



Programação Orientada a Objetos I

Prof. Carlos Sicsú, DSc.

Apresentação da Disciplina

CTC – CENTRO DE TECNOLOGIA E CIÊNCIAS
FCEE – FACULDADE DE CIÊNCIAS EXATAS E ENGENHARIAS
DEPCOMP – DEPARTAMENTO DE COMPUTAÇÃO
CURSOS DE COMPUTAÇÃO – TADS / CC



Prof. Dr. Eng^o. **Carlos** Augusto **Sicsú** Ayres do Nascimento

- Técnico em Edificações (CEFET);
 - Bacharel em Matemática Aplicada à Informática;
 - Bacharel em Engenharia Elétrica (Ênfase em Computação) ;
 - Especialista em Engenharia da Computação (UERJ);
 - Mestre em Engenharia da Computação (UERJ);
 - Doutor em Computação de Alto Desempenho (Eng. Civil/COPPE-UFRJ).
-
- E-mail: carlos.sicsu.uerj@gmail.com



PLANO DE ENSINO



Ementa:

- Introdução a Orientação a Objetos: Classes, Atributos e Métodos. Herança, polimorfismo, encapsulamento, agregação. Vetores e Matrizes. Tratamento de Exceções.



Objetivos da Disciplina:

- Identificar os paradigmas de linguagens de programação;
- Desenvolver programas computacionais utilizando o paradigma da Programação Orientada a Objetos;
- Distinguir programação imperativa de programação orientada a objetos;
- Modelar problemas computacionais empregando as técnicas de programação orientada a objetos.



Conteúdo Programático :

1 INTRODUÇÃO ÀS LINGUAGENS DE PROGRAMAÇÃO

- 1.1 Razões para estudar Linguagens de Programação (LPs);
- 1.2 O papel das LPs no Desenvolvimento de Software;
- 1.3 Propriedades desejáveis em uma LPs;
- 1.4 Especificação de LPs;
- 1.5 Métodos de Implementação de LPs;
- 1.6 Paradigmas de LPs;
 - 1.6.1 Paradigma Imperativo (Estruturado, OO e Concorrente);
 - 1.6.2 Paradigma Declarativo (Funcional e Lógico);
- 1.7 Origem e Evolução das LPs.



2 INTRODUÇÃO À LINGUAGEM JAVA

2.1 Comparação entre Programação Imperativa e Orientada a Objetos;

2.2 Vantagens da Linguagem Java;

2.3 Ambiente de Programação Java: J2SDK.



3 Programação OO Em Java - Objetos, Classes, Herança e Polimorfismo

3.1 Conceitos de Orientação a Objetos;

3.2 Classes, Objetos, Métodos e Atributos;

3.2.1 Conceitos;

3.2.2 Tipos de Dados, Variáveis e Constantes;

3.2.3 Métodos: construtor, acesso, modificador;

3.2.4 Atribuições, Inicializações e Operadores;

3.2.5 Manipulação de Strings;

3.2.6 Controle de Fluxo: Estruturas de Seleção e Repetição;

3.2.7 Arrays: dados e objetos;

3.2.8 Classes pré-definidas;



4 Conceitos de Orientação a Objetos

4.1 Herança: Superclasses e Subclasses;

4.2 Polimorfismo e Sobrecarga de Métodos;

4.3 Classes Abstratas;

4.4 Interfaces e Classes Internas;

4.5 Pacotes;

4.6 Tratamento de Exceções.



METODOLOGIA:

Aulas expositivas, práticas e dialogadas, podendo contar com o apoio de projeções, além do desenvolvimento de trabalhos, individuais e/ou em grupo, visando ao preparo dos alunos para uma sociedade comprometida com o desenvolvimento científico e tecnológico. Para isso, as atividades propostas favorecem a autonomia do aluno e a construção do conhecimento.



ATIVIDADES DISCENTES:

Resolver listas de exercícios e de problemas práticos em laboratório de informática (ou remotamente), os quais permitem, através de demonstrações e experiências, a fixação dos conteúdos através de exercícios e/ou trabalhos práticos (individuais ou em grupo).



