

■ Código e nome da disciplina

### ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

### Semana 1: Tema - 1. SOLUÇÃO DE PROBLEMAS E PENSAMENTO COMPUTACIONAL

3 Objetivos

- Compreender como é possível solucionar problemas do cotidiano com procedimentos ou raciocínio estruturado.

4 Tópicos (i)

### 1.1 INTRODUÇÃO AO PENSAMENTO COMPUTACIONAL

## 5 Procedimentos de ensino-aprendizagem 🔊

Apresentar problemas para serem resolvidos usando instruções simples em linguagem natural. Como sugestão, segue o roteiro abaixo:

#### - Situação-problema:

Suponha uma pessoa hipotética que conhece muito pouco de cozinha (sabe apenas o nome dos objetos, utensílios e ingredientes), e desejamos instruí-la a fritar um ovo. Como podemos descrever esse procedimento usando linguagem natural?

#### - Metodologia:

Iniciar exibindo o vídeo [2] "Exact Instructions Challenge - THIS is why my kids hate me." (Desafio das Instruções ao Pé da Letra - Por isso meus filhos me odeiam) de Josh Darnit.

Destacar que o computador age exatamente desta maneira, seguindo instruções exatas e partir para a criação coletiva em brainstorm para a concepção do passo a passo de como fritar um ovo.

O professor deve apenas mediar as sugestões dos alunos, e se ater a guiá-los no processo de construção coletiva da solução para situação-problema.

Ao final fazer uma validação geral da solução simulando o processo descrito para comprovar sua eficácia.

O professor pode usar [1], capítulo 1, para ampliar o conceito de solução de problemas e introduzir o conceito de algoritmo.

### - Atividade verificadora de aprendizagem:

Em grupos, alunos devem ser encorajados a propor um problema do cotidiano e criar uma solução estruturada para o mesmo, considerando as seguintes restrições: NÃO deve ser uma receita culinária, as instruções serão executadas por alguém que é ignorante no assunto escolhido.

O feedback deve ser no início da próxima aula e deve focar nos seguintes aspectos: se as instruções foram descritas em adequado nível de detalhamento, e analisar possíveis redundâncias (devendo ser baixo, sendo zero o desejável).

## 6 Recursos didáticos

- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com caixa de som e Datashow;
- Software Dev C++ (ou similar) instalado para todos.

# 7 Leitura específica

[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12

[2] Vídeo "Exact Instructions Challenge - THIS is why my kids hate me". (Desafio das Instruções ao Pé da Letra? Por isso meus filhos me odeiam), de Josh Darnit acessível em: https://www.youtube.com/watch?v=cDA3\_5982h8. O vídeo encontra-se em inglês. Ative a legenda, e também a tradução automática para português.

# 8 Aprenda + -

Vídeo: O que a maioria das escolas não ensinam, de Code.org acessível em: <a href="https://www.youtube.com/watch?v=nKIu9yen5nc">https://www.youtube.com/watch?v=nKIu9yen5nc</a>. O vídeo encontra-se em inglês. Ative a legenda, e também a tradução automática para português.

Atividade Autônoma Aura:

Ouestão 1:

Relacione as colunas a seguir:

- A Algoritmo
- B Computador
- C Programa
- 1 ... é um conjunto de instruções que descrevem uma tarefa a ser realizada por um computador.
- 2 ... máquina destinada ao processamento de dados, capaz de obedecer a instruções que visam produzir certas transformações nesses dados para alcançar um fim determinado.
- 3 ... é uma sequência lógica, algo bem parecido com uma receita ou tutorial que mostra passo a passo os procedimentos necessários para a resolução de uma tarefa.

- A) A1 B2 C3
- B) A1 B3 C2
- C) A2 B1 C3
- D) A2 B3 C1
- E) A3 B2 C1

### Questão 2:

No vídeo "Exact Instructions Challenge - THIS is why my kids hate me ", de Josh Darnit o autor do vídeo faz um desafio aos seus filhos, no qual eles devem escrever para o pai um conjunto de instruções para que ele possa fazer um sanduiche. A experiência se mostra bem demorada, precisando de vária tentativas até que o sanduíche finalmente ficasse pronto. Isso se deve ao fato de Josh seguir as instruções de seus filhos de forma literal, sem nenhum tipo de favorecimento (o que acabou irritando o caçula).

Neste vídeo as crianças, Josh, e as instruções, podem abstratamente serem comparados respectivamente a:

- A) Computador, Programador, Programa
- B) Computador, Programa, Programador
- C) Programador, Programa, Computador
- D) Programador, Computador, Programa
- E) Computador, Programa, Programador



1 Código e nome da disciplina 💷

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 2: Tema - 1. SOLUÇÃO DE PROBLEMAS E PENSAMENTO COMPUTACIONAL

3 Objetivos

- Apresentar as diferentes formas de expressar soluções computacionais baseadas em fluxogramas, algoritmos escritos em linguagem natural e em linguagem de programação;

4 Tópicos (i

1.2 FERRAMENTAS PARA REPRESENTAÇÃO DE SOLUÇÕES 1.3 PSEUDOCÓDIGO

## 5 Procedimentos de ensino-aprendizagem 🏐

Esta aula aborda a importância da representação formal de solução de problemas, e apresentar as três formas usadas (pseudocódigos, fluxogramas e linguagem de programação). Como sugestão, segue o roteiro abaixo:

#### - Situação-problema:

Tem-se em mãos duas jarras: uma com capacidade para quatro litros; e outra, com capacidade para três litros. As jarras não possuem marcações intermediárias, de forma que, se qualquer uma das jarras não estiver completamente cheia, não será possível saber quanta água haverá dentro dela. Existe uma torneira disponível para encher as jarras e um ralo que permite que a água seja jogada fora. As únicas instruções disponíveis são:

a. Encher completamente uma jarra, utilizando a torneira;

- b. Passar a água de uma jarra para outra; e
- c. Esvaziar completamente uma jarra, jogando a água que estiver dentro dela no ralo.

O objetivo é indicar como as jarras devem ser manipuladas, de modo que, ao final da manipulação, a jarra com capacidade para quatro litros de tenha exatamente dois litros de água.

#### - Metodologia:

Aula expositiva inicia com a exibição da solução em linguagem natural por parte do professor (verificar o anexo da aula 2).

Para facilitar a compreensão dos alunos antes de descrever a solução em linguagem, passamos por estágios intermediários, com aproximação gradual à linguagem do computador.

Para isso, deve-se traduzir a solução de linguagem natural primeiramente para pseudocódigo, depois para fluxograma e finalmente para linguagem de programação (usando C++ no paradigma estruturado,

sem muita preocupação com a sintaxe neste momento). Em cada estágio o professor deve expor os conceitos necessários e a importância de cada abordagem.

- Atividade verificadora de aprendizagem:

Utilizar a lista de exercícios disponibilizada no Anexo da Aula 2 para que os alunos exercitem a solução dos problemas usando as 3 metodologias apresentadas. A verificação deverá ocorrer com a geração e disponibilização do gabarito pelo docente, esclarecendo apenas as eventuais dúvidas.

