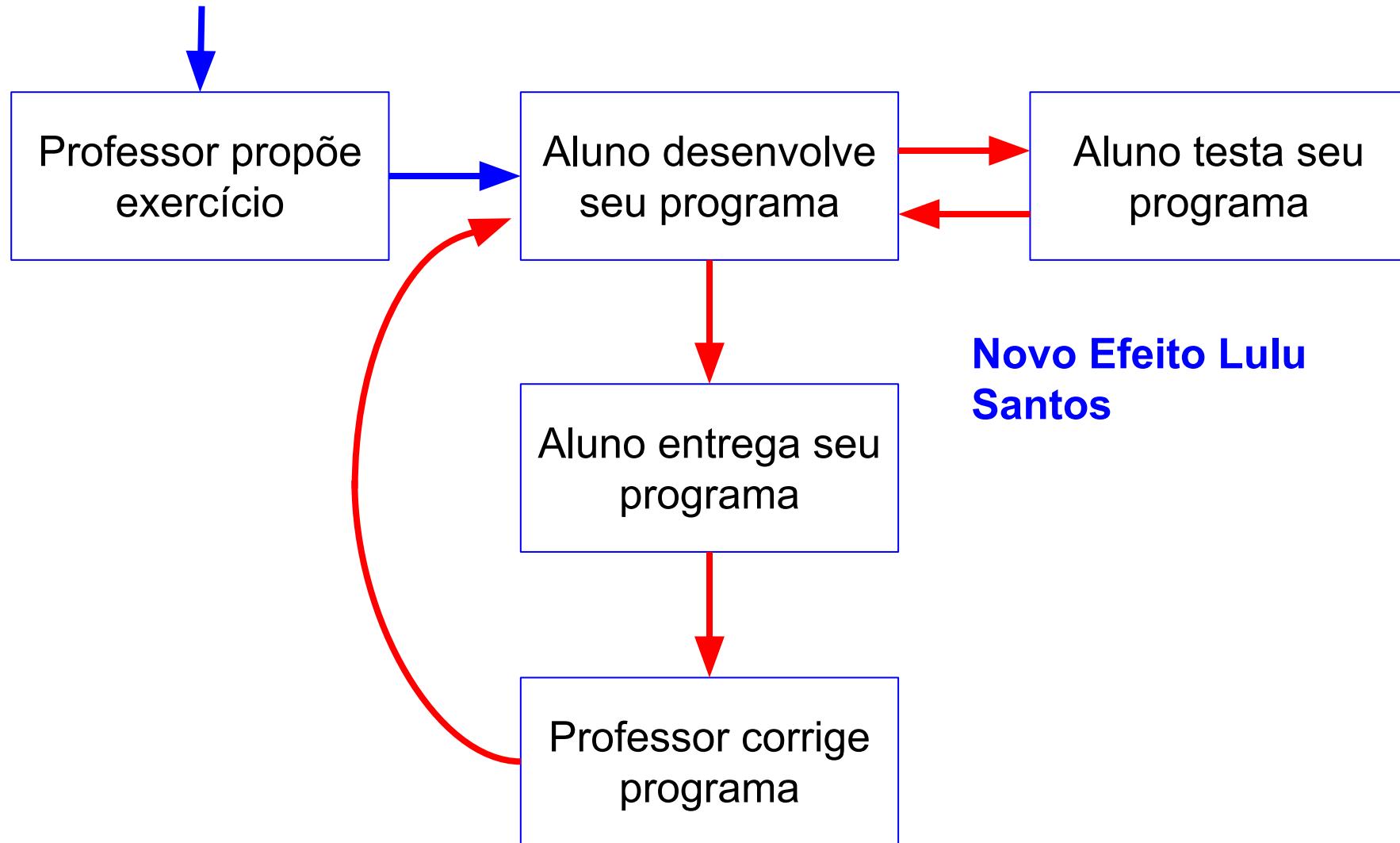
Processo em um Exercício de Programação



Processo em um Exercício de Programação

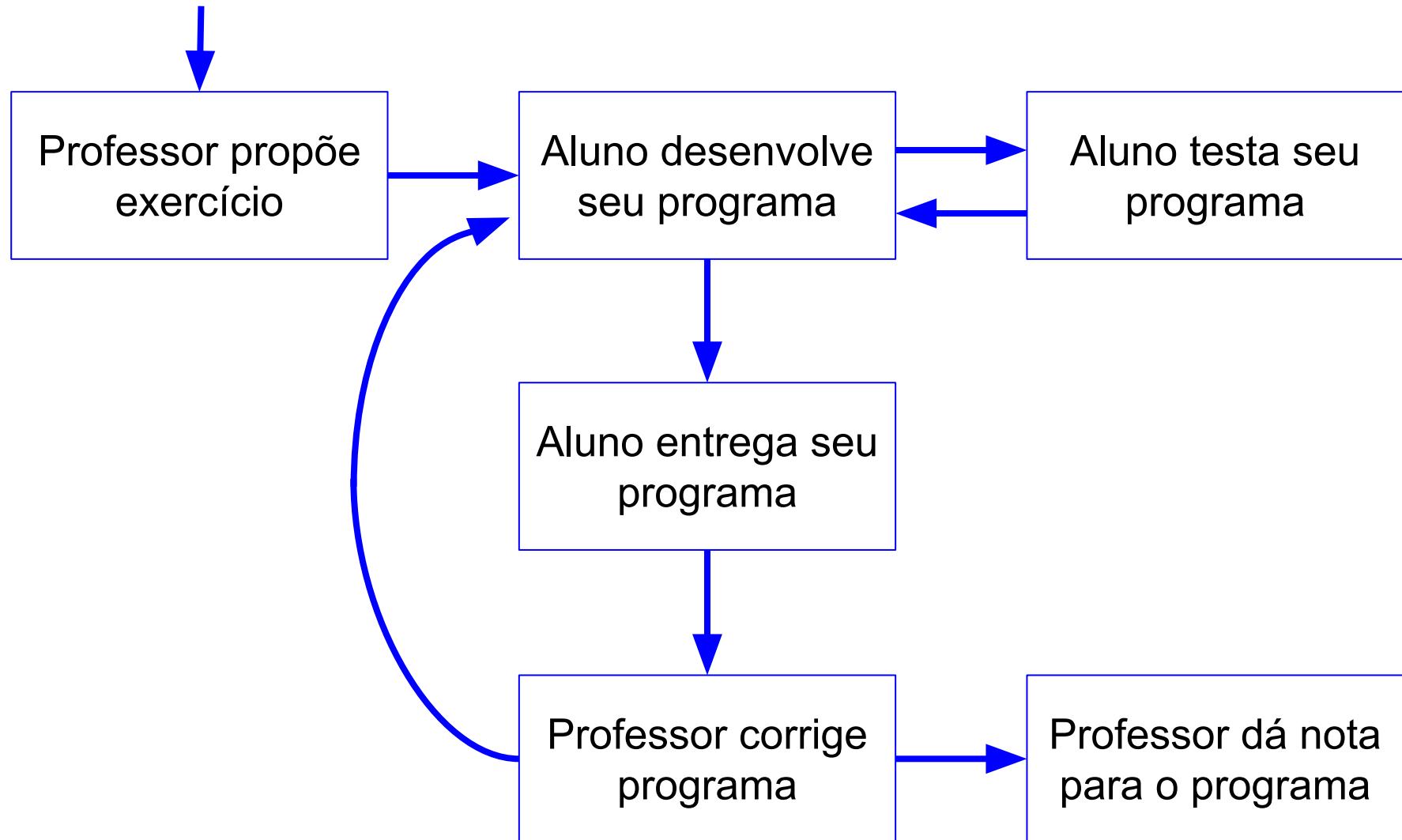


Processo em um Exercício de Programação



Novo Efeito Lulu Santos

Processo em um Exercício de Programação



Alguns Desafios neste Processo

- Aluno precisa testar seus programas
- Aluno depende dos testes realizados pelo professor
- Sobrecarga / ineficiência do professor para realizar correções
- Tempo de resposta da correção não é adequado
- Escalabilidade: Número de testes vs. de alunos