PROCEDIMENTOS DE AVALIAÇÃO:

Os alunos serão avaliados de acordo com normas regimentais, comunicados pela Pró-reitoria de Graduação, orientações normativas da Coordenação do Curso e critérios específicos do Professor da Disciplina, utilizando-se, no todo ou em partes, dos seguintes instrumentos:

- (i) Provas objetivas individuais;
 - (ii) Provas dissertativas individuais;
 - (iii) Provas Institucionais individuais;
 - (iv) Provas integradas individuais;
 - (v) Trabalhos em grupos, em sala e extra sala de aula.
- Provas AV1 e AV2 - prova escrita e/ou trabalhos práticos.
 - A prova AV3 - prova escrita valendo até 10 (dez) pontos.



PROCEDIMENTOS DE FREQUÊNCIA:

- Haverá chamada e/ou assinatura de ata de presença em todas as aulas.
- O aluno deverá comparecer a no mínimo 75% das aulas para aprovação.



BIBLIOGRAFIA BÁSICA:

- DEITEL, Harvey M.; DEITEL, Paul J. **Java: como programar**. São Paulo: Prentice-Hall, 2005.
- HORSTMANN, Cay S.; CORNELL, Gary. **Core Java 2**. São Paulo: Makron, 2001-2005. 2 v.
- CAMARÃO, Carlos; FIGUEIREDO, Lucília. **Programação de Computadores em Java**. Rio de Janeiro: LTC 2003.
- HORSTMANN, Cay S. **Big Java**. Porto Alegre: Bookman, 2004.

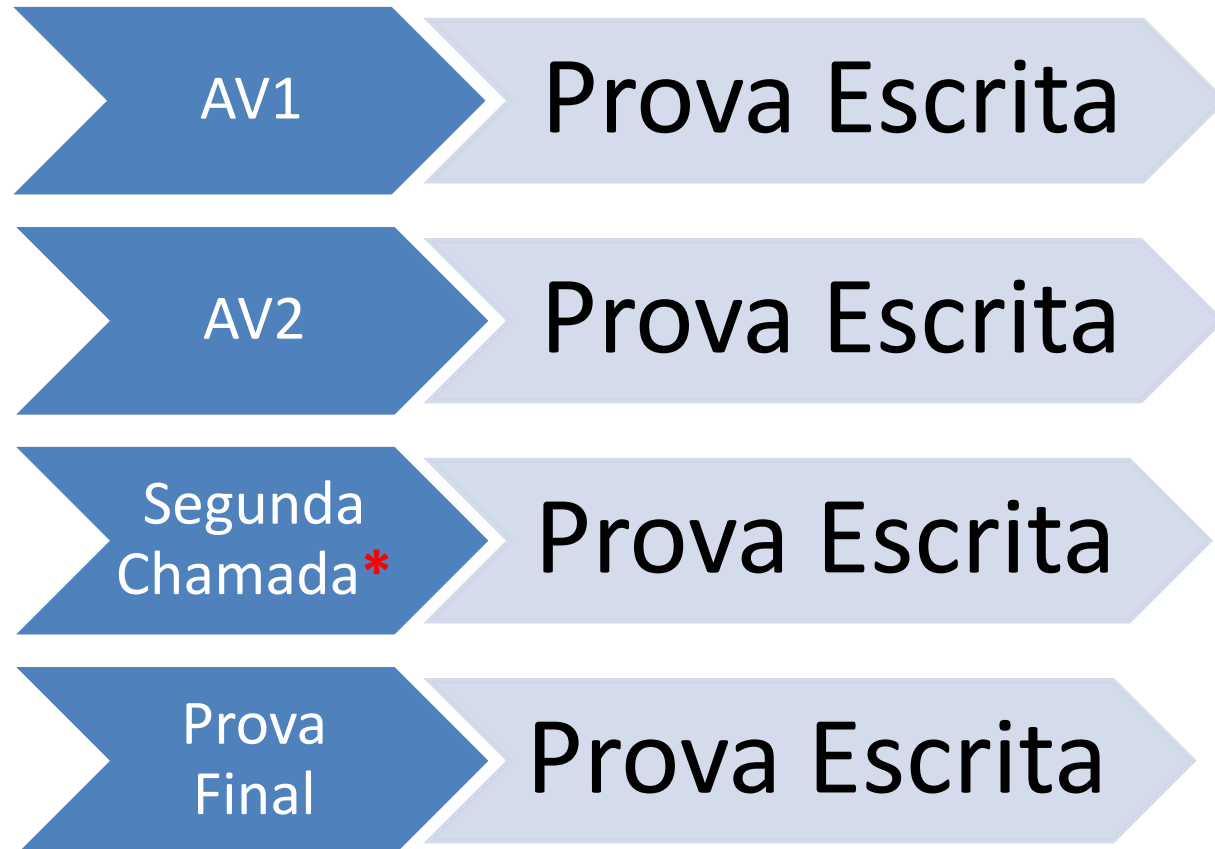


BIBLIOGRAFIA COMPLEMENTAR:

- JAMSA, Kris A.; KLANDER, Lars. Programando em C/C++: a bíblia. São Paulo - Makron, 1999.



