



UNIVERSIDADE ESTADUAL DE SANTA CRUZ - UESC

PRÓ-REITORIA DE GRADUAÇÃO – PROGRAD

DEPARTAMENTO DE CIÊNCIAS EXATAS - DCET

COLEGIADO DE CIÊNCIA DA COMPUTAÇÃO - COLCIC

Lista de Exercícios

Curso	Ciência da Computação
Disciplina	Linguagem de Programação III
Professor(es)	Otacílio José Pereira
Assunto	<ul style="list-style-type: none">- Lista de exercícios para iniciar com Java a partir dos tópicos aprendidos em Linguagem C (Programação Estruturada)- Vale lembrar que o Java será usada para o trato de orientação a objetos, a ideia é no primeiro momento iniciar com esta parte de programação estruturada

Comentários iniciais

- Após revisarmos C e sua abordagem de programação estruturada, o foco na disciplina será iniciar com a Linguagem Java, fazer esta transição.

- A proposta é, para aproveitarmos nosso conhecimento e revisão de C, é interessante fazer uma comparação ou equivalência entre as linguagens C e Java.

- Para esta comparação será usada a estrutura de tópicos que vimos.

- Em cada parte, após os exercícios haverá um quadro que permite fazer a compração.

Parte 1 : Conceito do Ambiente de Desenvolvimento, Projeto e Arquivos

Exercícios:

- 1) Escolha um ambiente de sua preferência, instale e dê os primeiros passos com o ambiente.
 - Perceba as partes do ambiente e crie um projeto ou mesmo um arquivo.
 - Observe as várias partes do ambiente:
Toolbar/Menus, Navegação pelo projeto e codificação.
- 2) Crie um programa simples, o “Hello World” e continue sua ambientação pelo IDE.
 - Identifique como executar um programa
 - Observe os resultados da execução do programa

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
Ambiente	Dev C++, Codeblocks e outros	Netbeans, Eclipse, VSCode e outros
Estrutura de Projeto	Em geral, trabalha-se apenas com um arquivo em C - O ponto de partida é a função main do arquivo C	- Existe um projeto - Dividido em vários pacotes (package) - Há um arquivo com uma classe principal - Que tem um método estático main - Que é o ponto de partida do programa
Arquivos	- Em geral se programa com um único arquivo pois tratam exercícios simples das disciplinas de Algoritmos e de Linguagem de Programação	- Vários arquivos - Um arquivo é configurado como principal no projeto da aplicação - Arquivos apresentam extensão .java

Parte 2 : Estrutura Geral de um programa

Exercícios para esta parte:

- 1) Observe as partes do seu programa "Hello World" e identifique:
 - Onde são importadas as bibliotecas? (No caso do Java chamado de pacotes)
 - Existe um bloco principal "main"
 - Como é feita a impressão de texto em tela?
- 2) Faça uma pesquisa e identifique:
 - O que são pacotes?
 - Qual o papel do System.in e do System.out? Qual relação com o "stdin" e "stdout" no C?
 - O que é um projeto com seus diversos arquivos ou classes?
 - Qual o papel da classe principal de um projeto e como configurá-la em um projeto?

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
Importação de bibliotecas	Através de #include <code>#include <stdio.h></code> <code>#include <stdlib.h></code>	Através de import <code>import java.util.*;</code>
Bloco principal	<code>#include <stdio.h></code> <code>#include <stdlib.h></code> main() { printf("Hello world!!"); system("pause"); }	<code>package mobilidadeurbana;</code> <code>import java.util.*;</code> <code>public class Agricultura40 {</code> <code>public static void main(String[] args) {</code> <code>System.out.print(" Hello World ! ");</code> <code>}</code> <code>}</code>
Início do programa	Primeira instrução da função principal main()	Deve ser configurada a classe principal no projeto e esta classe deve conter um método estático main.

Parte 3 : Entrada e Saída

Exercícios:

- 1) O que você entende por uma aplicação em “Modo Console”? Qual diferença de aplicações com Interface Gráfica?
- 2) Faça um programa que leia um número inteiro e calcule o quadrado deste número.

