LAPORAN PRAKTIKUM

STRUKTUR DATA



DISUSUN OLEH :

FATHUR RAMADHAN FAHMI

2311533012

DOSEN PENGAMPU :

Dr. Wahyudi, S.T.M.T

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS ANDALAS

2024

1. TUJUAN
2. Mahasiswa mampu memahami cara menggunakan Stack
3. Mahasiswa mampu memahami cara kerja Stack pada data
4. Mahasiswa mampu membuat sebuah program menggunakan Stack
5. PEMBAHASAN

Stack di Java adalah struktur data, seperti antrian, larik, daftar tertaut, atau pohon . Yang membedakannya dari yang lain adalah bahwa Java Stack didasarkan pada prinsip Last In, First Out (LIFO). Artinya adalah ketika Anda menggunakan dua perintah untuk menambah dan menghapus elemen dari tumpukan, yang pertama Anda hapus akan selalu menjadi yang terakhir Anda tambahkan.

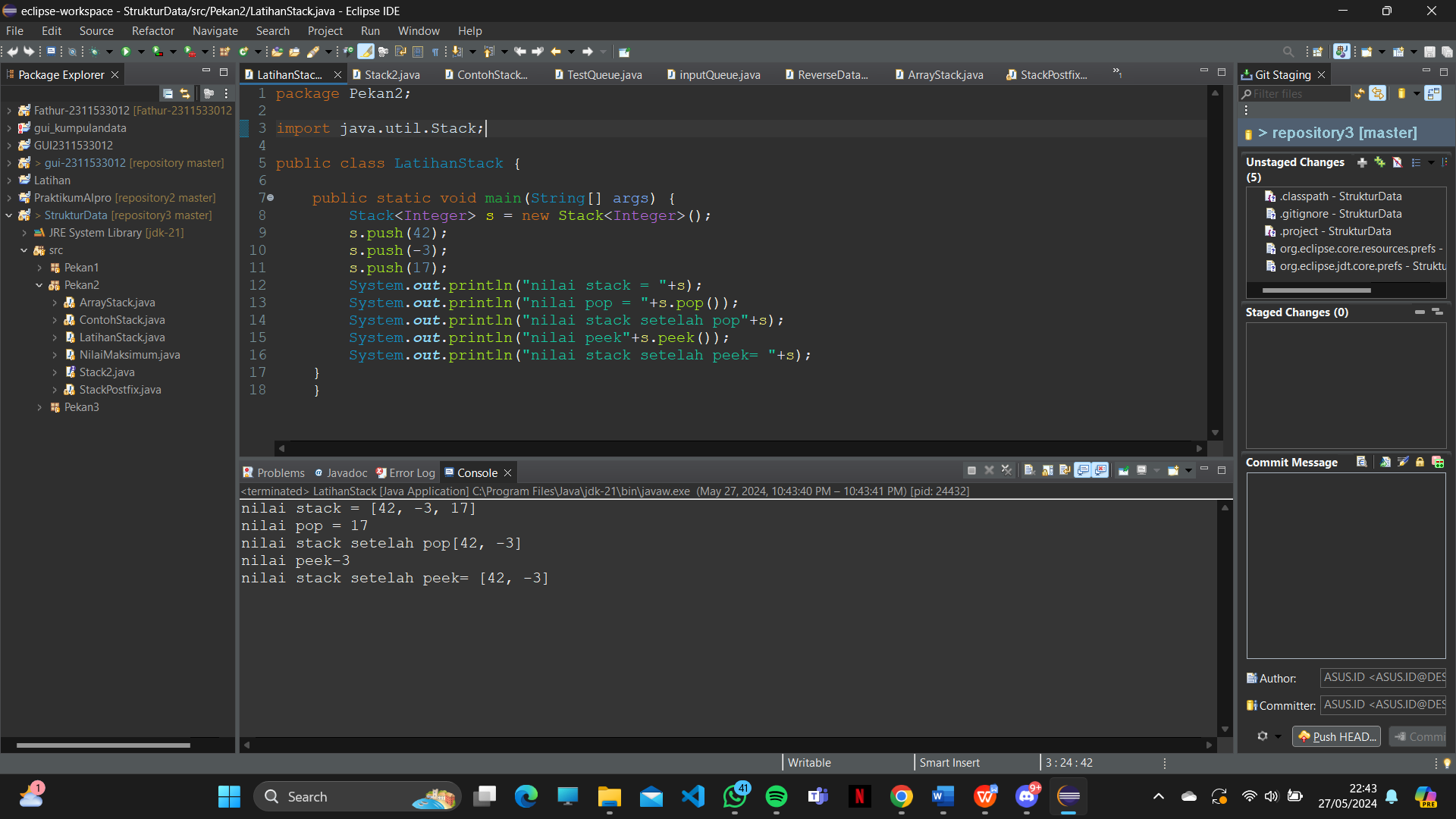
Ada beberapa operasi dasar yang dapat dilakukan pada stack:

* push(element): Menambahkan elemen baru ke atas stack.
* pop(): Menghapus elemen teratas dari stack dan mengembalikan nilainya.
* peek(): Mengambil nilai elemen teratas dari stack tanpa menghapusnya.
* isEmpty(): Mengembalikan nilai true jika stack kosong, dan false jika stack berisi elemen.
* size(): Mengembalikan jumlah elemen yang terdapat dalam stack.
* search(Object o): Mengembalikan posisi berbasis 1 dari elemen dalam stack. Jika elemen tidak ditemukan, mengembalikan -1.
* Inisialisasi Stack: **Stack<Integer> stack = new Stack<>();** membuat objek stack untuk menyimpan elemen-elemen bertipe Integer.
* push(): Menambahkan elemen 10, 20, dan 30 ke dalam stack.
* peek(): Mengambil elemen teratas (30) tanpa menghapusnya.
* pop(): Menghapus dan mengembalikan elemen teratas (30).
* isEmpty(): Memeriksa apakah stack kosong (tidak kosong dalam contohini).
* search(): Mencari posisi elemen 20 dalam stack.