Avaliação



*** O aluno só tem direito a 2ª Chamada após o Deferimento do Requerimento aberto dentro do prazo. Consultar o Manual do Aluno para saber em quais casos o aluno pode requerer 2ª Chamada.**



Datas Importantes*

AV1

- 19/maio

AV2

- 07/julho

2ª Chamada

- 21/julho

Prova Final

- 28/julho

* Sujeito a alteração (se ocorrer, será comunicado durante as aulas).



Critério de Aprovação

Média $(AV1 + AV2)/2$

Semestral A média semestral precisa ser **$\geq 4,0$** para poder realizar a Prova Final

APROVADO DIRETO Média Semestral $\geq 7,0$ e
Frequência $\geq 75\%$

Média Final $(\text{Média Semestral} + \text{Prova Final}) / 2$

APROVADO Média Final $\geq 5,0$ e
Frequência $\geq 75\%$

REPROVADO
(sem Prova Final)

Média Semestral $< 4,0$ OU Frequência $< 75\%$



Dúvidas

1- Se o aluno faltar AV1 ou AV2.

Deverá **requerer a 2ª Chamada no prazo** – anexando o comprovante (conforme Manual do aluno)

2- Posso fazer Prova Final tendo a Média Semestral $\geq 7,0$?

Não. Somente os alunos com Média Semestral $\geq 4,0$ e Média Semestral $< 7,0$.





- PCM (Plano de Conteúdo Mínimo) 2023-1:
 - Semana 1 (24/03/2023): Apresentação do professor e da disciplina, critérios de aprovação e de avaliação. Histórico da linguagem Java.
 - Semana 2 (31/03/2023): Exercícios Práticos de Revisão de Algoritmos.
 - **Semana 3 (07/04/2023): Feriado (Paixão de Cristo).**
 - Semana 4 (14/04/2023): Características da Linguagem Java. Estruturas de Controle de Fluxo em Java.
 - **Semana 5 (21/04/2023): Feriado (Tiradentes).**
 - **Semana 6 (28/04/2023): Classes e Objetos; Atributos e Métodos; Métodos *Setters* e *Getters*; Exemplos práticos. Exercícios práticos.**
 - Semana 7 (05/05/2023): Métodos Construtores; Sobrecarga de métodos construtores. Exemplos práticos. Exercícios práticos.



• Plano de Conteúdo Mínimo:

- Semana 8 (12/05/2023): Herança: Superclasses e Subclasses; Simples e Múltipla. Exemplos práticos. Exercícios práticos. Dúvidas e orientações para a prova AV1.
- **Semana 9 (19/05/2023): Avaliação (AV1).**
- Semana 10 (26/05/2023): Agregação e Particionamento. Exemplos Práticos e Exercícios práticos.
- Semana 11 (02/06/2023): Modificadores e Encapsulamento. Exemplos Práticos e Exercícios práticos.
- Semana 12 (09/06/2023): Classes Abstratas e Interfaces. Exemplos Práticos e Exercícios práticos.
- Semana 13 (16/06/2023): Vetores, Matrizes e *ArrayList*. Exemplos Práticos e Exercícios práticos.



• Plano de Conteúdo Mínimo:

- Semana 14 (23/06/2023): Tratamento de Exceções. Exemplos Práticos e Exercícios práticos.
- Semana 15 (30/06/2023): Dúvidas e orientações para a Prova AV2.
- **Semana 16 (07/07/2023): Avaliação (AV2).**
- Semana 17 (14/07/2023): Divulgação de graus. Dúvidas e orientações para a Prova Final.
- **Semana 18 (21/07/2023): Prova de 2ª chamada.**
- **Semana 19 (28/07/2023): Prova Final.**



Instalação da Ferramenta de Desenvolvimento.