# 6 Recursos didáticos 👙

- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.

# 7 Leitura específica

[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 16 a 29. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12

# 8 Aprenda + -

Alguns sites com jogos de raciocínio lógico e que podem ser usados como apoio para a aula:

- 1. Jogo das Jarras de água, Disponível em: https://www.novelgames.com/en/jars/
- 2. Jogo de travessia do rio, disponível em: <a href="https://rachacuca.com.br/jogos/o-lobo-e-a-ovelha/">https://rachacuca.com.br/jogos/o-lobo-e-a-ovelha/</a>
- 3. Jogo do Teste de Einstein, disponível em: <a href="http://rachacuca.com.br/teste-de-einstein/">http://rachacuca.com.br/teste-de-einstein/</a>

### Atividade Autônoma Aura:

#### Questão 1:

Um algoritmo pode ser escrito utilizando diferentes linguagens. Relacione o tipo de algoritmo com a respectiva descrição:

- A Linguagem Natural
- B Linguagem Gráfica
- C Pseudolinguagens
- 1 Foi criada com o objetivo de facilitar o aprendizado da lógica de algoritmos ou lógica de programação, devendo o aprendiz focar na resolução do problema ou lógica do problema, e não na estrutura e formalismo de como representá-lo de forma clara e sem ambiguidade.
- 2 A escolha das palavras e termos utilizados dependem diretamente da pessoa que está escrevendo e da compreensão de quem lê. Assim, uma mesma palavra pode ter diversos significados.
- 3 Baseadas em imagens e esquemas, de forma a tentar facilitar a visualização e o entendimento das etapas e processos.

- A) A1, B2, C3
- B) A2, B3, C1
- C) A3, B2, C1
- D) A1, B3, C2

### E) A2, B1, C3

#### Questão 2:

A cerca de Fluxogramas considere as seguintes afirmativas:

- I É um diagrama padronizado utilizado para representações esquemáticas.
- II É baseado na norma ISO 5807, que os padroniza, facilitando seu aprendizado e compreensão.
- III É composto basicamente de figuras geométricas que representam as possíveis diferentes etapas de um processo, conectadas por setas que indicam a sequência das etapas e suas transições.

Pode-se dizer que:

- A) Apenas as afirmativas I e II estão certas
- B) Apenas as afirmativas I e III estão certas
- C) Apenas as afirmativas II e III estão certas
- D) Todas as afirmativas estão certas
- E) Nenhuma afirmativa está certa



1 Código e nome da disciplina 🕕

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 3: Tema - 2. TIPOS DE DADOS, EXPRESSÕES, OPERADORES E TABELA VERDADE

- 3 Objetivos
  - Conhecer os tipos básicos de dados fornecidos por uma linguagem de programação;
  - Compreender o que são variáveis e constantes;
- 4 Tópicos (i

#### 2.1 TIPOS DE DADOS

## 5 Procedimentos de ensino-aprendizagem 🌇

Esta aula aborda os tipos de dados existentes e os conceitos relacionados a variáveis e constantes. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

O computador é uma máquina digital e basicamente só trabalha com zeros e uns. No entanto os programas de computador podem resolver problemas envolvendo os mais diversos tipos de dados. Os programas podem calcular o somatório de valores, ou a média desses valores. Pode localizar o nome de uma pessoa numa lista. Que tipo de dados são compreendidos pela máquina e como podem ser trabalhados?

#### - Metodologia:

Iniciar coma exposição dos tipos de dados existentes na linguagem C/C++ (int, double, char, bool e mencionar o agregado []) explicando que tipos de valores es representam. Explicar os conceitos de variáveis e constantes, citando exemplos e caracterizando a diferença entre eles. O professor deve usar pequenos programas em C++ para demonstrar o armazenamento de valores específicos em variáveis e constantes de tipos de dados diferentes.

- \* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.
- Atividade verificadora de aprendizagem:

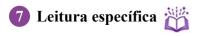
Aplicar exercício em forma de questionário objetivo em que os alunos devem responder o melhor tipo de variável para o tipo de dados que deve ser armazenado nela. Aconselha-se o uso da ferramenta

Kahoot (https://kahoot.com/).

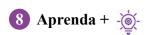
Exemplos: Que tipo de dado seria usado para representar a idade de uma pessoa? Que tipo de dado seria usado para representar a altura de uma pessoa? Que tipo de dado seria usado para representar o sexo de uma pessoa? Que tipo de dado seria usado para representar se uma pessoa está presente ou não?

# 6 Recursos didáticos 🧬

- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.



[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 32 a 39. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12



Artigo: C Plus Plus: Variáveis e tipos. Acessível em: <a href="http://www.cplusplus.com/doc/tutorial/variables/">http://www.cplusplus.com/doc/tutorial/variables/</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Artigo: C Plus Plus: Constantes. Acessível em: <a href="http://www.cplusplus.com/doc/tutorial/constants/">http://www.cplusplus.com/doc/tutorial/constants/</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Artigo: CpPreference.com: Tipos de Dados do C++. Acessível em:

https://pt.cppreference.com/w/cpp/data types

#### Atividade Autônoma Aura:

#### Ouestão 1:

Considere as seguintes afirmativas:

- I Uma variável é um elemento sintático do algoritmo que tem como funcionalidade armazenar um determinado valor.
- II Uma Constante é um elemento sintático do algoritmo que tem como funcionalidade armazenar mais de um determinado valor.
- III O valor associado à variável depende do tipo de informação que se deseja manipular.
- IV O valor associado à constante não depende do tipo de informação que se deseja manipular. Está(ão) correto(s):

#### Alternativas:

- A) apenas o item I.
- B) apenas o item III.
- C) apenas os itens I e III.
- D) apenas os itens II e IV.
- E) os itens I, II e III.

#### Questão 2:

Considere as seguintes afirmativas:

I - Para que possamos utilizar uma variável, um algoritmo ou programa, precisamos

primeiro declará-la.

- II Para que possamos declarar uma variável, primeiro precisamos definir que tipo de dado ela irá armazenar.
- III Para se declarar uma variável, usamos a sintaxe [tipo de dado] [nome da variável]
- IV Os tipos de dados primitivos são aqueles fornecidos nativamente nas linguagens de programação, e são obrigatoriamente os mesmos em todas as linguagens. Está(ão) correto(s):

( )

- A) apenas os itens I e II.
- B) apenas os itens I e III.
- C) apenas os itens I e IV.
- D) apenas os itens I, II e III.
- E) apenas os itens I, II e IV.



1 Código e nome da disciplina 💷

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema

Semana 4: Tema - 2. TIPOS DE DADOS, EXPRESSÕES, OPERADORES E TABELA VERDADE

- 3 Objetivos
- Conhecer e compreender o uso dos operadores matemáticos, lógicos e relacionais de uma linguagem de programação;
- Entender a ordem de precedência entre os operadores de uma linguagem de programação;
- 4 Tópicos (j

#### 2.2 OPERADORES

## 5 Procedimentos de ensino-aprendizagem 🏐

Esta aula aborda os operadores matemáticos, lógicos, relacionais e de atribuição, bem como a precedência ente eles. Como sugestão, segue o roteiro abaixo:

- Situação problema:

Um computador é basicamente uma calculadora com muito mais poder. Seu nome inclusive "Computador" vem do verbo computar, que quer dizer literalmente calcular. Que tipo de operações um computador pode realizar e como elas se relacionam?