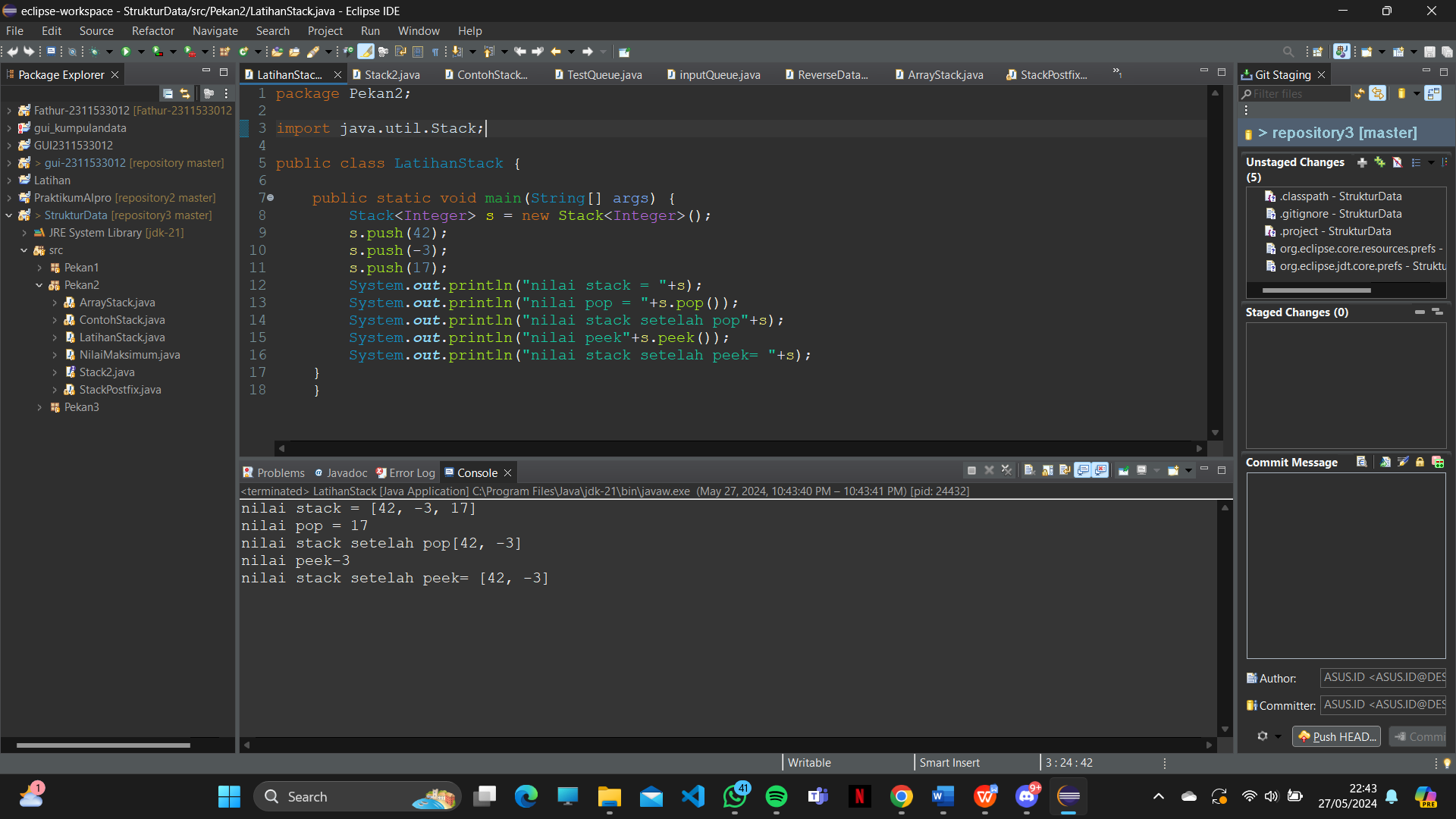
1. LANGKAH PRAKTIKUM
2. Buka Eclipse IDE
3. Lalu, buat package pekan2
4. Kemudian, buat class dan kodingan seperti dibawah ini :

* Class LatihanStack

Input :

