Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра ЕОМ



**Лабораторна робота № 2**

з дисципліни: «Адгоритми та методи обчислень»,

на тему: «Асимптотичні характеристики складності алгоритму; алгоритми з поліноміальною та експоненціальною складністью»  
Варіант № 25

Виконав: ст. гр. КІ-28

Степанов В.В.

Посилання на ГітХаб проекту: https://github.com/YourAloneFriend/AMC/tree/main

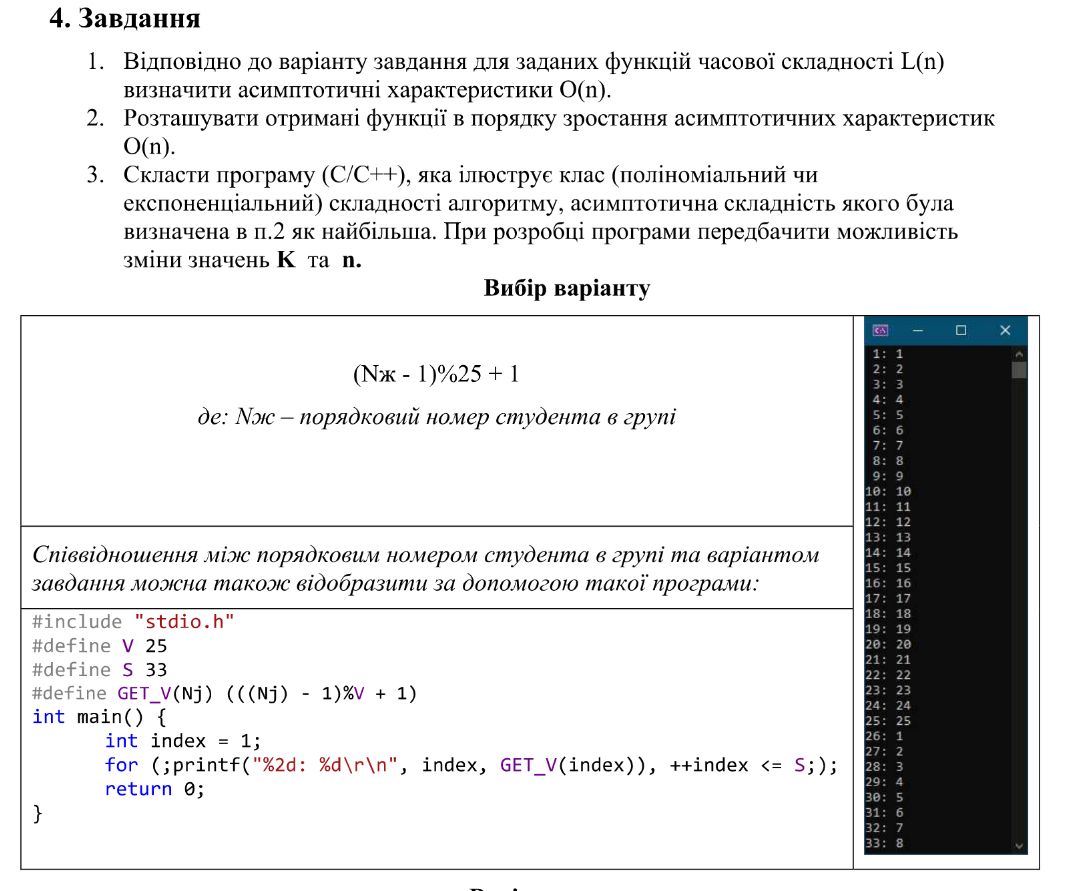
Прийняв:

Замроз П.І.

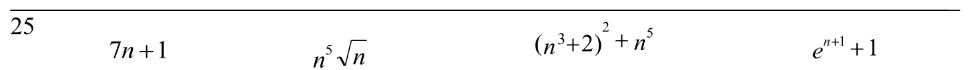
Львів – 2022

**Мета:** Ознайомитись з асимптотичними характеристиками складності та класами складності алгоритмів.

**ЗАВДАННЯ та дані згідно варіанту**



**РОЗВ’ЯЗУВАННЯ**

**Алгоритм програми:**

**РОЗВ’ЯЗУВАННЯ**

1) Задані функції часової складності L(n) для чотирьох алгоритмів:

1. L(n)1= 7n+1; 2. L(n)2= n5√n; 3. L(n)3= (n3+2)2+n5; 4. L(n)4= en+1+1;

Використавши правило сум і правило добутків знайдемо O(n):

O(n)1= n; O(n)2= n5√n; O(n)3= n6; O(n)4= en;