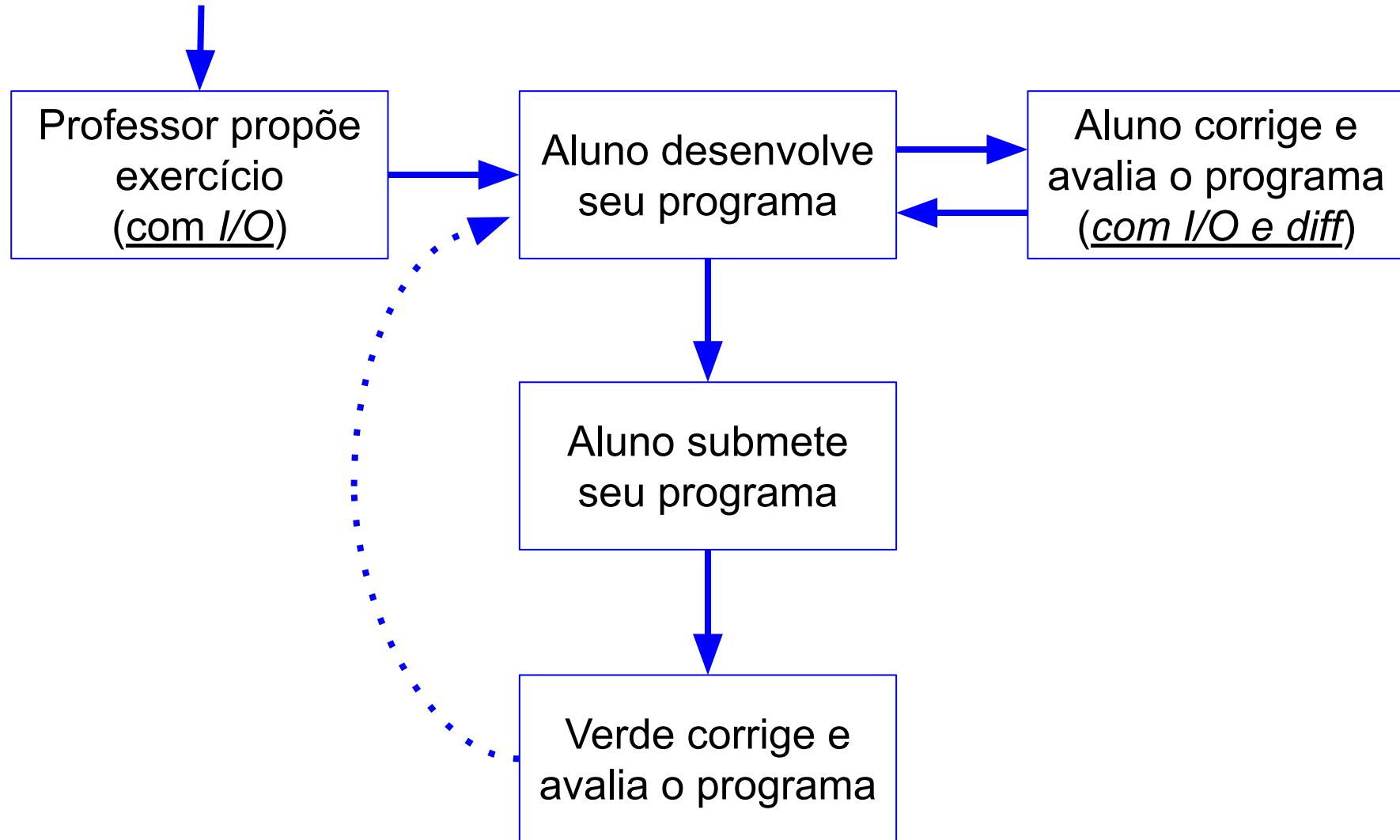
Agenda

- **Introdução**
 - Comparação
 - Redirecionamento
 - Plataforma
 - Exercícios
- Contextualização: *for AEDs II*
 - Questões iniciais
 - Processo em um Exercício de Programação
 - **Processo AEDs II de Exercícios**

Objetivos

- Eliminar o tempo de resposta da correção
- Eliminar a tarefa de correção, otimizando o tempo do professor para atividades mais nobres (e.g., comentários e qualidade de código)
- Aumentar o número de testes

Processo AEDs II de Exercícios



Agenda

- Introdução
- **Comparação de Arquivos com *DIFF***
- Redirecionamento de Entrada e Saída
- Plataforma Verde
- Exercícios

Comando *DIFF*

- Recebe dois arquivos texto e verifica se são iguais



ComputerHope.com

Exercício Resolvido (1): Usando o *DIFF*

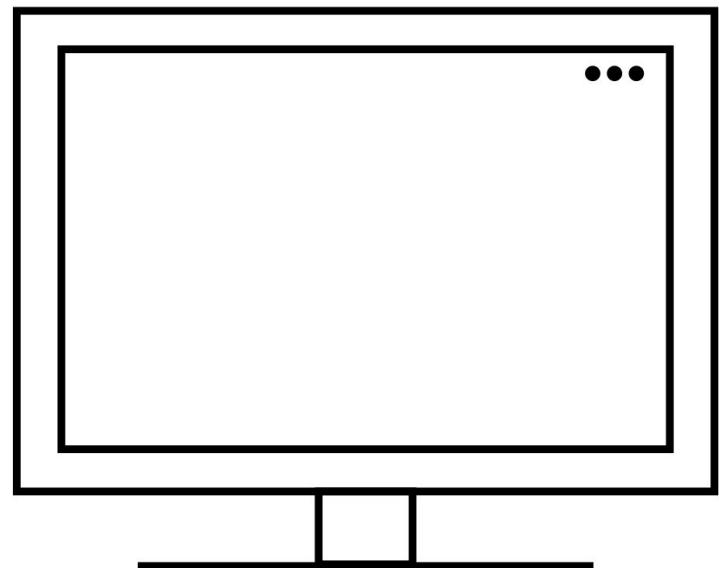
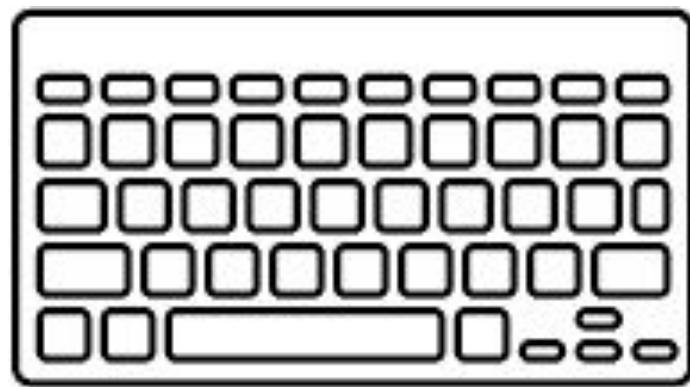
- (1) Leia os arquivos *matriz.c* e *matriz.cc* na pasta 01
- (2) No terminal, compare os arquivos *matriz.c* e *matriz.cc*
comando: *diff -wB matriz.c e matriz.cc*
- (3) No terminal, faça uma cópia do *matriz.c*
comando: *cp matriz.c matriz2.c*
- (4) No terminal, compare os arquivos *matriz.c* e *matriz2.c*