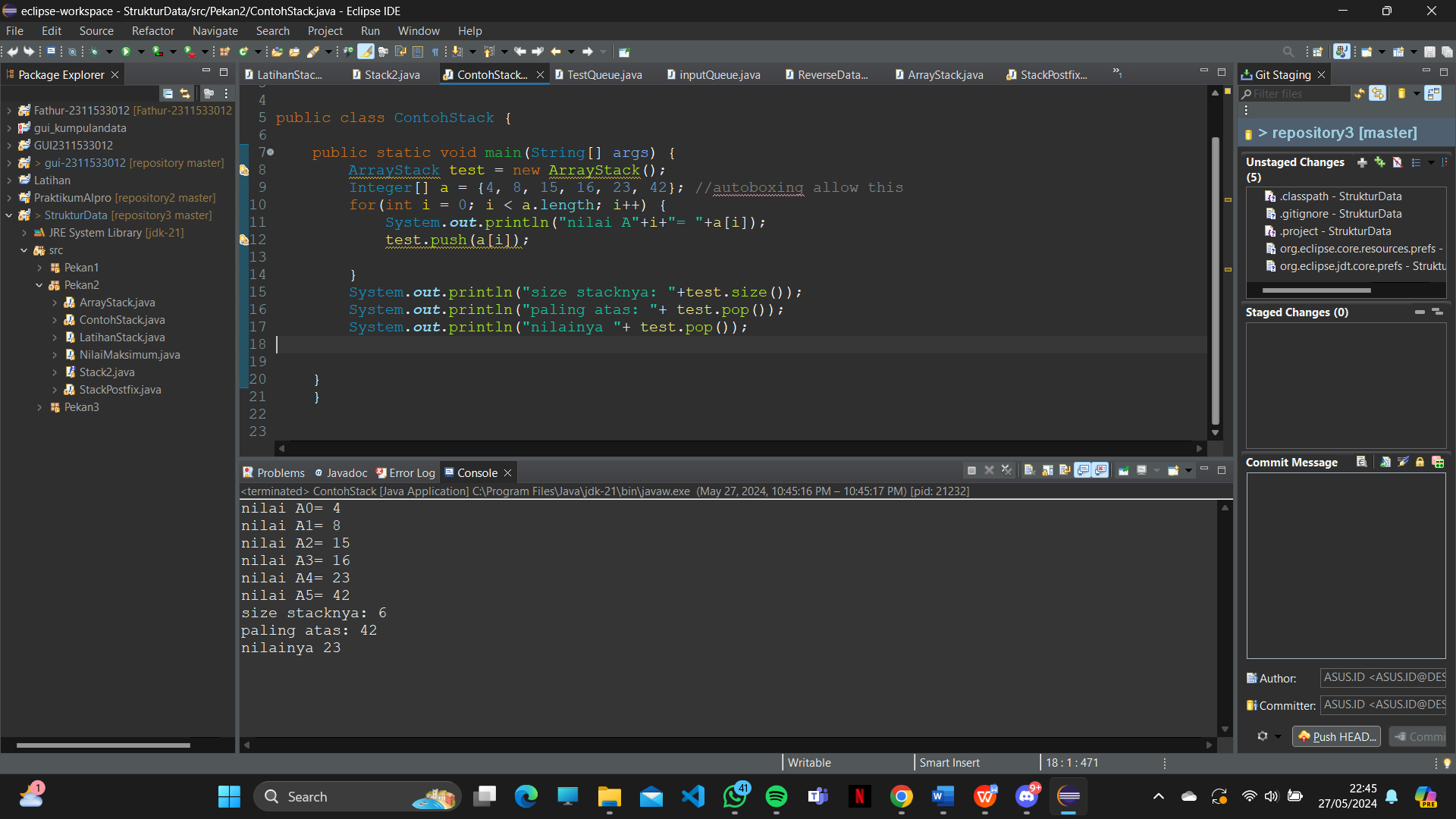


Output :