- Metodologia:

Iniciar com a exposição dos operadores matemáticos existentes na linguagem C/C++ (+, -, \*, /, %), seguir para exposição dos operadores relacionais existentes na linguagem C/C++ (>, <, >=, <=, ==, !=), e finalizar com a exposição dos operadores lógicos existentes na linguagem C/C++ (!, &&, ||, ^). Sempre explicando que tipos de valores podem ser operados e suas peculiaridades. A seguir mencionar como funciona a atribuição de valores a variáveis e finalizar explicando a ordem de precedência existente entre as operações e como podem ser subvertidas com uso de parênteses.

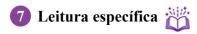
- \* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.
- Atividade verificadora de aprendizagem:

Os alunos devem realizar um programa para calcular a média aritmética entre 3 notas. Deve-se destacar que os alunos podem observar resultados diferentes se usarem ou não parênteses para

expressar a função de cálculo da média. Então deve-se fazer um desafio para que o cálculo possa ser feito de forma correta sem uso de parênteses.

## 6 Recursos didáticos 🤌

- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.



[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 46 a 54.Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12



Artigo: C Plus Plus: Operadores. Acessível em: <a href="http://www.cplusplus.com/doc/tutorial/operators/">http://www.cplusplus.com/doc/tutorial/operators/</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português). Artigo: CpPreference.com: Precedência de Operadores C++. Acessível em: <a href="https://pt.cppreference.com/w/cpp/language/operator-precedence">https://pt.cppreference.com/w/cpp/language/operator-precedence</a>

### Atividade Autônoma Aura:

#### Ouestão 1:

Considerando o trecho de código abaixo, é seguro dizer que, ao fim da execução do referido trecho, o resultado da variável c será:

int a, b, c;

a = 7;

b = 2;

c = a / b;

Alternativas:

- A) 10
- B) 3,5
- C) 14
- D) 1
- E) 3

#### Questão 2:

Considerando que os valores de a = 5; b = 10; c = 20; Qual das expressões abaixo teria sua avaliação como "falso"?

- A) (a > b) || (c >= b)
- B) (b > a) && (c >= a)
- C) (! (a > b)) && (c >= b)
- D) ( ( a != b ) || ( a == c) ) && (b != c)
- E) (c!=a) && (b==c)



1 Código e nome da disciplina 📺

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

## Semana 5: Tema - 3. COMANDOS DE ENTRADA E SAÍDA

- 3 Objetivos
- Compreender como funcionam e como atribuir expressões com variáveis;
- Conhecer as tabelas verdade dos operadores lógicos mais comuns;
- 4 Tópicos j

## 3.1 ATRIBUIÇÃO

## 5 Procedimentos de ensino-aprendizagem 🌇

Esta aula aborda o uso atribuições com expressões de variáveis e o uso da tabela da verdade. Como sugestão, segue o roteiro abaixo:

### - Situação problema:

Você está em uma loja de brinquedos e visualiza um homem acompanhado de uma criança, muito provavelmente pai e filho, e escuta o seguinte diálogo entre eles:

Criança: Tá chegando meu aniversário, posso fazer minha escolha?

Adulto: Pode. Você quer o vídeo game OU a bicicleta?

C: Que legal! Quero os dois então!

A: Mas assim não pode. Só pode uma coisa OU a outra. Decida-se ou ficará sem nenhum...

Note que houve um equívoco de entendimento relacionado com o conceito de operadores lógicos que estudamos na aula passada. Qual foi o equívoco? E, como o pai deveria ser expressar para que o filho percebesse que tem apenas uma alternativa como presente?

#### - Metodologia:

Iniciar com a exposição das tabelas verdades dos operadores lógicos existentes na linguagem C/C++ (!, &&, ||, ^) e frisar a diferença principal entre as noções de AND, OR e XOR. Em lógica o AND tem o mesmo significado que E em português; o conceito do XOR na lógica é o equivalente ao OU do português; e o OR não tem equivalência linguística. A seguir realizar um programa em que se utilize variáveis como os elementos de uma função, permitindo a leitura de diferentes valores. Intuitivamente pode-se realizar um programa que responde a uma linha de uma tabela verdade (sugere-se começar com o E) pedindo o valor das duas entradas a serem informadas pelos alunos e a resposta processada pelo programa.

\* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação

em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.

- Atividade verificadora de aprendizagem:

Como atividade os alunos devem adaptar o programa da tabela verdade do E construído junto com o professor para responder também a tabela verdade do OU e do Ou Exclusivo.

Nesta aula o professor deve passar a segunda Atividade Acadêmica Avaliativa.

A atividade consiste em uma pesquisa em livros (pode ser da nossa bibliografía básica, complementar ou outros, mas deve ser em livros) de lógica de programação ou linguagens de programação. Cada aluno deverá escolher dois exercícios do livro e deverá:

- Fornecer a referência acadêmica completa e correta de cada exercício escolhido.
- Fornecer a solução do mesmo.
- Propor uma alteração no enunciado original que altere a resolução do mesmo.
- Fornecer a solução para o exercício alterado em Português estruturado, C/C++ e fluxograma para o mesmo.
- A entrega deverá se dar por meio de arquivo digital a ser entregue pelo SAVA ou similar.

# 6 Recursos didáticos

- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.

# 7 Leitura específica 📸

[1] FORBELLONE, A.L. V; EBERSPACHER, H. Lógica de programação. 3ª Ed. São Paulo: Makron Books, 2005. Páginas 22 a 24. Disponível em:

https://plataforma.bvirtual.com.br/Acervo/Publicacao/323

# 8 Aprenda + -

Artigo: Ferrara, Mario. Maxi Educa: Tabelas-Verdade, onde utilizarei isto em minha vida? Disponível em: <a href="https://blog.maxieduca.com.br/tabela-verdade-uso/">https://blog.maxieduca.com.br/tabela-verdade-uso/</a>

Atividade Autônoma Aura:

### Questão 1:

Considere as seguintes afirmativas:

- I A tabela verdade da conjunção (E) retorna verdade somente quanto todos os valores analisados são verdade.
- II A tabela verdade da disjunção (OU) retorna verdade somente quando os valores analisados são diferentes (verdade ou falso; falso ou verdade).
- III A tabela verdade da negação (NÃO) inverte o valor lógico recebido.