Agenda

- Introdução
- Comparação de Arquivos com *DIFF*
- **Redirecionamento de Entrada e Saída**
- Plataforma Verde
- Exercícios

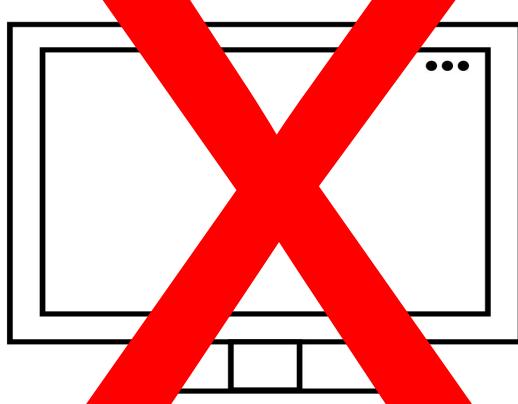
Redirecionamento de Entrada e Saída

- Quando executamos um programa o dispositivo padrão de entrada é o teclado e o de saída, a tela

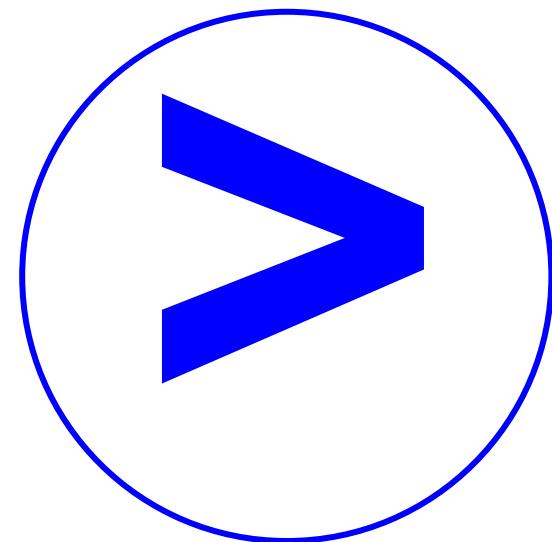
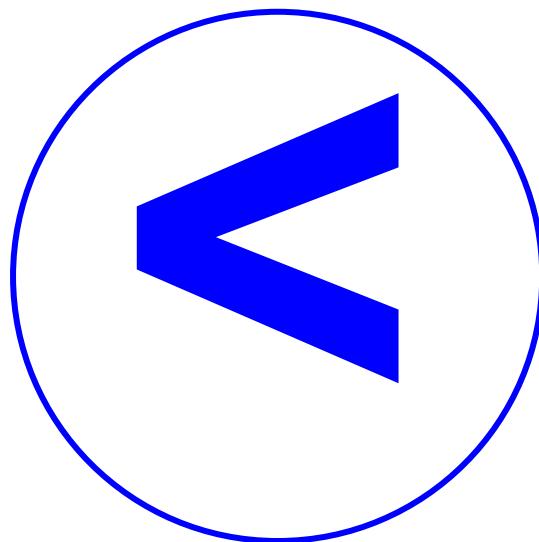


Redirecionamento de Entrada e Saída

- Podemos modificar os dispositivos padrão de entrada e de saída para serem arquivos texto

Entrada	 A black and white line drawing of a computer keyboard. A large red 'X' is overlaid on the image, indicating that keyboard input is not the intended method.	 A black and white line drawing of a scroll or document with the text "Arquivo de entrada" (Input file) written on it.
Saída	 A black and white line drawing of a computer monitor displaying a window with three dots at the top right. A large red 'X' is overlaid on the image, indicating that screen output is not the intended method.	 A black and white line drawing of a scroll or document with the text "Arquivo de saída" (Output file) written on it.

Comandos para Redirecionamento de E/S



Exercício Resolvido (2): Redirecionamento E/S

- Faça um programa em C, C++ e Java para ler e mostrar uma *string*, *int*, *double* e *char*. Execute redirecionando E/S



Exercício Resolvido (2): Redirecionamento E/S

- Faça um programa em C, C++ e Java para ler e mostrar uma *string*, *int*, *double* e *char*. Execute redirecionando E/S



```
char str[100];
int inteiro;
double real;
char caractere;

printf("\nEntre com uma palavra: ");
scanf ("%s", str);
printf("\nEntre com um inteiro: ");
scanf ("%i", &inteiro);
printf("\nEntre com um real: ");
scanf ("%lf", &real);
scanf("%*c");//limpar o buffer
printf("\nEntre com um caractere: ");
scanf ("%c", &caractere);

printf("\nSua string: %s", str);
printf("\nSeu inteiro: %i", inteiro);
printf("\nSeu real: %.1f", real);
printf("\nSeu caractere: %c\n\n", caractere);
```