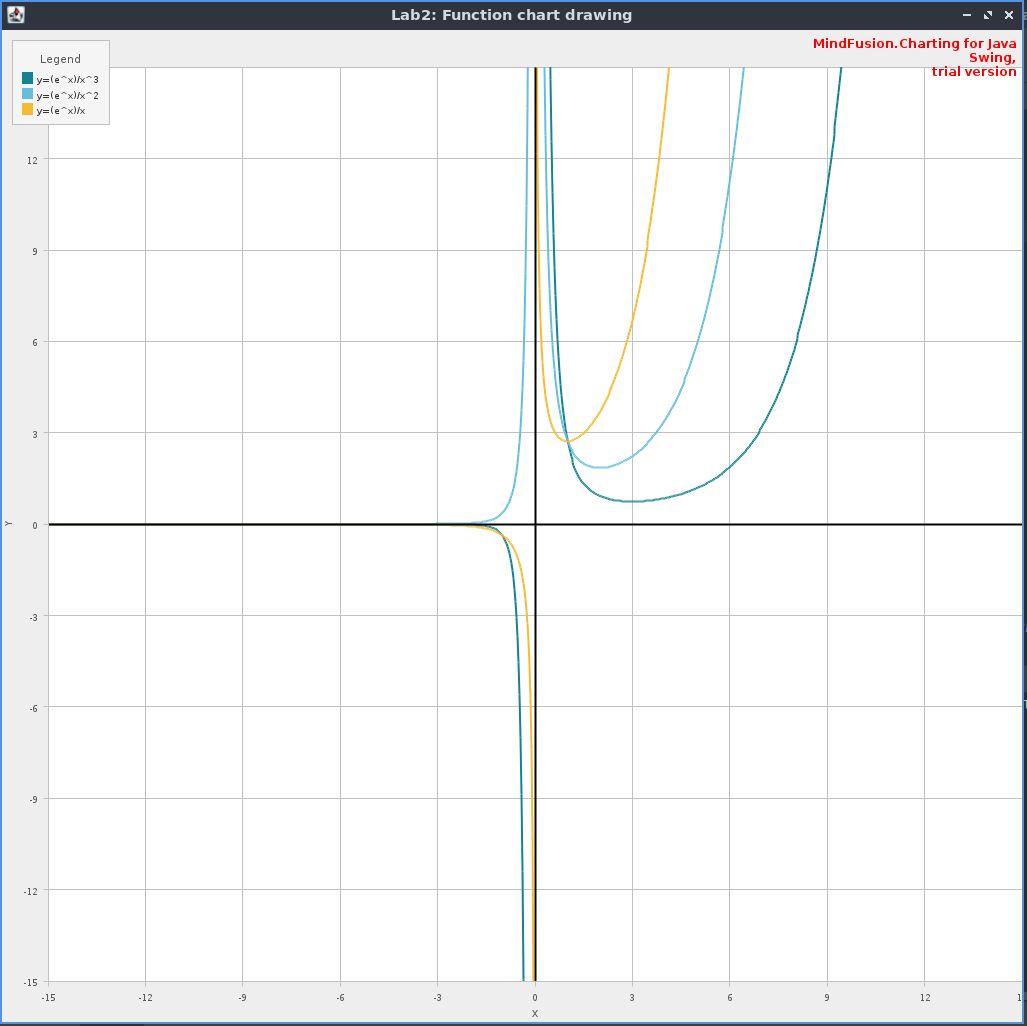
2) Розташування функцій O(n) у порядку зростання:

O(n)1= n; O(n)4= en; O(n)2= n5√n; O(n)3= n6;

**Код main.java:**

package amo\_lab2;  
  
import java.awt.BorderLayout;  
import java.awt.Color;  
import java.util.Arrays;  
  
import javax.swing.JFrame;  
import javax.swing.SwingUtilities;  
  
import com.mindfusion.charting.FunctionSeries;  
import com.mindfusion.charting.GridType;  
import com.mindfusion.charting.swing.LineChart;  
import com.mindfusion.drawing.SolidBrush;  
  
public class Main extends JFrame{  
  
 Main()  
 {  
 setDefaultCloseOperation(*EXIT\_ON\_CLOSE*);  
 setSize(1024, 1024);  
 setTitle("Lab2: Function chart drawing");  
 getContentPane().add(drawChart(), BorderLayout.*CENTER*);  
 }  
  
 public static void main(String[] args)  
 {  
 SwingUtilities.*invokeLater*(new Runnable()  
 {  
 public void run()  
 {  
 try  
 {  
 new Main().setVisible(true);  
 }  
 catch (Exception exp)  
 {  
 }  
 }  
 });  
 }  
  
 private LineChart drawChart()  
 {  
 LineChart lineChart= new LineChart();  
  
 lineChart.getXAxis().setMinValue(-15.0);  
 lineChart.getXAxis().setMaxValue(15.0);  
 lineChart.getXAxis().setOrigin(0.0);  
 lineChart.getXAxis().setTitle("X");  
  
  
 lineChart.getYAxis().setMinValue(-15.0);  
 lineChart.getYAxis().setMaxValue(15.0);  
 lineChart.getYAxis().setOrigin(0.0);  
 lineChart.getYAxis().setTitle("Y");  
  
 lineChart.setGridType(GridType.*Crossed*);  
  
 lineChart.getTheme().setCommonSeriesStrokes(  
 Arrays.*asList*(  
 new SolidBrush(new Color(19, 130, 147)),  
 new SolidBrush(new Color(102, 190, 223)),  
 new SolidBrush(new Color(246, 187, 50))  
 )  
 );  
  
  
 lineChart.getTheme().setCommonSeriesFills(  
 Arrays.*asList*(  
 new SolidBrush(new Color(19, 130, 147)),  
 new SolidBrush(new Color(102, 190, 223)),  
 new SolidBrush(new Color(246, 187, 50))  
 )  
 );  
  
  
 FunctionSeries f1, f2, f3;  
  
 try  
 {  
 f1 = new FunctionSeries("Exp(x)/Pow(x, 3)", 1000, -15, 15);  
 f1.setTitle("y=(e^x)/x^3");  
 lineChart.getSeries().add(f1);  
  
 f2 = new FunctionSeries("Exp(x)/Pow(x, 2)", 1000, -15, 15);  
 f2.setTitle("y=(e^x)/x^2");  
 lineChart.getSeries().add(f2);  
  
 f3 = new FunctionSeries("Exp(x)/x", 1000, -15, 15);  
 f3.setTitle("y=(e^x)/x");  
 lineChart.getSeries().add(f3);  
 }catch (Exception e)  
 {  
 e.printStackTrace();  
 }  
  
 return lineChart;  
 }  
  
}

**Скріншоти програми:**



**Висновок:** Я ознайомився з асимптотичними характеристиками складності та класами складності алгоритмів.