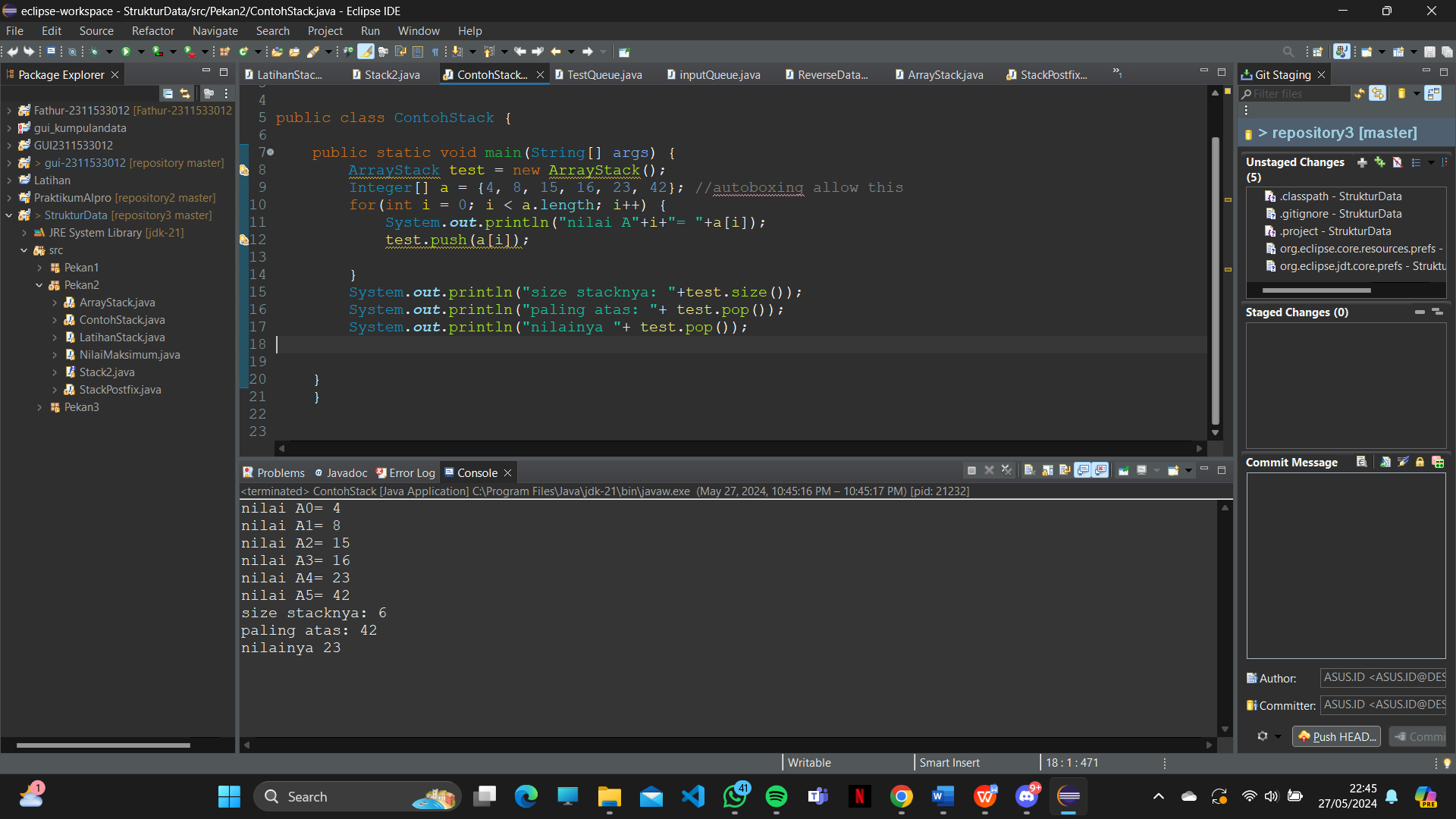


* Class ContohStack

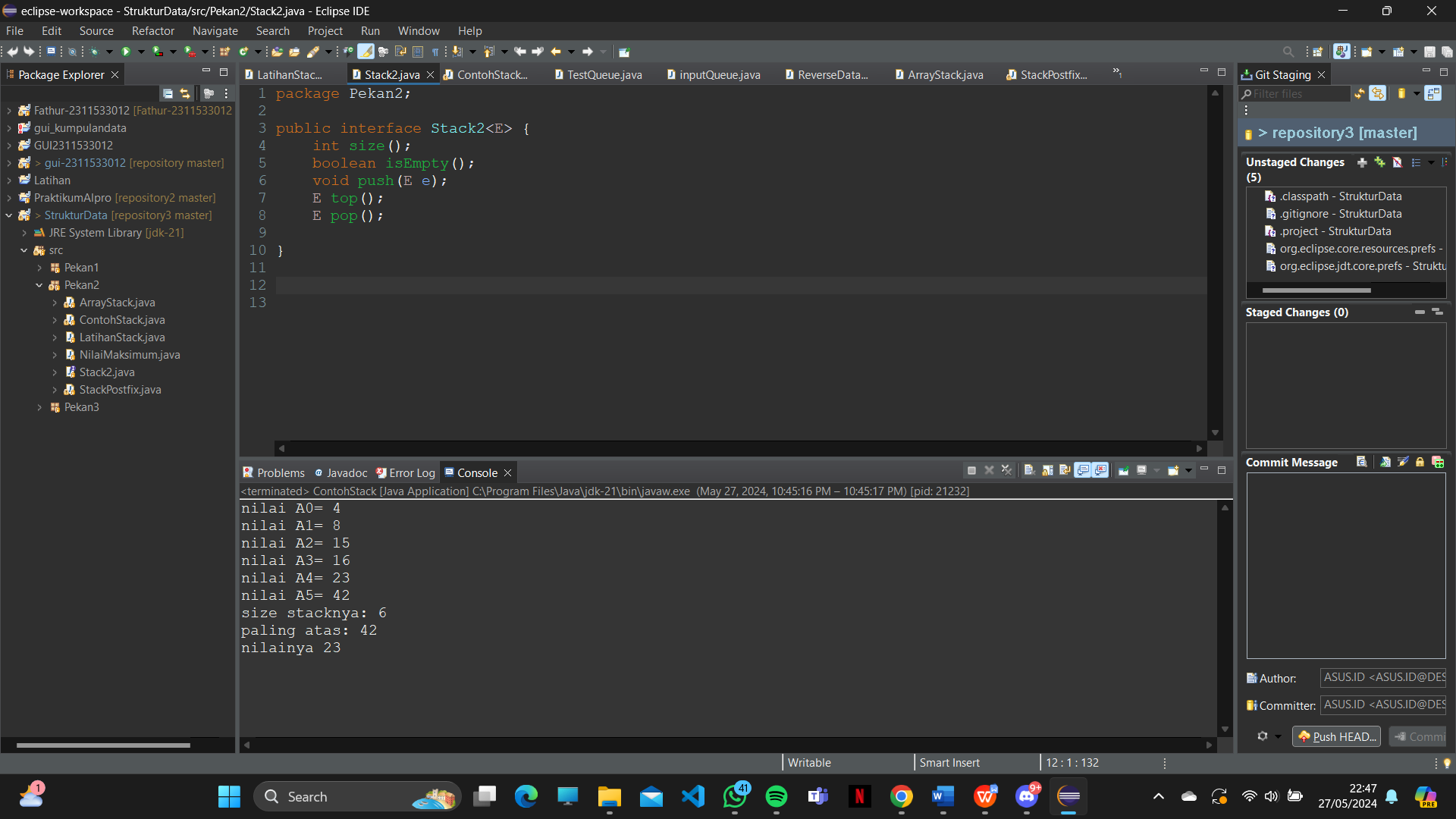
Input :



Output :