Exercício Resolvido (2): Redirecionamento E/S

- Faça um programa em C, C++ e Java para ler e mostrar uma *string*, *int*, *double* e *char*. Execute redirecionando E/S



```
char str[100];
int inteiro;
double real;
char caractere;

cout << "\nEntre com uma palavra: ";
cin >> str;
cout << "\nEntre com um inteiro: ";
cin >> inteiro;
cout << "\nEntre com um real: ";
cin >> real;
cout << "\nEntre com um caractere: ";
cin >> caractere;

cout << "\nSua string: " << str;
cout << "\nSeu inteiro: " << inteiro;
cout << "\nSeu real: " << real;
cout << "\nSeu caractere: " << caractere << "\n\n";
```

Exercício Resolvido (2): Redirecionamento E/S

- Faça um programa em C, C++ e Java para ler e mostrar uma *string*, *int*, *double* e *char*. Execute redirecionando E/S



```
String str;
int inteiro;
double real;
char caractere;

System.out.println("Entre com uma palavra: ");
str = MyIO.readString();
System.out.println("Entre com um inteiro: ");
inteiro = MyIO.readInt();
System.out.println("Entre com um real: ");
real = MyIO.readDouble();
System.out.println("Entre com um caractere: ");
caractere = MyIO.readChar();

System.out.println("Sua string: " + str);
System.out.println("Seu inteiro: " + inteiro);
System.out.println("Seu real: " + real);
System.out.println("Seu caractere: " + caractere);
```

Exercício Resolvido (2): Redirecionamento E/S

- Compile o programa anterior



`g++ exemplocpp.c -o exec`



`gcc exemploc.c -o exec`



`javac ExemploMyIO.java`

- Executar o programa



`./exec`



`./exec`



`java ExemploMyIO`

Exercício Resolvido (2): Redirecionamento E/S

- Executar o programa redirecionando **entrada**



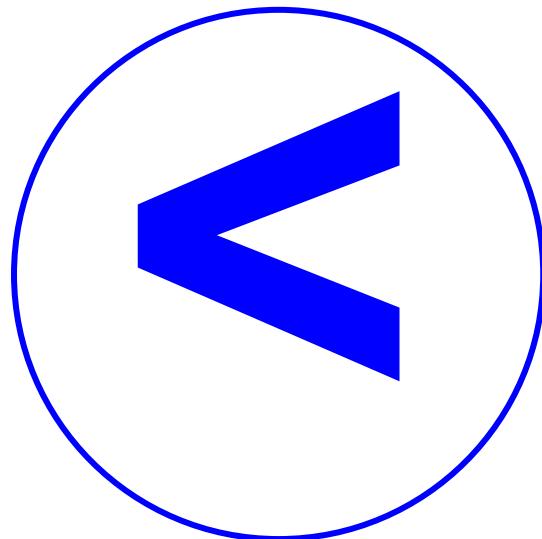
`./exec < pub.in`



`./exec < pub.in`



`java ExemploMyIO < pub.in`



Exercício Resolvido (2): Redirecionamento E/S

- Executar o programa redirecionando entrada e **saída**



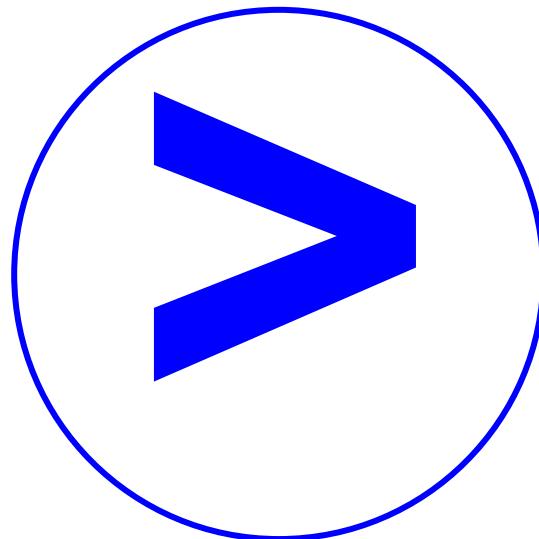
./exec < pub.in > aluno.out



./exec < pub.in > aluno.out



java ExemploMyIO < pub.in > aluno.out



Exercício Resolvido (3): Matriz

- Faça um programa que leia uma matriz quadrada de inteiros e mostre na tela a soma dos elementos de cada coluna

Entrada: Leia um inteiro $n < 10$ (número de linhas e colunas) e os n^2 elementos da matriz

Saída: n números inteiros indicando as somas das colunas

Execute seu programa com a entrada pub.in e compare sua saída com a pub.out

Exercício Resolvido (3): Matriz

- Faça um programa que leia uma matriz quadrada de inteiros e mostre na tela os elementos da matriz.