1º - Download do Java (JDK):

<https://www.oracle.com/java/technologies/downloads/>

Java SE Development Kit 8u321 ou superior

Instalar e REINICIAR o computador.

Cadastro Oracle:

Opção: Development tools.

Observações:

Empresa = Estudante

Função = Estudante



2º - Download do NetBeans 15 ou superior:

<https://netbeans.apache.org/download/nb15/>

Clicar em Download e escolher uma das opções:

1. Binaries (sem instalador): [netbeans-15-bin.zip](#) (SHA-512, PGP ASC)

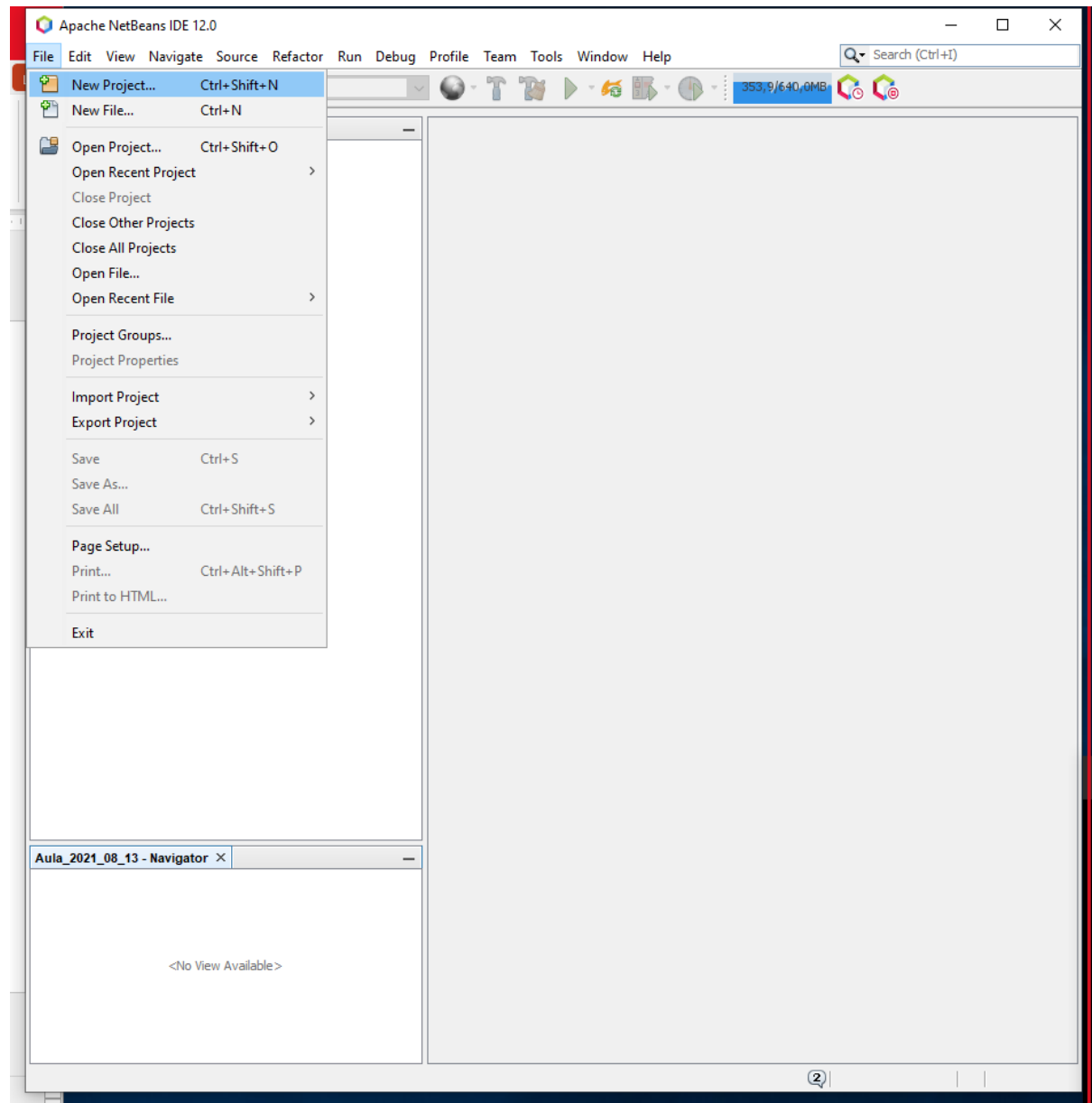
Baixar a versão ZIP, descompactar, mover para uma pasta do drive para criar o atalho. Para o atalho, abrir a pasta do NetBeans, acessar a pasta /bin e clicar sobre o ícone do executável com o botão da direita do mouse e selecionar enviar para Área de Trabalho (criar atalho).