Pode-se dizer que:

- A) Apenas as afirmativas I e II estão certas
- B) Apenas as afirmativas I e III estão certas
- C) Apenas as afirmativas II e III estão certas
- D) Todas as afirmativas estão certas

## E) Nenhuma afirmativa está certa

## Questão 2:

Considerando o trecho de código abaixo, é seguro dizer que, ao fim da execução do referido trecho, os resultados das variáveis a e b serão respectivamente:

bool a, b;

int c, d, e, f;

c = 5;

d = 2;

e = 5 / 2;

f = 5 % 2;

a = e > f;

b = e == f;

- A) 2.5 e 2
- B) 2 e 1
- C) 2 e 2.5
- D) Verdadeiro e Falso
- E) Verdadeiro e Verdadeiro



1 Código e nome da disciplina 📺

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 6: Tema - 3. COMANDOS DE ENTRADA E SAÍDA

- 3 Objetivos
- Compreender o funcionamento dos comandos de entrada e saída da linguagem de programação C/C++;
- Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (i
  - 3.2 SAÍDA DE DADOS 3.3 ENTRADA DE DADOS
- 5 Procedimentos de ensino-aprendizagem

Esta aula aborda os comandos de entrada e de saída da linguagem C/C++ (CIN / COUT / SCANF/PRINTF) e a prática de exercícios de teste de mesa. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

Um programa é tão útil quanto a percepção do usuário dizer que ele o é. Note que mesmo que um programa faça todos os cálculos e procedimentos para resolver um problema se o mesmo ao final do processamento não retornar uma resposta que seja compreendida pelo usuário o mesmo será dado como inútil ou errado. Como podemos com os comandos vistos até o momento fazer uma boa comunicação com o usuário do nosso programa?

### - Metodologia:

Esta será uma aula de formalização de conceitos dado que os alunos já devem ter tido contato com comandos de entrada e saída de dados em aulas anteriores. Deve-se iniciar com um brainstorm para verificar a percepção dos alunos de o que são e quais são os comandos de entrada e saída conhecidos. O professor deve formalizar a sintaxe de todos eles (CIN / COUT / SCANF/PRINTF), mas enfatizar o uso do CIN/COUT contra o SCANF/PRINTF por conta de simplicidade didática.

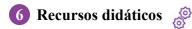
O professor deverá então fazer o programa 1 da lista de exercícios de exemplo em anexo a esta aula: calcular a área e o perímetro de uma sala retangular, sendo que as dimensões da sala são fornecidas pelo usuário. Ao final realizar uma dinâmica mostrando através de um teste de mesa (popularmente conhecido como chinês) o funcionamento do programa.

\* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação

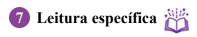
em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.

- Atividade verificadora de aprendizagem:

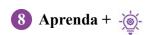
Dividir a turma em grupos e aplicar para cada grupo um dos exercícios disponíveis na lista do arquivo em anexo para esta aula. Ao final da solução cada grupo deve eleger um aluno para apresentar o programa para os demais e outro para fazer o teste de mesa do mesmo.



- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.



[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 39 a 46. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12



Artigo: C Plus Plus: CIN. Acessível em: <a href="http://www.cplusplus.com/reference/iostream/cin/?kw=cin">http://www.cplusplus.com/reference/iostream/cin/?kw=cin</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Artigo: C Plus Plus: COUT. Acessível em: <a href="http://www.cplusplus.com/reference/iostream/cout/?kw=cout">http://www.cplusplus.com/reference/iostream/cout/?kw=cout</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Artigo: C Plus Plus: SCANF. Acessível em: <a href="http://www.cplusplus.com/reference/cstdio/scanf/?kw=scanf">http://www.cplusplus.com/reference/cstdio/scanf/?kw=scanf</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Artigo: C Plus Plus: PRINTF. Acessível em: <a href="http://www.cplusplus.com/reference/cstdio/printf/?kw=printf">http://www.cplusplus.com/reference/cstdio/printf/?kw=printf</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

#### Atividade Autônoma Aura:

#### Questão 1:

Considere as seguintes afirmativas:

- I Com o comando utilizado para entrada de dados é possível transferir uma informação digitada pelo usuário com o teclado para uma determinada variável no escopo do algoritmo.
- II O comando de saída de dados pega o conteúdo da memória interna e joga no dispositivo de saída padrão, logo ele irá imprimir ou mostrar o conteúdo na tela do usuário.

Pode-se dizer que:

Alternativas:

- A) A afirmativa I está incorreta e a afirmativa II está correta.
- B) A afirmativa II está incorreta e a afirmativa I está correta.
- C) A afirmativa II está correta e a afirmativa I está correta.
- D) A afirmativa I está correta se a afirmativa II está correta.
- E) A afirmativa I está incorreta e a afirmativa II está incorreta.

#### Questão 2:

O Teste de Mesa é um processo manual que é utilizado para validar a lógica de um determinado

algoritmo. Ele é utilizado principalmente em algoritmos quando a linguagem utilizada não possui nenhuma ferramenta automatizada de depuração. Como as linguagens de programação costumam possuir tais ferramentas, é mais comum utilizá-las a fazer o teste de mesa propriamente dito, embora para quem ainda é iniciante, não se recomenda utilizá-lo, visto que provavelmente não terá domínio sobre a ferramenta de depuração.

Assim, faça um teste de mesa que determine o que será exibido na tela pelo trecho de programa listado abaixo:

```
abaixo:
int a = 8, b = 2, c, d;
cin >> c;
d = a + b;
c = a - b;
c = a;
a = b;
b = c;
cout << "A: " << a << "B: " << b;

Alternativas:
A) A: 2 B: 8
B) A: 8 B: 2
C) A: a B: b
D) A: 10 B: 6
E) A: b B: a
```



1 Código e nome da disciplina 🕕

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 7: Tema - 4. ESTRUTURAS DE DECISÃO

## 3 Objetivos

- Escrever estruturas de seleção simples, baseando-se nas regras da linguagem de programação C/C++, para a elaboração de soluções algorítmicas compreensíveis ao computador;
- Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (i

#### 4.1 DECISÃO SIMPLES E COMPOSTA

## 5 Procedimentos de ensino-aprendizagem 🇿

Esta aula aborda os conceitos de estruturas de decisão (seleção) utilizando o bloco de decisão simples. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

Imagine que você está parado em um ponto de ônibus e gostaria de pegar uma linha específica em um carro que não esteja lotado. Assim quando um ônibus se aproxima você começa a análise da situação: "Esse é o ônibus que estou esperando?", "Ainda tem espaço nele?", "Ao fazer sinal de parada, ele parou?" Note que para cada uma das questões podemos ter uma resposta positiva ou negativa. E em cada caso só a positiva nos interessa. Como podemos programar esta estratégia?