Entrada: n
colunas

Saída: n
inteiros

4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

pub.in

e leia uma matriz quadrada de inteiros a dos elementos de cada coluna

teiro $n < 10$ (número de linhas e mentos da matriz

pub.out

inteiros indicando 28 32 36 40 das colunas

Execute seu programa e compare sua saída com

com a entrada pub.in e compare sua

Exercício Resolvido (3): Matriz

- Faça um programa que leia uma matriz quadrada de inteiros e mostre na tela a soma dos elementos de cada coluna

Entrada: n
número de linhas e colunas

Saída: n
números inteiros indicando a soma dos elementos das colunas

pub.in

4			
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

e leia uma matriz quadrada de inteiros a dos elementos de cada coluna

teiro $n < 10$ (número de linhas e mentos da matriz

nteiros indicando a soma das colunas

com a entrada pub.in e compare sua

pub.out

28	32	36	40
----	----	----	----

Exercício Resolvido (3): Matriz

- Faça um programa que leia uma matriz quadrada de inteiros e mostre na tela os inteiros da diagonal principal.

Entrada: n
número de linhas e colunas

Saída: n
inteiros indicando as diagonais das colunas

pub.in

4			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			

e leia uma matriz quadrada de inteiros a dos elementos de cada coluna

teiro $n < 10$ (número de linhas e mentos da matriz

inteiros indicando as diagonais das colunas

pub.out

22	32	36	40
----	----	----	----

com a entrada pub.in e compare sua saída com

Exercício Resolvido (3): Matriz

- Faça um programa que leia uma matriz quadrada de inteiros e mostre na tela os inteiros da diagonal principal.
- Entrada:** n = 4
colunas
- Saída:** n = 4
inteiros indicando as colunas
- Execute seu programa
- saída com
-
- Diagram illustrating the mapping of input values to the main diagonal of a 4x4 matrix:
- Input values: 3, 7, 11, 15
 - Output values: 28, 32, 36, 40
 - Mapping: 3 → 28, 7 → 32, 11 → 36, 15 → 40
- The value 36 is circled in red.

Exercício Resolvido (3): Matriz

- Faça um programa que leia uma matriz quadrada de inteiros e mostre na tela os inteiros da diagonal principal.

Entrada: n
número de linhas e colunas

Saída: n
números inteiros das colunas

pub.in

4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

e leia uma matriz quadrada de inteiros a dos elementos de cada coluna

inteiro $n < 10$ (número de linhas e mentos da matriz

inteiros indicando 28 32 36 40 das colunas

com a entrada pub.in e compare sua

saída com

Exercício Resolvido (4): Letras Maiúsculas

- Faça um programa que leia várias frases e mostre o número de letras maiúsculas de cada frase

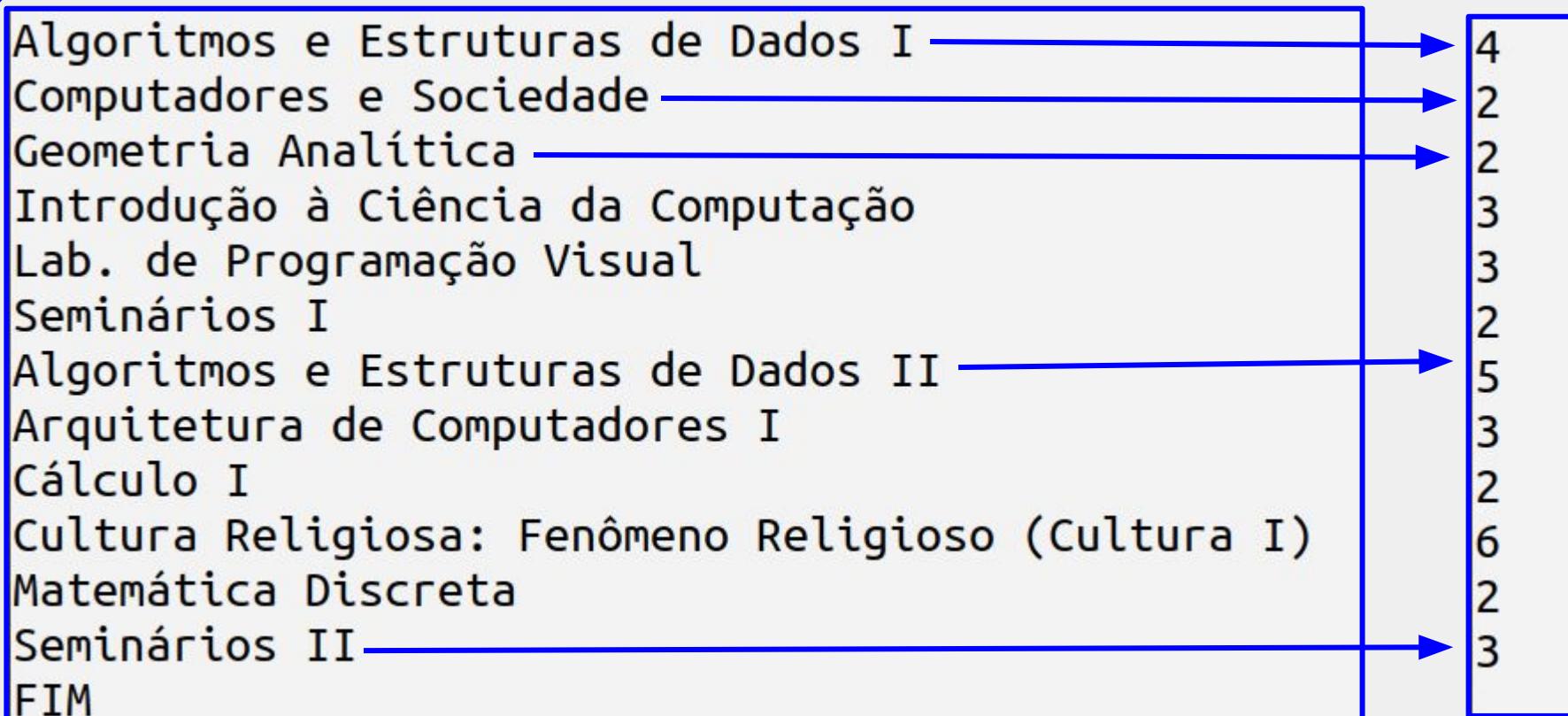
Entrada: composta por várias linhas sendo que a última contém a palavra FIM