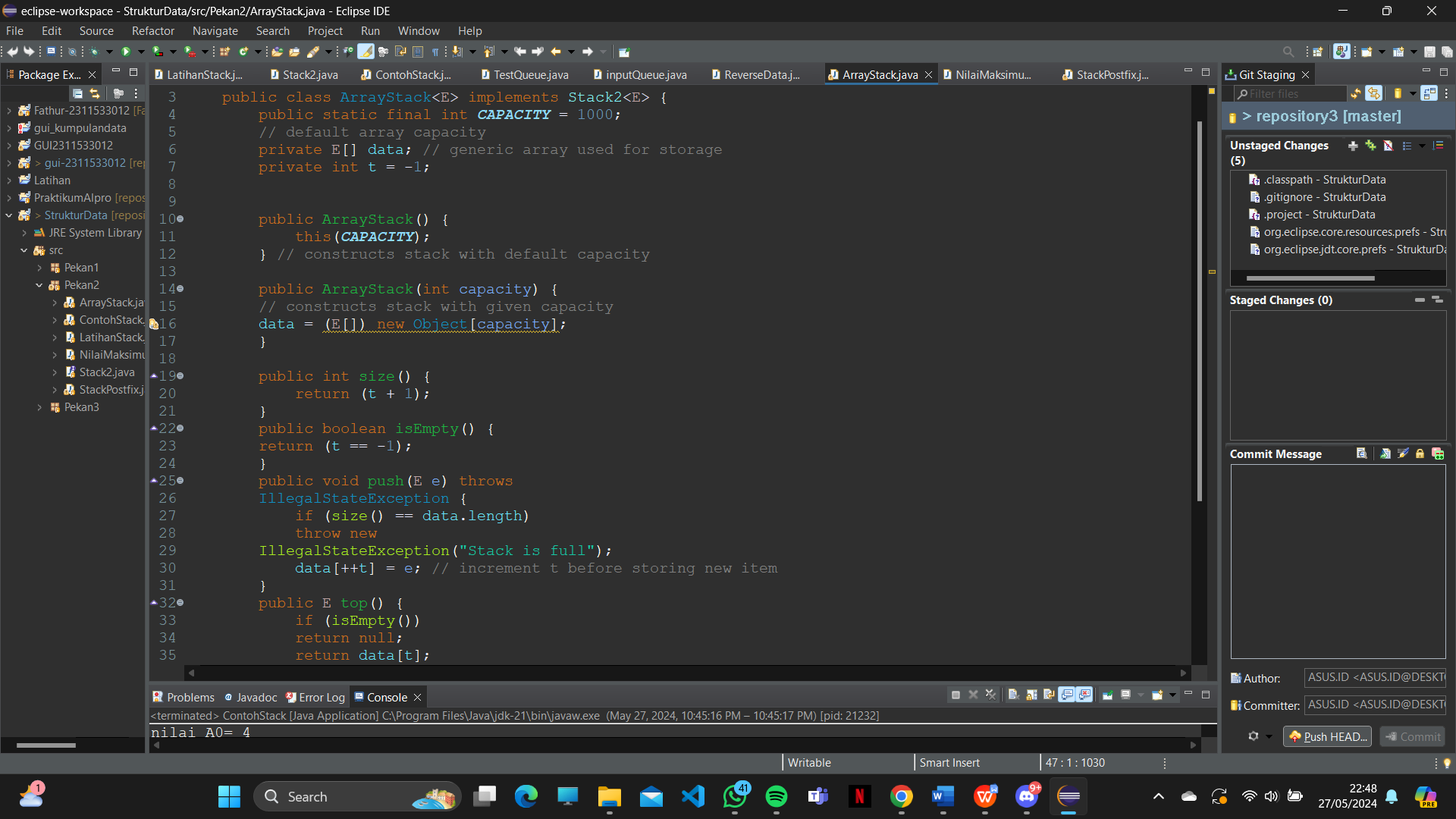


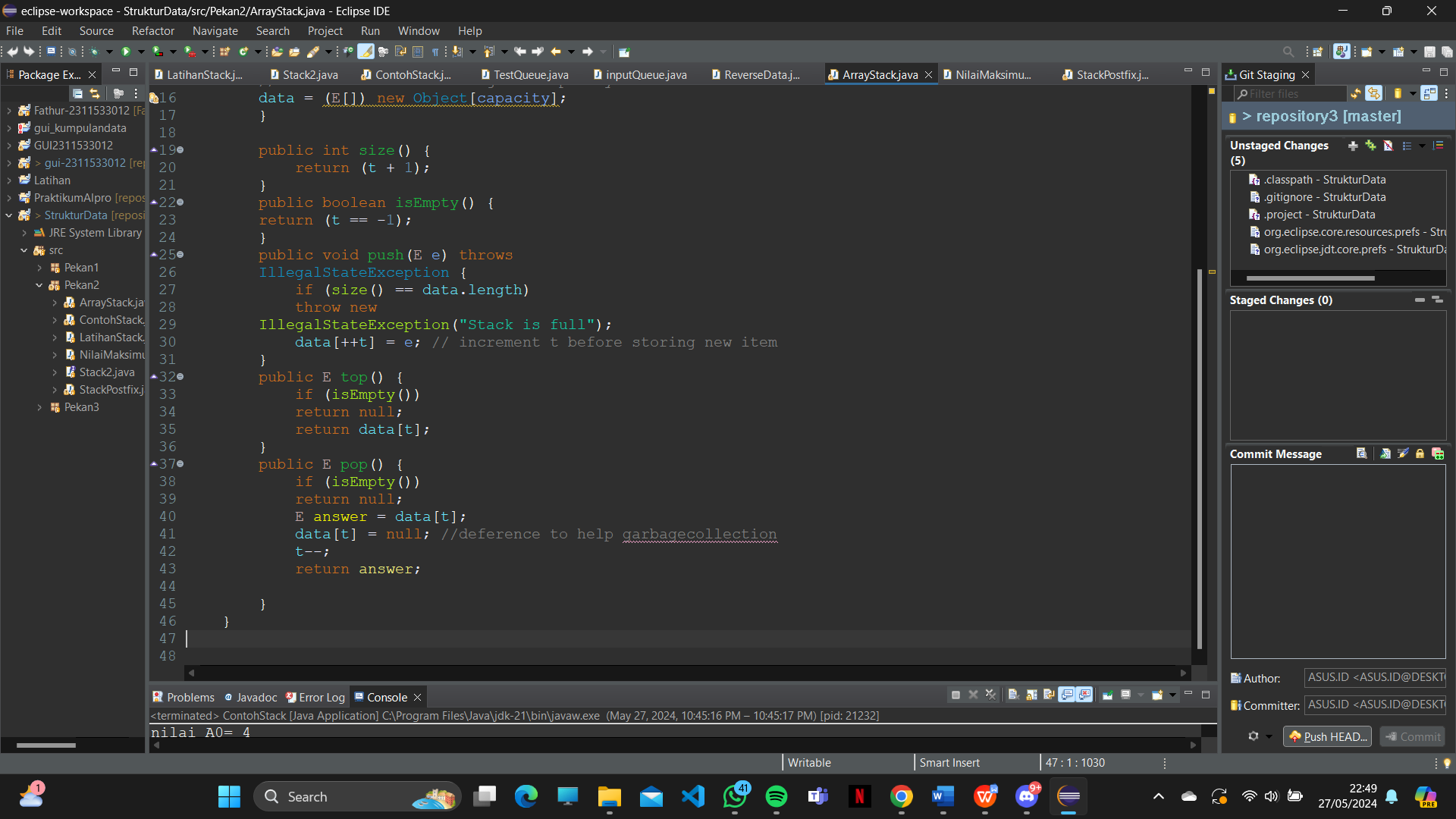