#### - Metodologia:

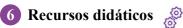
Expor a sintaxe do bloco de decisão simples (IF) frisando que ele delimita um bloco de comandos que só será realizado caso o teste lógico por ele mapeado forneça valor final de Verdadeiro (true). Realizar exercícios de comparação simples, como ler o valor de uma conta de luz e, caso o valor seja maior do que R\$ 150,00, apresentar a mensagem: "Você está gastando muito", e, caso contrário, não exiba mensagem alguma. É importante que o professor medie a solução de forma que os alunos percebam que não é necessário usar a instrução SE NÃO. Uma outra alternativa excelente é a de fazer um programa que verifica a possível ocorrência de um erro, como divisão por zero, e no caso forçar a saída do programa para não propagar uma execução errada; ou solicitar duas notas (de AV1 e AV2) e dar um alerta apenas no caso de o aluno ser obrigado a realizar a AV3 para ser aprovado. Espera-se que após estas dinâmicas o aluno perceba que ele poderia tomar decisões análogas para a situação testada pela expressão lógica caso ela retornasse falso. Como avisar que o aluno já está

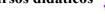
aprovado no casso do exemplo da AV1 e AV2. Este será o impulso para a próxima aula.

\* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.

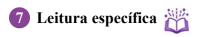
- Atividade verificadora de aprendizagem:

Guiar os alunos a solução de dois problemas: ler um número e dizer se é par ou ímpar, e dado um número dizer se ele é positivo, negativo ou neutro; sempre usando apenas o IF sem o ELSE. Depois realizar brainstorm com a turma se eles veem uma maneira mais interessante de resolver estes problemas. Essa colocação deve juntar-se com o início da próxima aula.





- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.



[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 56 a 62. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12

# 

Artigo: C Plus Plus: Declarações e controle de fluxo. Acessível em: http://www.cplusplus.com/doc/tutorial/control/ (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Atividade Autônoma Aura:

#### Ouestão 1:

Considerando o trecho de código a seguir qual valor deve ser digitado a fim de que seja impresso na tela apenas Mensagem B?

```
int b;
cin >> b;
if (b >= 10) { cout << "Mensagem A"; }
cout << "Mensagem B";</pre>
```

Alternativas:

- A) 13
- B) 10
- C) 3
- D) b
- E) Verdade

Questão 2:

Considere que o trecho de programa abaixo foi executado 4 vazes, cada uma com os seguintes valores sendo digitados como o valor de x: 3, -7, 0, 21.

```
cin >> x;
if (x > 0) { cout << "P";}
if (x == 0) { cout << "O";}
if !(x > 0) { cout << "N";}
```

O que seria impresso na tela?

- A) PONP
- B) P N O P
- C) P N ON P
- D) PN N ON PN
- E) NPON



1 Código e nome da disciplina 📺

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema

Semana 8: Tema - 4. ESTRUTURAS DE DECISÃO

- 3 Objetivos
- Escrever estruturas de decisão compostas e encadeadas, baseando-se nas regras da linguagem de programação C/C++, para a elaboração de soluções algorítmicas compreensíveis ao computador; Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (j

#### 4.1 DECISÃO SIMPLES E COMPOSTA

## 5 Procedimentos de ensino-aprendizagem 🇿

Esta aula aborda os conceitos de estruturas de decisão (seleção), agora utilizando blocos de decisão compostos e seleção encadeada. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

As estruturas de decisão são a primeira ferramenta revolucionária no tangente a representação de estratégias de resolução de problemas computacionais pois permitem realizar uma operação chamada controle de fluxo. Se assemelha a uma bifurcação em uma estrada, onde se escolhemos ir pela esquerda não podemos acompanhar o caminho da direita, e vice-versa. Nos testes lógicos podemos sempre ter duas respostas (verdadeiro (true) ou falso (false)), definindo assim duas possibilidades para qualquer teste. Baseado no exposto, será que conseguimos melhorar nossos exercícios da aula passada?

#### - Metodologia:

Expor a sintaxe do bloco de decisão composta (IF-ELSE) frisando que ele delimita dois blocos de comandos mutuamente exclusivos e que o primeiro que só será realizado caso o teste lógico por ele mapeado forneça valor final de Verdadeiro (true), enquanto o segundo, demarcado pelo ELSE só será realizado caso o teste lógico por ele mapeado forneça valor final de Falso (false). Refazer com os alunos os exercícios de verificação de paridade e sinal do fim da última aula. Em seguida o professor deve demonstrar que os programas passaram a ter desempenho melhor por conseguir chegar ao mesmo resultado com um número menor de testes lógicos, já que num mesmo teste se mapeia duas possibilidades de execução.

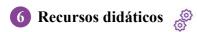
Espera-se que após estas dinâmicas o aluno perceba que ele deve valorizar a estrutura de decisões

composta sempre que possível dado que o status de falso também agrega valor estratégico no programa.

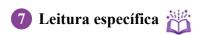
Finalizar exibindo o trailer de [2] "BLACK MIRROR: BANDERSNATCH | Oficial Netflix (2018)" que mostra uma obra cinematográfica interativa em que o telespectador pode decidir o rumo da mesma com decisões simples (se sim e não) sobre os eventos da trama. No Tecnoblog, na página [3] "Quantos finais tem Black Mirror: Bandersnatch?" encontra-se um fluxograma que demonstra todas as possibilidades da obra.

- \* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.
- Atividade verificadora de aprendizagem:

Mediar os alunos na solução do problema: fazer um programa que solicita o calor das três avaliações de um aluno (AV1, AV2 e AV3) e verifica seu status seguindo regras similares às da nossa IES (ignorando o fator de presença).



- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.



- [1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 63 a 73. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12
- [2] BLACK MIRROR: BANDERSNATCH | Oficial Netflix (2018) Acessível em: https://www.youtube.com/watch?v=i9Wc82gCZLE
- [3] Cossetti, Melissa Cruz. Tecnoblog: Quantos finais tem Black Mirror: Bandersnatch? Acessível em: https://tecnoblog.net/273570/quantos-finais-tem-black-mirror-bandersnatch/



- C Plus Plus: Declarações e controle de fluxo. Acessível em: <a href="http://www.cplusplus.com/doc/tutorial/control/">http://www.cplusplus.com/doc/tutorial/control/</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).
- Hypeness: Como desbloquear o game secreto jogável escondido no final de Black Mirror Bandersnatch. Acessível em: <a href="https://www.hypeness.com.br/2019/01/como-desbloquear-o-game-secreto-jogavel-escondido-no-final-de-black-mirror-bandersnatch/">https://www.hypeness.com.br/2019/01/como-desbloquear-o-game-secreto-jogavel-escondido-no-final-de-black-mirror-bandersnatch/</a>
- Wikibooks: Programar em C++/ Decisão e controle de fluxo/If-Else. Acessível em: <a href="https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Decis%C3%A3o\_e\_controle\_de\_fluxo#if-else">https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Decis%C3%A3o\_e\_controle\_de\_fluxo#if-else</a>

Atividade Autônoma Aura:

## Questão 1:

Considerando o trecho de código a seguir, o que deve ser escrito dentro dos parênteses do comando if a fim de que se reflita que um aluno é aprovado apenas quando consegue que sua média esteja acima de 6 (seis)?

```
media = (a + b + c) / 3;

if (______) { cout << "Aprovado"; }

else { cout << "Reprovado";}

Alternativas:
```

- A) media  $\geq = 6$
- B) media > 6
- C) media > 5.9
- D) media > 6.1
- E) media == 6