Saída: contém um inteiro para cada linha da entrada

Execute seu programa com a entrada pub.in e compare sua saída com a pub.out

Exercício Resolvido (4): Letras Maiúsculas

- Faça um programa que leia várias frases e mostre o número de letras maiúsculas.



Exercícios
saídos

Agenda

- Introdução
- Comparação de Arquivos com *DIFF*
- Redirecionamento de Entrada e Saída
- **Plataforma Verde**
- Exercícios

Plataforma Verde



Plataforma Verde

- **Cadastro de exercício:** nome, pub.in, pub.out, pri.in, pri.out e linguagem



Cadastrar Novo Problema

Nome do Problema:

Entrada Publica: Nenhum arquivo selecionado

Saida Publica: Nenhum arquivo selecionado

Entrada Privada: Nenhum arquivo selecionado

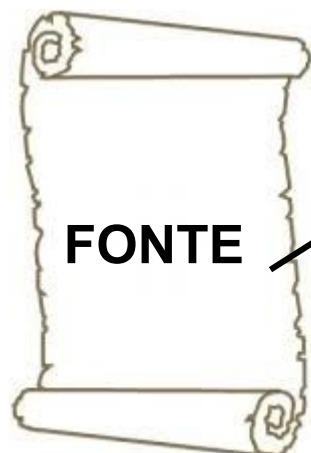
Saida Privada: Nenhum arquivo selecionado

Java

C++

Plataforma Verde

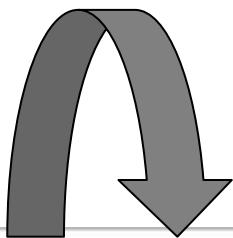
- Submissão de exercício



Atenção: Arquivos em C/C++ devem ter a extensão .cpp

Plataforma Verde

- Verde efetua a correção



CARREGANDO...



- (1) Compilar
- (2) Executar redirecionando entrada (pub.in) e saída (aluno.out)
- (3) Comparar os arquivos aluno.out com (pub.out)
- (4) Executar redirecionando entrada (pri.in) e saída (aluno.out)
- (5) Comparar os arquivos aluno.out com (pri.out)