* Stack2



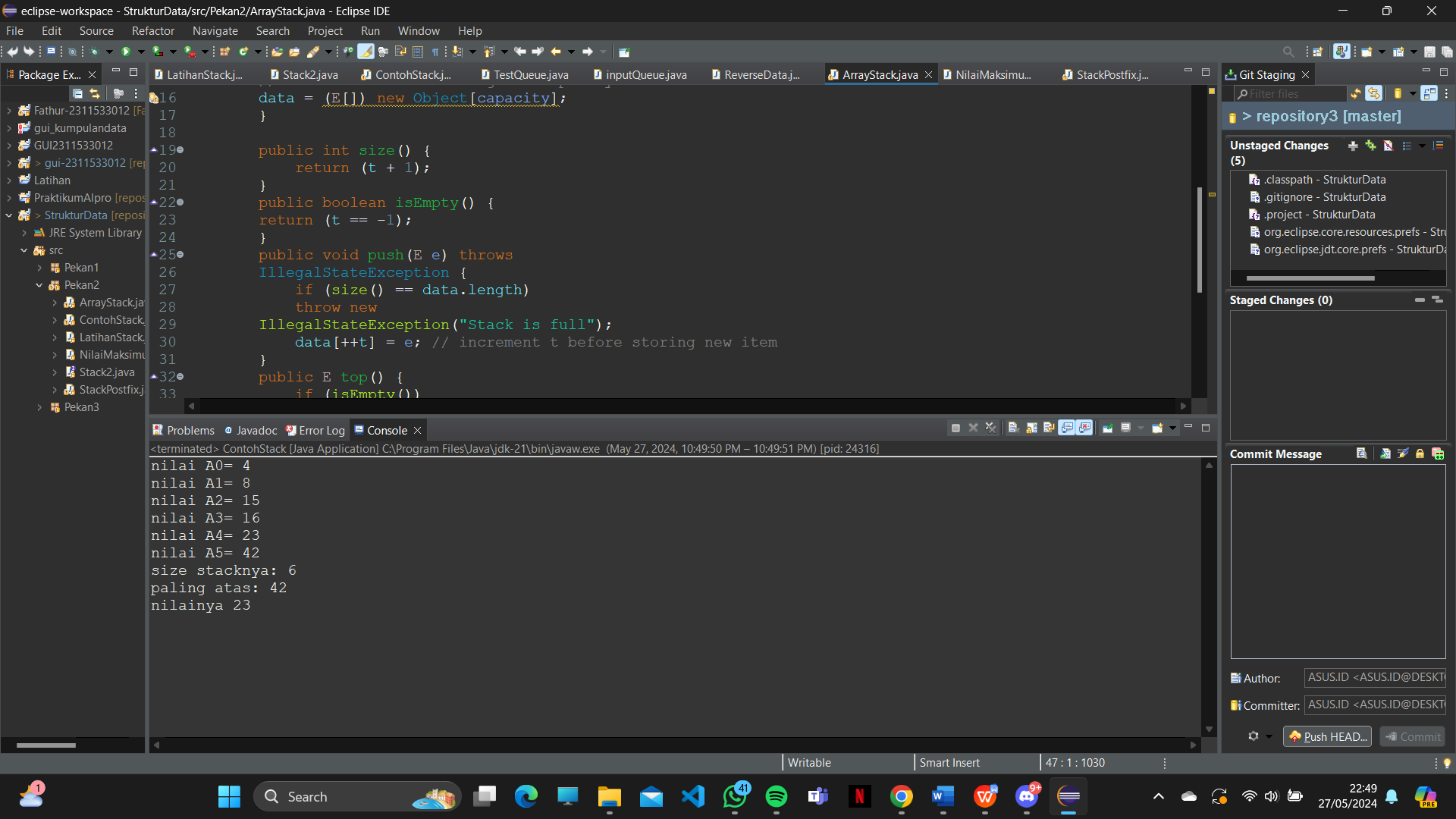
* Class ArrayStack

Input :