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
Saída de dados	<pre>int i = 10; printf("Valor digitado : %d ", i);</pre>	<p>Análogo ao C, com uso de <code>System.out</code></p> <pre>System.out.print(<expressões>); System.out.printf(<expressões>);</pre>
Uso de coringas	<pre>%d : Inteiro %c : Caractere %f : Float %s : String</pre>	<p>São impressos sem coringas com <code>System.out.print(i);</code></p> <p>Ou como no C através de <code>System.out.printf("Valor digitado : %d ", i);</code></p> <p>Vale lembrar que não é vocação do Java ser usado em modo Console. Existe esta forma para impressões de erros e exceções.</p>
Entrada de dados	<pre>int i; scanf("%d", &i);</pre>	<p>Existem várias formas de entrar com dados. A que apresenta conceito similar ao que é usado em C, através de “scan” é apresentada abaixo.</p> <pre>import java.util.Scanner; Scanner s = new Scanner(System.in); opcao = (char) s.nextLine().charAt(0); Numero = s.nextInt();</pre>
Uso de coringas	Vide printf	Vide printf

Parte 4 : Variáveis e tipos de dados

Exercícios para Parte 4 – Variáveis e Parte 5 – Operações aritméticas

- 1) Uma serralheria precisa de realizar orçamento de peças circulares que ela realiza. Faça um programa que leia o nome de um cliente, o raio da peça e a quantidade e apresente um orçamento considerando que o metro quadrado (m2) do material é R\$25,00.
- 2) Faça um programa que leia um valor a sacar e indique qual a quantidade mínima de notas a emitir pelo caixa eletrônico considerando que ele está abastecido com notas de 10, 20 e 50 reais.
- 3) Existem muitas diferenças entre a escrita destes programas em C e em Java? Faça uma pesquisa sobre a origem do Java e entenda qual o motivo disso.

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
Declaração de variáveis	<tipo> <nome da variável>; int i;	- Similar ao C - Classes são consideradas tipos
Tipos principais	int; float; char; char[];	- Análogo ao C - Vetor de char no Java é uma String
Escopo das variáveis	Delimitado pelos { } Em geral pelos { } da função principal (main)	- Análogo ao C

Parte 5 : Operações aritméticas

Tópico	Em C	Em Java
Operações aritméticas	+ - * /	- Análogo ao C
Divisão inteira	/ : aplicada a inteiros % : Resto da divisão (equivalente a mod em Pascal)	- Análogo ao C

Parte 6 : Comandos condicionais

Exercícios:

- 1) No problema do orçamento, tratado nas partes anteriores, acrescente as seguintes regras de desconto no orçamento.
Quando área acima ou igual a 100 m² - Desconto de 20%
Quando área entre 40 e 100 m² - Desconto de 10%
Áreas inferiores a 40 m² - Sem desconto
- 2) Faça um programa que leia o dia e mês de uma data no ano de 2023 e indique quantos dias corridos e qual dia da semana (segunda, terça, ...)

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
if	<pre>if <condição> <bloco para condição verdadeira> else <bloco para condição falsa> if ((n % 2) == 0) { printf("Numero par"); } else { printf("Numero ímpar"); }</pre>	- Análogo ao C
switch	<pre>switch (variável) { case 0 : { bloco } break; case 1 : { bloco } break; case 2 : { bloco } break; default { bloco } break; }</pre> <p>- Variável deve ser inteiro ou char</p>	- Análogo ao C

Parte 7 : Comandos de repetição

Exercícios:

- 1) Faça um programa que leia 10 itens de uma compra e calcule qual o total da compra, qual o valor do item mais caro e a média dos itens.
- 2) Altere o programa anterior para que em vez de ler 10 itens o programa solicite itens conforme o usuário indique que ainda há itens a processar.

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
for	<pre>for (<inicio>; <condição fim>; <passo>) { <instruções para repetição> } for (i=0; i<10; i++) { printf(" Valor = %d \n", i); }</pre>	- Análogo ao C
while	<pre>while (<condição>) { <instruções para repetição> } i = 0; while (i<10) { printf(" Valor = %d \n", i); i++; }</pre>	- Análogo ao C

Parte 8 : Vetores e matrizes

Exercícios para aplicar o quadro:

- 1) Faça um programa que leia 6 números e depois imprima na ordem inversa com que foi lido
- 2) Observação: o uso de vetores com tipos simples (inteiro, double, float e outros) é similar entre C e Java. Quando se utiliza vetores de estruturas em C, há diferenças em relação ao Java que serão explicadas ao se explorar Orientação a Objetos.