### Questão 2:

Considere que o trecho de programa abaixo confere uma senha numérica e, baseado na sua composição, identifica o usurário. Assim, qual deve a ser a senha do usuário Bruce?

```
cin >> senha;
if ((senha \% 2) == 0)
if (senha > 0)
cout << "Bruce";</pre>
} else {
cout << "Clark";
} else {
if (senha > 0)
cout << "Diana";</pre>
} else {
cout << "Barry";</pre>
}
}
Alternativas:
A) 1342
B) -4231
C) 0
D) 2561
```

E) -8246



1 Código e nome da disciplina 💷

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema

Semana 9: Tema - 4. ESTRUTURAS DE DECISÃO

- 3 Objetivos
  - Escrever estruturas de decisão com múltiplas alternativas, baseando-se nas regras da linguagem de programação C/C++, para a elaboração de soluções algorítmicas compreensíveis ao computador;
  - Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (j)

## 4.2 DECISÃO ENCADEADA, ANINHADA E MÚLTIPLAS ALTERNATIVAS

## 5 Procedimentos de ensino-aprendizagem 🏐

Esta aula aborda os conceitos de estruturas de decisão (seleção), agora utilizando decisão com múltiplas alternativas. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

Um problema frequente que ocorre é quando temos uma situação podendo assumir diversos valores diferentes, em que, para cada valor, teremos uma ação associada. Para resolver esse problema de imediato, pensaríamos em um conjunto de IFs aninhados. Mas essa seria a melhor opção?

#### - Metodologia:

Expor a sintaxe do bloco de decisão com múltiplas alternativas (SWITCH) frisando que ele controla um grande bloco formado por N sub blocos, onde cada sub bloco corresponde a um mapeamento de uma dos possíveis valores a serem assumidos pela variável tratada do grande bloco. Não esquecer de mencionar a possibilidade de execução de um bloco padrão (default).

O professor deve mediar um brainstorm a fim de os alunos perceberem a limitação do comando em apenas realizar teste de igualdade, e que com isso sua utilização será sempre privilegiada a estruturas de funcionamento similares a menus.

Para a demonstração sugere-se o programa que solicita o número de um dos meses do ano, e informa para o usuário o nome do mesmo por extenso.

Espera-se que após estas dinâmicas o aluno perceba que consegue simular qualquer bloco de SWITCH com um conjunto de IFs, mas o contrário nem sempre é verdade.

\* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no

aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.

- Atividade verificadora de aprendizagem:

Mediar os alunos na solução do problema: construir um programa que solicita ao usuário dois valores e uma operação aritmética (+,-,\*,/, RAIZ) e retorne o resultado da operação selecionada sobre os números informados. O professor deve orientar a turma para contornar os possíveis cenários de erro, como operações inválidas, divisão por zero e o cálculo da raiz que deve ser para apenas um valor.

## 6 Recursos didáticos

- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.

# 7 Leitura específica 📸

[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 74 a 81. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12

# 8 Aprenda + -

Artigo: C Plus Plus: Switch Case. Acessível em: <a href="http://www.cplusplus.com/forum/beginner/5152/">http://www.cplusplus.com/forum/beginner/5152/</a> (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

- Wikibooks: Programar em C++/ Decisão e controle de fluxo/Switch. Acessível em: <a href="https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Decis%C3%A3o\_e\_controle\_de\_fluxo#switch">https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Decis%C3%A3o\_e\_controle\_de\_fluxo#switch</a>

#### Atividade Autônoma Aura:

#### Questão 1:

Para o uso da estrutura escolha de caso (switch-case), precisamos estar atentos a exigências, como as listadas a seguir:

- I A variável tem que ser a mesma em todos os testes.
- II A variável tem que ser do tipo enumerável: inteira ou de um caractere.
- III O operador relacional pode ser o de igualdade ou o de diferença.
- IV A opção de retorno padrão é obrigatória.

Podemos dizer que estão corretas as afirmativas:

- A) As alternativas I, II e III estão corretas.
- B) As alternativas II, III e IV estão corretas.
- C) As alternativas I e II estão corretas.
- D) As alternativas II e III estão corretas.
- E) As alternativas III e IV estão corretas.

#### Questão 2:

Considerando o trecho de código a seguir, o que seria impresso pelo programa?

```
op = 2;

switch ( op ) {

case 1 : cout << "um "; break;

case 2 : cout << "dois ";

case 4 : cout << "tres "; break;

default: cout << "zero "; break;

}

Alternativas:

A) um

B) dois

C) tres

D) zero

E) dois tres
```



1 Código e nome da disciplina 🕕

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 10: Tema - 5. ESTRUTURAS DE REPETIÇÃO

- 3 Objetivos 🍥
- Escrever estruturas de repetição com variável de controle, baseando-se nas regras da linguagem de programação C/C++, para a elaboração de soluções algorítmicas compreensíveis ao computador;
  Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (i

#### 5.1 COMANDO FOR

## 5 Procedimentos de ensino-aprendizagem 🍿

Esta aula aborda os conceitos de estruturas de repetição, iniciando utilizando a repetição com variável de controle. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

Existe a possibilidade de esquecer uma condição que leva a saída do laço de repetição pode fazer com que o programa fique em um looping infinito. Este tipo de processamento faria o programa rodar sem parar chegando a ocupar o processador 100% do tempo para um processamento que eventualmente está completo ou errado, e devemos ter consciência desta situação pois o percentual de uso de tempo e de processador é um dos itens monitorados pela T.I. verde. Há algum outro tipo de estrutura de repetição que pode nos ajudar a evitar o problema do looping infinito?

#### - Metodologia:

Expor a sintaxe do bloco de repetição com variável de controle (FOR) demonstrando sua estrutura diferenciada do WHILE, possuindo três elementos de controle dentro dos parênteses (inicialização, teste e incremento). Não esquecer de mencionar a vantagem de se utilizar esta estrutura quando se sabe a priori o volume de dados a ser processado, ou seja, o número de passos que o programa deverá repetir. O professor deve mediar um processo de criação de um programa simples como a exibição de uma sequência numérica como todos os pares de 0 a 100 e solicitar a implementação da solução usando as três estruturas de repetição conhecidas pelos alunos (FOR, WHILE e DO..WHILE). Espera-se que após esta dinâmica o aluno perceba os nuances entre as estruturas e que a maioria dos problemas que solicita uma estrutura de repetição pode ser resolvido com a utilização de qualquer uma delas.

- \* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.
- \* Esta aula precede a aula de conteúdo digital, é importante que o professor lembre aos alunos que devem consumir o material digital antes da próxima aula.
- Atividade verificadora de aprendizagem:

Apresentar para os alunos a famosa sequência de Fibonacci e solicitar que criem um programa que imprime os N primeiros elementos da sequência, sendo o N um número informado pelo usuário.