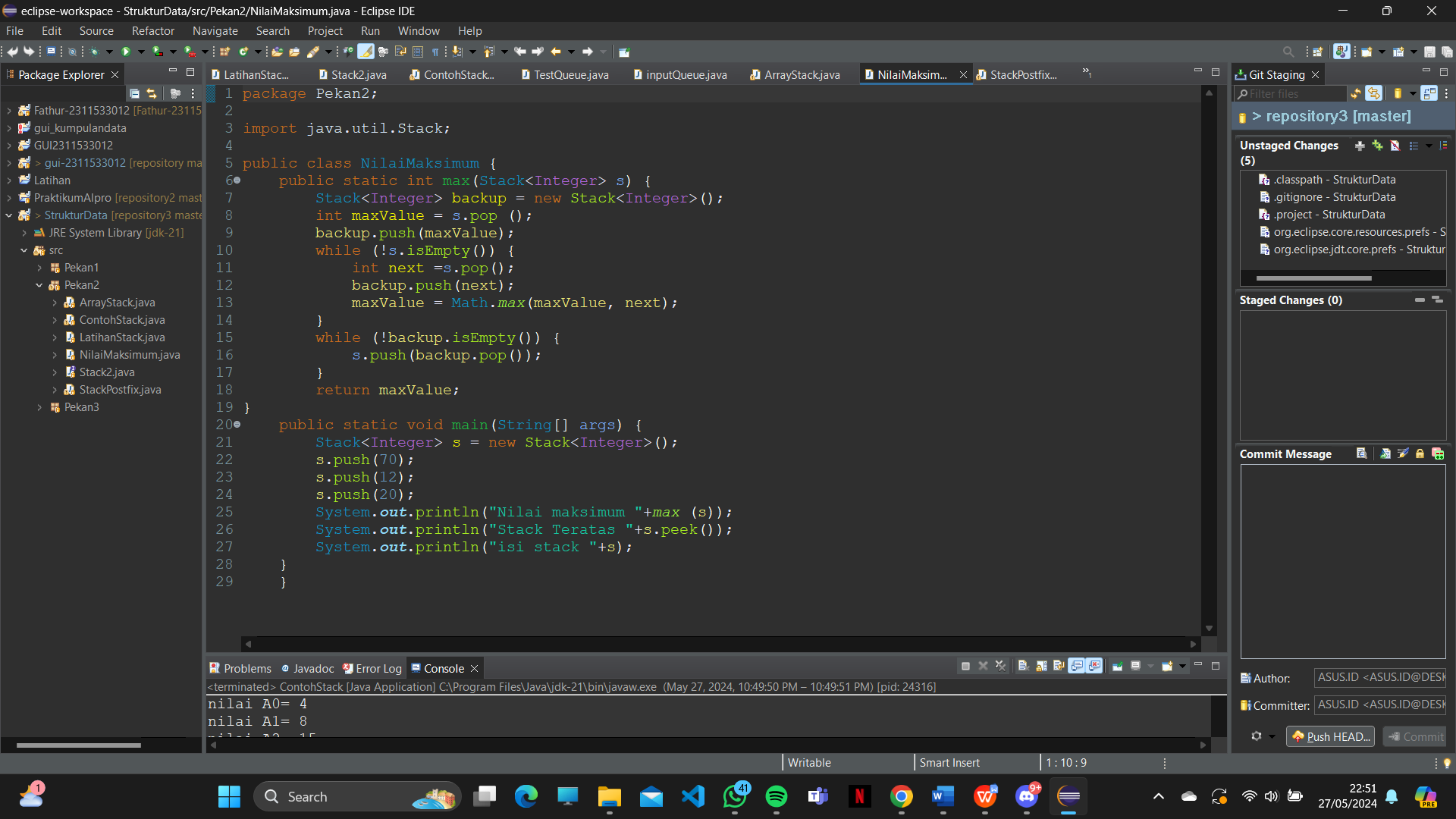


Output :