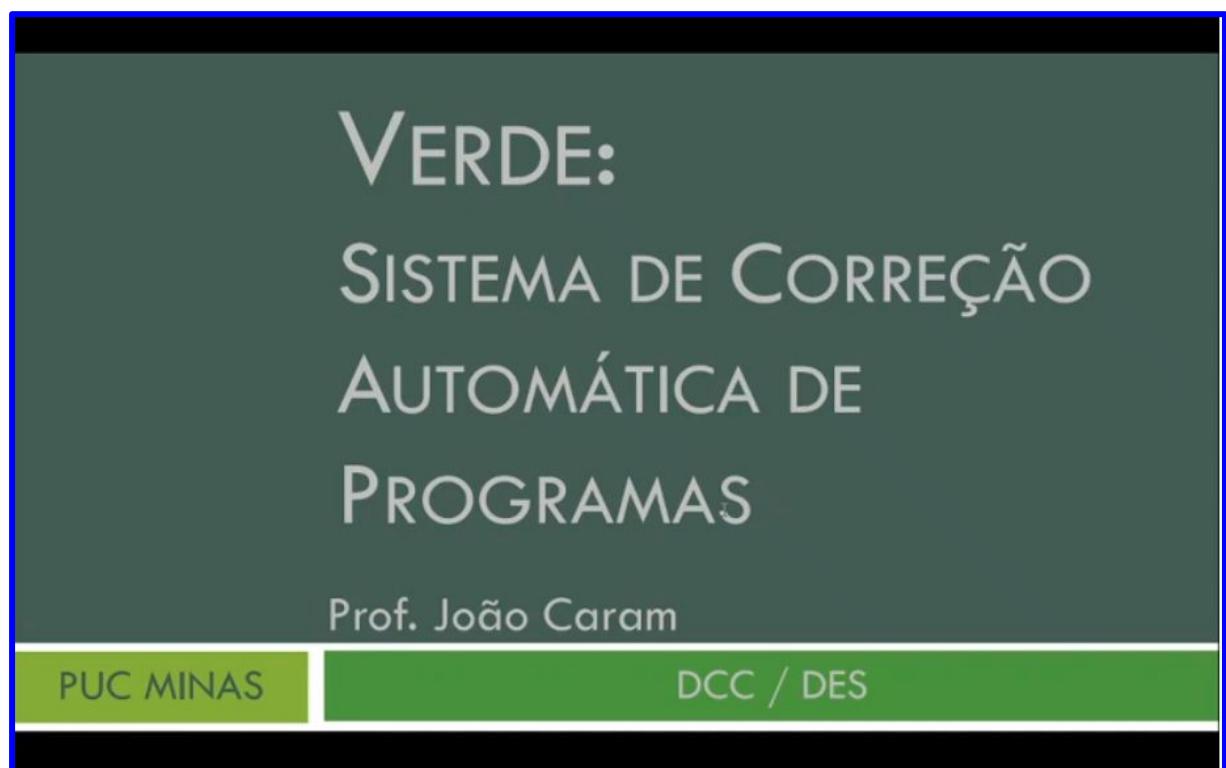
Plataforma Verde

• Submetidos

Arquivos Submetidos											
#	Problema	Data	Arquivo	Publico	Privado	Detalhes	Fonte	Remover			

Verde - Treinamento Alunos

- Veja os vídeos do Prof. João Caram
 - <https://www.youtube.com/playlist?list=PLD8tg9ADnIVjk0Y15Lm-r1NrVUTq9IVxr>



Cadastro no Verde

- Todos os alunos de IC (CC/ES) estão cadastrados na turma de IC do Verde
- maratona.crc.pucminas.br (login *matrícula* e senha *1234*)
- Entre no Verde agora e ganhe seus primeiros totais!!!

Agenda

- Introdução
- Comparação de Arquivos com *DIFF*
- Redirecionamento de Entrada e Saída
- Plataforma Verde
- **Exercícios**

Submeta no Verde

- 1) **Matriz em Java**
- 2) **Matriz em C/C++** (extensão .cpp)
- 3) **Letras Maiúsculas em Java**
- 4) **Letras Maiúsculas em C/C++** (extensão .cpp)

Submeta no Verde

5) **Palíndromo em Java** - Faça um programa recebe várias frases e identifica se elas são palíndromos

Entrada: composta por várias linhas sendo que a última contém a palavra FIM

Saída: para cada linha de entrada, escreva SIM / NÃO, indicando se a linha é um palíndromo

6) **Palíndromo em C/C++** - Refaça a questão anterior

Submeta no Verde

7) Dona Mônica em Java - Dona Mônica é mãe de três filhos que têm idades diferentes. Ela notou que, neste ano, a soma das idades dos seus três filhos é igual à idade dela. Neste problema, dada a idade de dona Mônica e as idades de dois dos filhos, seu programa deve computar e imprimir a idade do filho mais velho. Por exemplo, se sabemos que dona Mônica tem 52 anos e as idades conhecidas de dois dos filhos são 14 e 18 anos, então a idade do outro filho, que não era conhecida, tem que ser 20 anos, pois a soma das três idades tem que ser 52. Portanto, a idade do filho mais velho é 20. Em mais um exemplo, se dona Mônica tem 47 anos e as idades de dois dos filhos são 21 e 9 anos, então o outro filho tem que ter 17 anos e, portanto, a idade do filho mais velho é 21.

Entrada: composta por várias linhas, em cada uma temos 3 inteiros sendo a idade de Dona Mônica e dos seus dois filhos mais novos. A última linha contém apenas um zero

Saída: para cada linha de entrada, escreva um número inteiro indicando a idade do filho mais velho

8) Dona Mônica em C/C++

Submeta no Verde

9) Cometa em Java - O cometa Halley é um dos cometas de menor período do Sistema Solar, completando uma volta em torno do Sol a cada 76 anos. Na última ocasião em que ele tornou-se visível do planeta Terra, em 1986, várias agências espaciais enviaram sondas para coletar amostras de sua cauda e assim confirmar teorias sobre suas composições químicas.

Escreva um programa que, dado o ano atual, determina qual o próximo ano em que o cometa Halley será visível novamente do planeta Terra. Se o ano atual é um ano de passagem do cometa, considere que ele já passou neste ano, logo, nesse caso, considere sempre o próximo ano de passagem, não considerando o ano atual.

Entrada: composta por várias linhas, em cada uma temos um ano entre 2020 e 3000. A última linha contém apenas um zero

Saída: para cada linha de entrada, escreva um número inteiro indicando o ano da próxima passagem

10) Cometa em C/C++