Nesta aula o professor deve apresentar também a Atividade Acadêmica Avaliativa que contemplará 5 pontos para AV2. Os alunos devem organizar-se em pequenos grupos (no máximo de 3 alunos - se possível com o critério do método 300 - detalhes em [2]) e criar um pequeno jogo que possa ser executado com o ferramental disponível na disciplina. No caso de sugestões são possíveis jogo da senha, batalha naval, forca, jogo da velha, entre vários outros. A entrega deve ocorrer até a AV2 em mídia digital pelo SAVA ou similar.

## 🌀 Recursos didáticos 🦃



- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.

# 7 Leitura específica

[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 86 a 91 e de 103 a 106. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12

[2] Metodologia: método Trezentos pode ser aplicado à educação básica. Disponível em: https://desafiosdaeducacao.grupoa.com.br/metodo-trezentos-ensino-basico/

# 

Artigo: C Plus Plus: Declarações e controle de fluxo. Acessível em:

http://www.cplusplus.com/doc/tutorial/control/ (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

- Wikibooks: Programar em C++/Estruturas de repetição/For. Acessível em: https://pt.wikibooks.org/wiki/Programar em C%2B%2B/Estruturas de repeti%C3%A7%C3%A3o#F <u>or</u>

Atividade Autônoma Aura:

Ouestão 1:

Considere o trecho de código a seguir:

```
int i, r = 1;
for (i = 5; i > 0; i--) \{ r = r * i; \}
```

```
Pode-se dizer que este trecho:
A) Imprime a sequência de números entre 5 e 0;
B) Calcula uma P.A. de início em 5 e razão -1
C) Calcula uma P.A. de início em 5 e razão 1
D) Calcula uma P.G. de início em 5 e razão -1
E) Calcula 5!, ou seja, o fatorial de 5
Questão 2:
Considere uma estrutura de comando de repetição com variável de controle:
para ([1];[2];[3]) {
// bloco de comandos
Os elementos demarcados com os números 1, 2 e 3 são respectivamente:
A) valor inicial; expressão de teste; incremento
B) expressão de teste; incremento; valor inicial
C) incremento; valor inicial; expressão de teste
D) valor inicial; incremento; expressão de teste
E) incremento; expressão de teste; valor inicial
```



■ Código e nome da disciplina

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema

### Semana 11: Tema - 5. ESTRUTURAS DE REPETIÇÃO

3 Objetivos

- Escrever estruturas de repetição com teste no início, baseando-se nas regras da linguagem de programação C/C++, para a elaboração de soluções algorítmicas compreensíveis ao computador;
- Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (j

## 5.2 REPETIÇÃO COM TESTE NO INÍCIO

## 5 Procedimentos de ensino-aprendizagem 🏐

Esta aula aborda os conceitos de estruturas de repetição, utilizando a repetição com teste no início. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

Provavelmente você utilizou um despertador ou serviço de lembrete. Se desejarmos criar um programa que funciona de forma similar, basta criarmos um programa que emita um alerta, em determinado momento, sempre que programado. Porém, repetir comandos para realizar o alerta tantas vezes quantos forem os momentos que desejamos ser lembrados não parece interessante... Sem contar que o ideal é que o momento do alerta deveria ser configurado pelo usuário. Será que já conseguimos fazer isso?

#### - Metodologia:

Expor a sintaxe do bloco de repetição com teste no início (WHILE) frisando que ele controla um bloco de comandos e fica repetindo toda sua execução enquanto a condição teste se manter verdadeira (true). Não esquecer de mencionar como controlar para que o bloco para de repetir em determinado momento.

O professor deve mediar um processo de criação de um programa que execute quantas vezes o usuário desejar. Sugere-se que aproveite o programa de calculadora finalizado na aula passada e acrescente um loop de forma ao mesmo ficar em execução até que o usuário desista de utilizar o mesmo. Espera-se que após estas dinâmicas o aluno perceba que pode criar situações em que o programa pode ficar em execução contínua sem parar.

\* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação

em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita no momento de representá-los com uma linguagem de programação.

- Atividade verificadora de aprendizagem:

Voltar a situação problema e mediar os alunos na criação de um programa que funcione como um software de alerta de eventos do teclado, o programa deve ficar em loop contínuo até que o usuário digite 'X', e deve emitir um alerta sempre que uma vogal for digitada. O alerta pode ser sonoro (beep) ou visual.

## 6 Recursos didáticos



- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.

## 7 Leitura específica 🎢

[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 84, 85 e de 92 a 98. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12

# 

Artigo: C Plus Plus: Declarações e controle de fluxo. Acessível em:

http://www.cplusplus.com/doc/tutorial/control/ (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

Artigo: C Plus Plus: Beep. Acessível em: http://www.cplusplus.com/forum/beginner/36345/ (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

- Wikibooks: Programar em C++/Estruturas de repetição/While. Acessível em: https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Estruturas\_de\_repeti%C3%A7%C3%A3o# While

Atividade Autônoma Aura:

#### Ouestão 1:

Considere o trecho de código a seguir:

```
num = 0;
while ( num != 12 ){
cout << num << " ";
num = num + 1;
num = num + 1;
}
```

Pode-se dizer que este trecho:

- A) Realiza uma contagem regressiva
- B) Imprime os números de 0 a 12
- C) Imprime os números de 0 a 11
- D) Imprime os números de 0 a 10

```
E) Imprime os números pares de 0 a 10

Questão 2:
Considere o trecho de código a seguir:

int num;
cout << "Informe um numero [0-9]: ";
cin >> num;
while ( ( num < 0 ) || (num > 9 ) ) {
cout << "Informe um numero [0-9]: ";
cin >> num;
}

Pode-se dizer que este trecho:
A) Le um número inteiro qualquer
B) Le um número qualquer 10 vezes
C) Le 10 números diferentes
D) Le um número inteiro desde que entre 0 e 9 (inclusive)
E) Fica lendo números até que o usuário digite 0 ou 9
```



1 Código e nome da disciplina 🕕

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

## Semana 12: Tema - 5. ESTRUTURAS DE REPETIÇÃO

3 Objetivos

- Escrever estruturas de repetição com teste no final, baseando-se nas regras da linguagem de programação C/C++, para a elaboração de soluções algorítmicas compreensíveis ao computador;
- Validar códigos-fonte, utilizando teste de mesa, para verificação e checagem da corretude da solução;
- 4 Tópicos (i

## 5.3 REPETIÇÃO COM TESTE NO FINAL

## 5 Procedimentos de ensino-aprendizagem 🏐

Esta aula aborda os conceitos de estruturas de repetição, utilizando a repetição com teste no final. Como sugestão, segue o roteiro abaixo:

#### - Situação problema:

Já estamos estudando conceitos fundamentais de programação faz algumas aulas, e recentemente começamos conseguir utilizar o potencial computacional para realizar um conjunto de tarefas muito rapidamente. Na aula passada fizemos uma melhoria na nossa implementação de calculadora digital. Ela já realiza 5 operações (+, -, \*, /, Raiz) e fica executando em loop. Será que podemos melhorar?

#### - Metodologia:

Expor a sintaxe do bloco de repetição com teste no final (DO..WHILE) frisando que ele controla um bloco de comandos e fica repetindo toda sua execução enquanto a condição teste se manter verdadeira (true). Não esquecer de mencionar que por definição o laço sempre será executado no mínimo uma vez dado que o programa entra no laço sem fazer teste nenhum, performando o mesmo só no final.