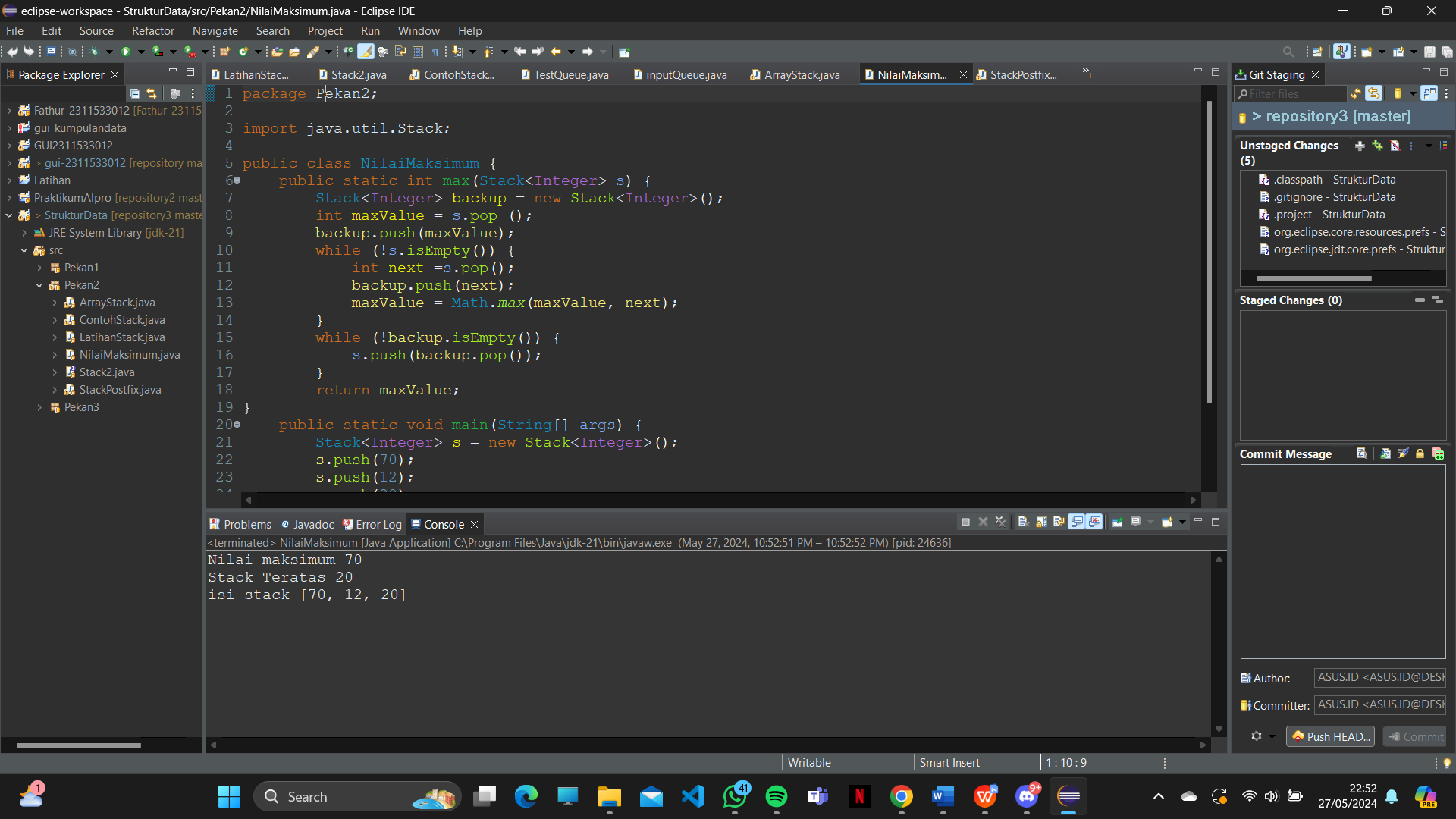


* Class NilaiMaksimum

Input :

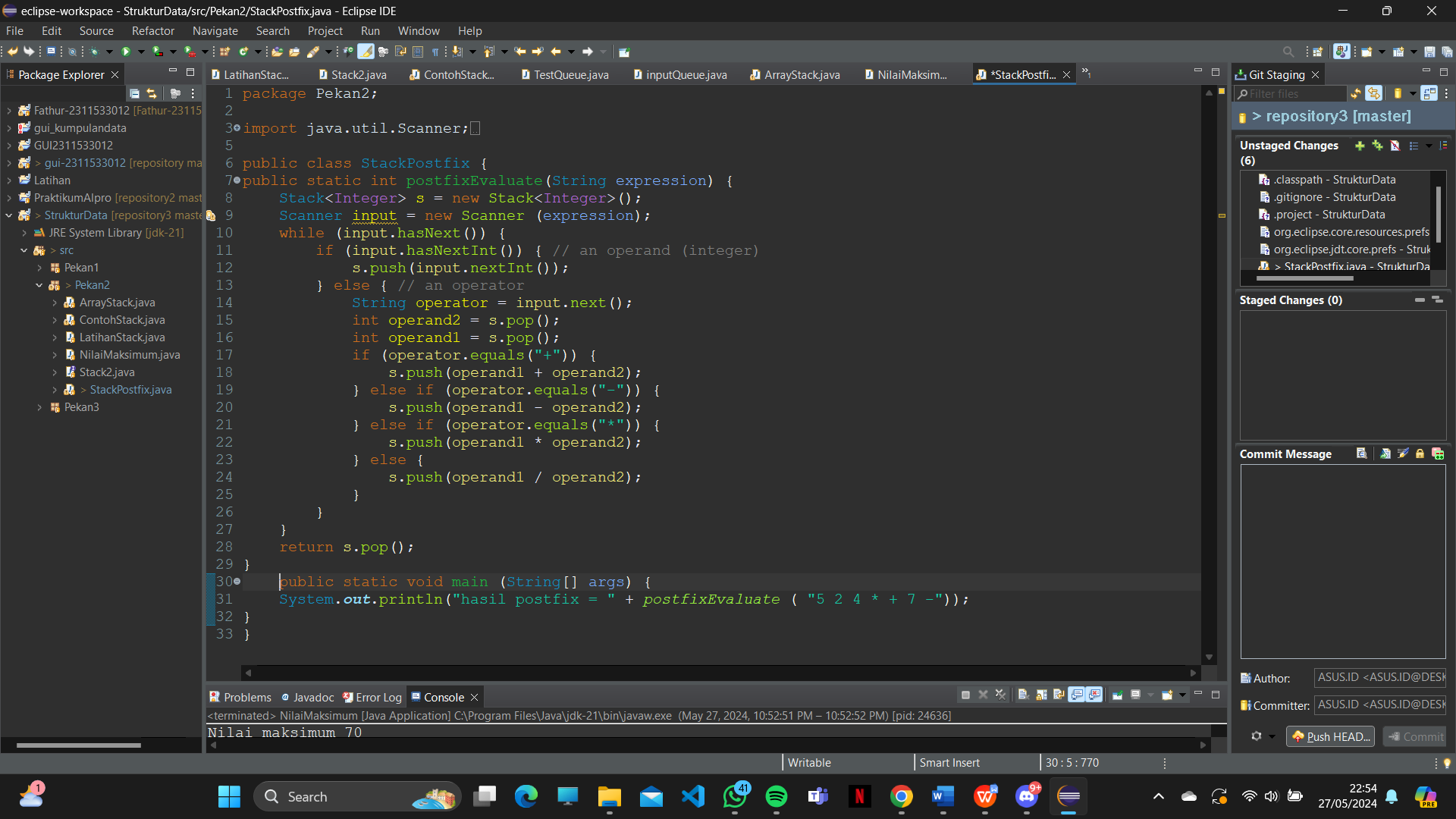


Output :



* Class StackPostfix

Input :



Output :

hasil postfix = 6