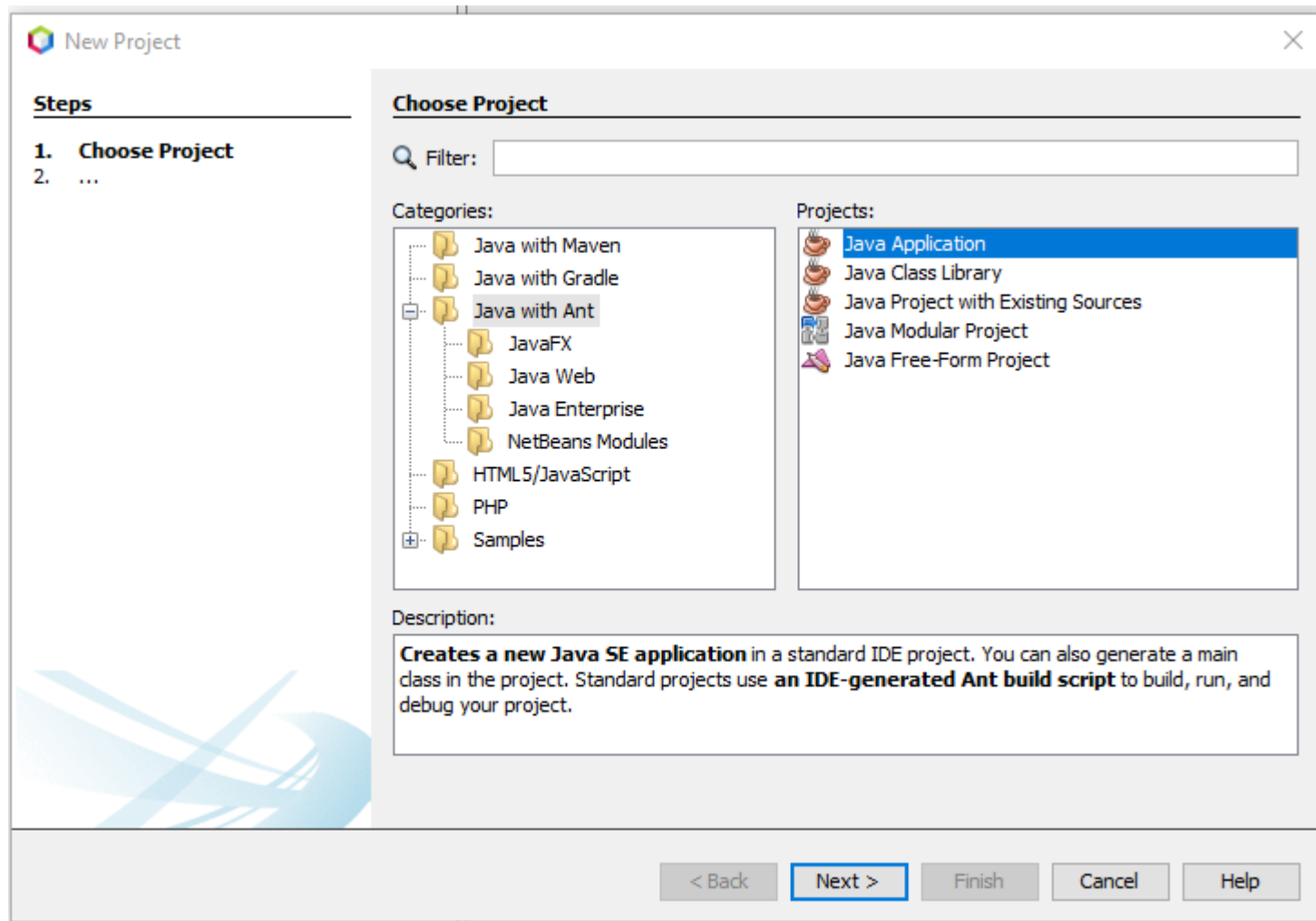
OU

2. Instalador:

- [Apache-NetBeans-15-bin-windows-x64.exe](#) (SHA-512, PGP ASC)
- [Apache-NetBeans-15-bin-linux-x64.sh](#) (SHA-512, PGP ASC)
- [Apache-NetBeans-15-bin-macosx.dmg](#) (SHA-512, PGP ASC)

Seguir os passos da instalação.