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
Declaração	<code>int vetor[10];</code>	<code>int vetor[] = new int[10];</code> - Um vetor deve ser declarado e instanciado (alocado em memória) <code>int vetor[];</code> <code>vetor = new int[10];</code>
Acesso	<code>vetor[2] = 20;</code>	Idem C
Detalhes	Inicia na posição 0 . <code>int vetor[10];</code> Vetor tem as posições de 0 a 9	Idem C

Parte 9 : Estruturas

Exercícios para aplicar o quadro:

- 1) Faça um programa que leia os dados de um sensor de temperatura que poderia ser usado por exemplo em uma casa inteligente. Um sensor possui ID, descrição e valor. Após ler, o programa testa se a temperatura está acima de 24° e imprime os dados do sensor.

Em relação a esta parte, você pode usar este quadro para comparar o C e Java.

Tópico	Em C	Em Java
Declaração	<pre>Fora do bloco main struct <Nome> { <campos>; } struct Onibus{ int numero, lotacao; char modelo[20], placa[10]; };</pre>	<pre>São usadas as classes class <nome da classe> { <campos>; } public class Onibus { public int Numero; public int Lotacao; public String Placa; public String Modelo; }</pre> <p>- O modificador de acesso public indica que campos podem ser acessados de qualquer parte do programa</p>
Declaração	<pre>// Apenas um ônibus em uma variável // simples struct Onibus onibus1; // Um vetor de ônibus para cadastro struct Onibus cadastro[100];</pre>	<pre>Onibus onibus1 = new Onibus(); // Em geral usam-se as coleções para objetos de uma classe ArrayList<Onibus> l = new ArrayList<Onibus>(); // Veremos no decorrer da disciplin.</pre>
Acesso	<pre>Os campos são acessados via campo. onibus1.lotacao; onibus[10].lotacao</pre>	<pre>onibus1.lotacao;</pre>
Detalhes	<pre>struct Onibus onibus1; // A variável onibus1 já está com seu // espaço de memória</pre>	<pre>Os objetos precisam ser instanciados (explicitamente alocar espaço), enquanto as estruturas em C já são automaticamente instanciadas. Onibus onibus1 = new Onibus();</pre>

Parte 10 : Funções

Exercícios para aplicar esta parte:

- 1) Faça um programa que apresente um menu de opções para um sistema de controle de irrigação em um cenário de Agricultura 4.0. O menu deve conter as seguintes opções:
 - (1) Cadastrar Sensor
 - (2) Cadastrar Equipamento
 - (3) Monitorar umidade
 - (9) Sair

Perceba que o programa não precisa processar as opções, apenas mostrar as opções para o usuário e daí imprimir qual opção foi escolhida.

- 2) Faça um programa que leia o raio da base e a altura de um cilindro e calcula qual o seu volume.

Tópico	Em C	Em Java
	Existem isoladas, e podem ser invocadas no corpo principal.	A grande diferença é que as funções em Java são métodos de uma classe, não existem isoladas. Por isso, precisam ser invocadas a partir de um Objeto.
Procedimento	- Sem retorno, uso do void para retorno nulo void <nome função> (<parâmetros>) { } void inicial() { system("cls"); printf(" ----- \n"); printf(" Mobilidade \n"); printf("Versao:0.1 \n"); printf("Data: 24/08/2014 \n"); printf(" ----- \n\n"); }	Além do detalhe acima, a declaração de métodos obedece a mesma lógica das funções.
Retorno	<tipo retorno> <nome função> (<parâmetros>) { <Bloco de instruções>; return <variável ou expressão>; }	Além do detalhe acima, a declaração de métodos obedece a mesma lógica das funções.

	<pre> int menu() { char opcao; printf(" -- Menu -- \n"); printf(" (I) Novo Onibus\n"); printf(" (A) Alterar Onibus\n"); printf(" (S) Sair \n"); printf("\n Digite a opcao: "); fflush(stdin); scanf("%c", &opcao); return opcao; } </pre>	
Parâmetros	<pre> int soma (int a, int b) { int total; total = a+b; return total; } </pre>	<p>Além do detalhe acima, a declaração de métodos obedece a mesma lógica das funções.</p>