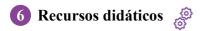
O professor deve mediar um brainstorm com a turma para determinar situações em que a garantia de execução de no mínimo uma passada do bloco seja necessária. Espera-se que após estas dinâmicas o aluno perceba que pode melhorar a calculadora substituindo o WHILE pelo DO..WHILE. Sugere-se que o professor modere o ajuste do software anterior junto da turma.

\* Importante: Sempre que forem apresentados comandos, apresentar em paralelo a sua representação em fluxograma e/ou em pseudocódigo (de preferência, lado a lado no quadro). Isso vai criando, no aluno, o conceito de que existem diferentes formas de representar um mesmo comando, o que facilita

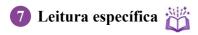
no momento de representá-los com uma linguagem de programação.

- Atividade verificadora de aprendizagem:

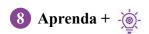
Aproveitar o conhecimento acumulado para resolver os exercícios do livro "Algoritmos" na página 105 e exercítar testes de mesa com os alunos.



- Laboratório de Informática com Internet com navegador Web instalado;
- Computador do professor com Datashow;
- Software Dev C++ (ou similar) instalado para todos.



[1] ANDRADE, Mairum C. Algoritmos. Rio de Janeiro: SESES, 2014. Páginas 98 a 103. Disponível em: http://repositorio.savaestacio.com.br/site/index.html#/objeto/detalhes/377757CE-7C5A-438D-A742-482DDDC54D12



Artigo: C Plus Plus: Declarações e controle de fluxo. Acessível em:

http://www.cplusplus.com/doc/tutorial/control/ (Acesse com o Chrome, clique com o botão direto e selecione traduzir para o português).

- Wikibooks: Programar em C++/Estruturas de repetição/Do-While. Acessível em: <a href="https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Estruturas\_de\_repeti%C3%A7%C3%A3o#Do-While">https://pt.wikibooks.org/wiki/Programar\_em\_C%2B%2B/Estruturas\_de\_repeti%C3%A7%C3%A3o#Do-While</a>

Atividade Autônoma Aura:

#### Ouestão 1:

Considere o trecho de código a seguir:

```
char op;
do {
// bloco de comandos
cin >> op;
} while ( ( op == 's') || ( op == 'S') );
```

Pode-se dizer que este trecho:

- A) Mostra um bloco de seleção optando entre os valores 's' e 'S'
- B) Mostra um bloco de repetição que executa enquanto op é igual a 's' ou 'S'
- C) Mostra um bloco de repetição imprimindo de 's' até 'S'
- D) Mostra um bloco de seleção optando entre os valores 's' ou 'S'
- E) Mostra um bloco de repetição que executa enquanto op é igual a 's' e 'S'

#### Ouestão 2:

Considerando as estruturas de repetição com teste no início e as com teste no fim, analise as afirmativas a seguir:

- I A principal diferença entre as duas estruturas está no fato de a estrutura enquanto realizar o teste condicional para determinar o término da repetição no início da instrução; por outro lado, a estrutura faca...enquanto realiza o teste de parada no final.
- II O faça...enquanto primeiro executa o bloco de código, depois realiza o teste lógico de controle do laço, o que causa obrigatoriamente a execução do bloco uma vez, e com o bloco controlado pelo enquanto o bloco pode não executar.
- A) A afirmativa I está certa e a afirmativa II está certa
- B) A afirmativa I está certa e a afirmativa II está errada
- C) A afirmativa I está errada e a afirmativa II está certa
- D) A afirmativa I está errada e a afirmativa II está errada
- E) Não há relação entre as duas afirmativas

# a

#### Plano de Aula

1 Código e nome da disciplina 🕕

ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 13: Tema - 6. VETORES E MATRIZES (ATIVIDADE PRÁTICA SUPERVISIONADA)

- 3 Objetivos
- Empregar o vetor para armazenamento de dados em um programa
- 4 Tópicos (i

6.1 APLICAÇÃO DE VETOR

5 Procedimentos de ensino-aprendizagem 🌇

Nesta aula, estaremos conectados com o conteúdo digital. O aluno explora e estuda, previamente, o conteúdo digital disponível em seu ambiente virtual.

6 Recursos didáticos

A aula será realizada no ambiente virtual de aprendizagem.

7 Leitura específica 📸

O aluno deverá consultar a bibliografia proposta no tema.

8 Aprenda + -

# q

#### Plano de Aula

1 Código e nome da disciplina 🕕

ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

Semana 14: Tema - 6. VETORES E MATRIZES (ATIVIDADE PRÁTICA SUPERVISIONADA)

- 3 Objetivos
- Empregar o vetor para armazenamento de dados em um programa
- 4 Tópicos (i

6.1 APLICAÇÃO DE VETOR

5 Procedimentos de ensino-aprendizagem 🌇

Nesta aula, estaremos conectados com o conteúdo digital. O aluno explora e estuda, previamente, o conteúdo digital disponível em seu ambiente virtual.

6 Recursos didáticos

A aula será realizada no ambiente virtual de aprendizagem.

7 Leitura específica 📸

O aluno deverá consultar a bibliografia proposta no tema.

8 Aprenda + ---

# a

#### Plano de Aula

■ Código e nome da disciplina

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

## Semana 15: Tema - 6. VETORES E MATRIZES (ATIVIDADE PRÁTICA SUPERVISIONADA)

- 3 Objetivos
- Empregar a matriz para armazenamento de dados em um programa.
- 4 Tópicos (i

## 6.1 APLICAÇÃO DE MATRIZ

5 Procedimentos de ensino-aprendizagem 🌇

Nesta aula, estaremos conectados com o conteúdo digital. O aluno explora e estuda, previamente, o conteúdo digital disponível em seu ambiente virtual.

6 Recursos didáticos

A aula será realizada no ambiente virtual de aprendizagem.

7 Leitura específica 📸

O aluno deverá consultar a bibliografia proposta no tema.

8 Aprenda + ---

# a

#### Plano de Aula

■ Código e nome da disciplina

## ARA0017 INTRODUÇÃO À PROGRAMAÇÃO ESTRUTURADA EM C

2 Semana/Tema 🛗

## Semana 16: Tema - 6. VETORES E MATRIZES (ATIVIDADE PRÁTICA SUPERVISIONADA)

- 3 Objetivos
  - Empregar a matriz para armazenamento de dados em um programa.
- 4 Tópicos j

## 6.2 APLICAÇÃO DE MATRIZ

5 Procedimentos de ensino-aprendizagem 🌇

Nesta aula, estaremos conectados com o conteúdo digital. O aluno explora e estuda, previamente, o conteúdo digital disponível em seu ambiente virtual.

6 Recursos didáticos

A aula será realizada no ambiente virtual de aprendizagem.

7 Leitura específica 🃸

O aluno deverá consultar a bibliografia proposta no tema.

8 Aprenda + -