- Se for apresentada uma tela de download e instalação, você deve realizar o download e ativar a ferramenta.
- Continue com NEXT



New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☒ Create Main Class

< Back Next > **Finish** Cancel Help



New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name:

Project Location:

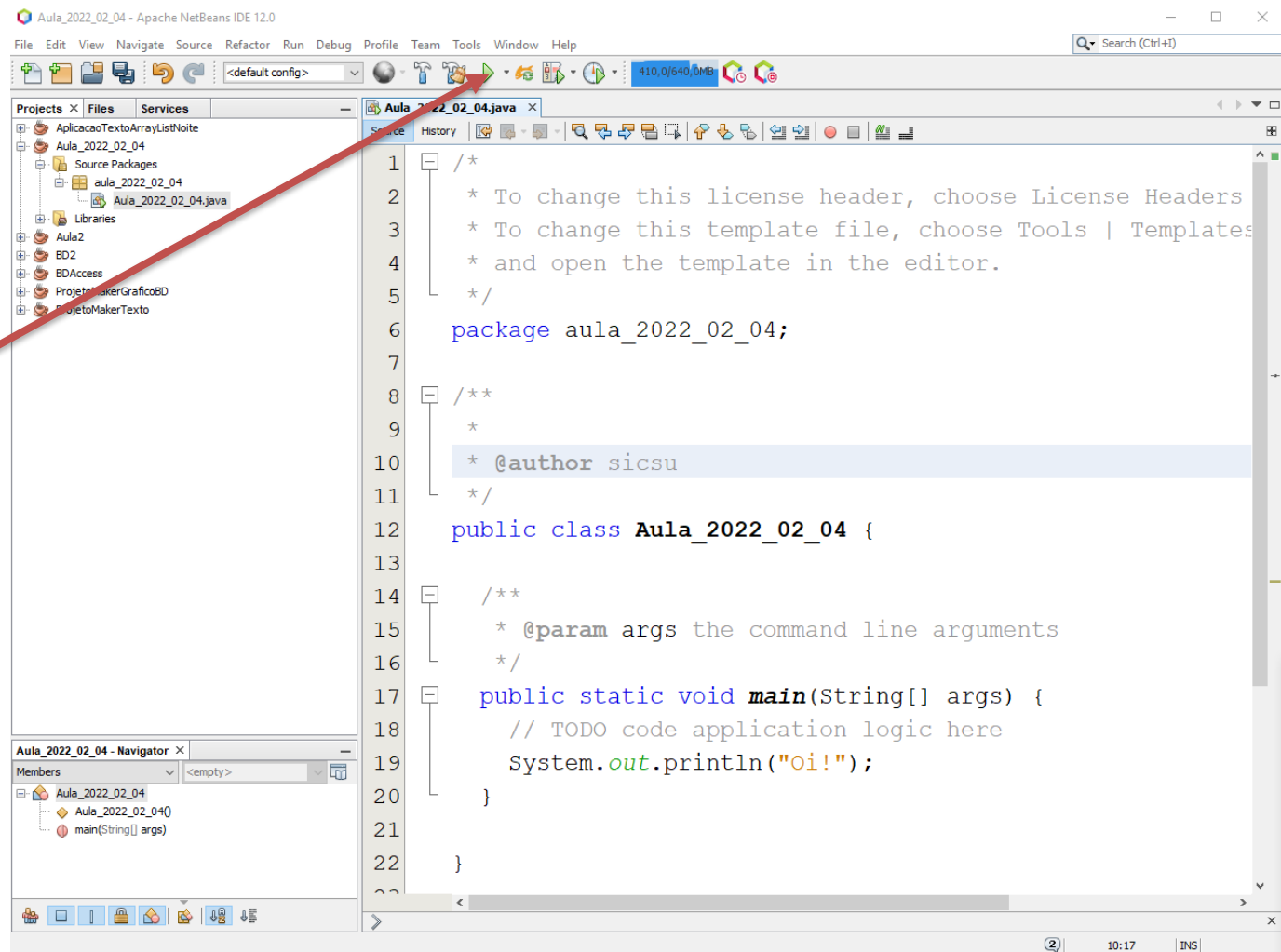
Project Folder:

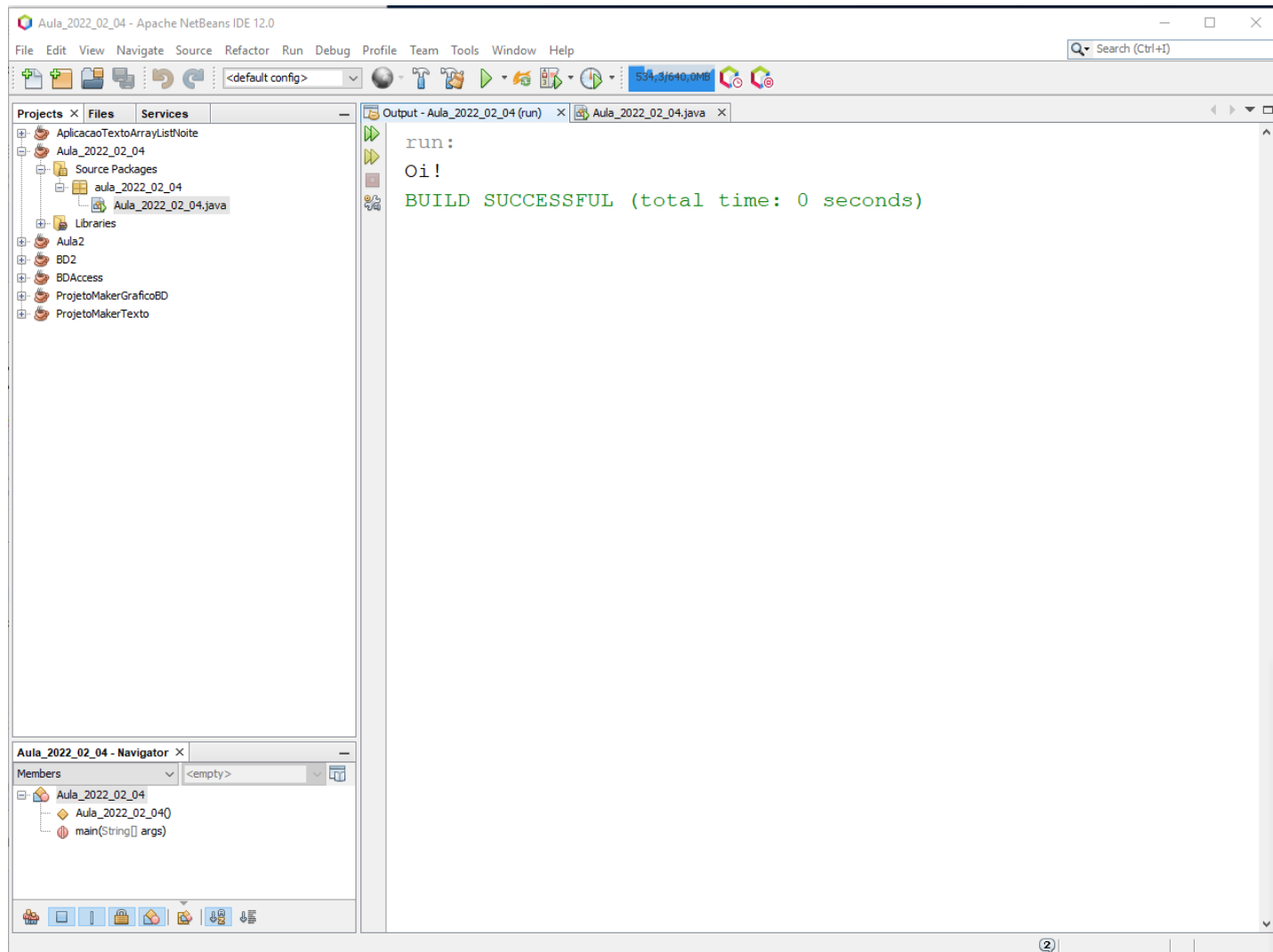
☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

☒ Create Main Class







Obrigado pela atenção.