Data wykonania projektu: 20.11.2023

Data oddania projektu: 02.12.2023

POLITECHNIKA ŚLĄSKA

WYDZIAŁ MATEMATYKI STOSOWANEJ

Autor: Liliana Kołczyk

Numer albumu: 305298

TEMAT:

MNOŻENIE BEZ KALKULATORA

LISTOPAD 2023, GLIWICE

1. Opis pobieranych danych przez program – wejście do programu :

* Pierwsza liczba zmiennoprzecinkowa (a):

Program rozpoczyna od oczekiwania na wprowadzenie przez użytkownika pierwszej liczby zmiennoprzecinkowej.

Komunikat wyświetlany na konsoli informuje użytkownika o potrzebie podania tej wartości.

Użytkownik wprowadza liczbę zmiennoprzecinkową, która jest następnie przypisywana do zmiennej a.

* Druga liczba zmiennoprzecinkowa (b):

Program wyświetla kolejny komunikat, prosząc użytkownika o podanie drugiej liczby zmiennoprzecinkowej.

Użytkownik wprowadza drugą liczbę zmiennoprzecinkową, która jest przypisywana do zmiennej b.

*double a, b;*

* Zmienna result:

result jest zmienną, w której program przechowuje wynik mnożenia dwóch liczb zmiennoprzecinkowych.

Jest zainicjowana na 0.0 na początku programu.

W trakcie działania programu, wartość tej zmiennej jest aktualizowana w procesie mnożenia.

*double result = 0.0;*

* Zmienna bInt:

bInt to zmienna przechowująca zaokrągloną wartość liczby b do najbliższej liczby całkowitej.

Wykorzystywana jest do rozdzielenia liczby b na jej część całkowitą i ułamkową.

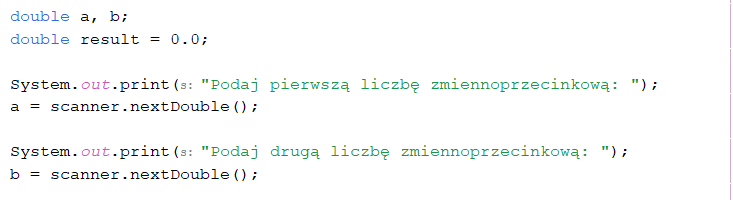
Ta wartość jest używana w pętli for do określenia, ile razy należy dodać wartość a do result (część całkowitą mnożenia).

*int bInt = (int) Math.round(b);*

Po zadeklarowaniu zmiennych pobieramy od użytkownika wartości zmiennych „a” i „b”:

*a = scanner.nextDouble();*

*b = scanner.nextDouble();*



1. Opis otrzymanych rezultatów – wydruk z programu:

Program po wykonaniu szeregu instrukcji zwraca użytkownikowi wynik mnożenie dwóch pobranych od niego liczb za pomocną instrukcji:

*System.out.printf("Iloczyn liczb %.2f i %.2f wynosi: %.2f%n", a, b, result);*

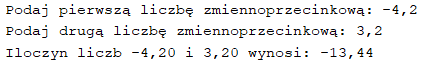


Możliwe wyniki:

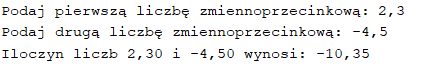
1. Mnożenie dwóch liczb dodatnich:



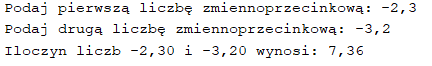
1. Mnożenie liczby ujemnej i dodatniej:



1. Mnożenie liczby dodatniej i ujemnej:



1. Mnożenie dwóch liczb ujemnych:



1. Zastosowanych algorytm do rozwiązania zadania.
2. Inicjalizacja obiektu Scanner dla wejścia z konsoli.

2. Deklaracja zmiennych:

a. double a - pierwsza liczba zmiennoprzecinkowa

b. double b - druga liczba zmiennoprzecinkowa

c. double result - wynik iloczynu

d. int bInt - liczba całkowita (wartość bezwzględna z b)

3. Wprowadzenie pierwszej liczby zmiennoprzecinkowej od użytkownika i przypisanie do zmiennej a.

4. Wprowadzenie drugiej liczby zmiennoprzecinkowej od użytkownika i przypisanie do zmiennej b.

5. Obliczenia iloczynu:

a. Zainicjowanie pętli for iterującej bInt razy:

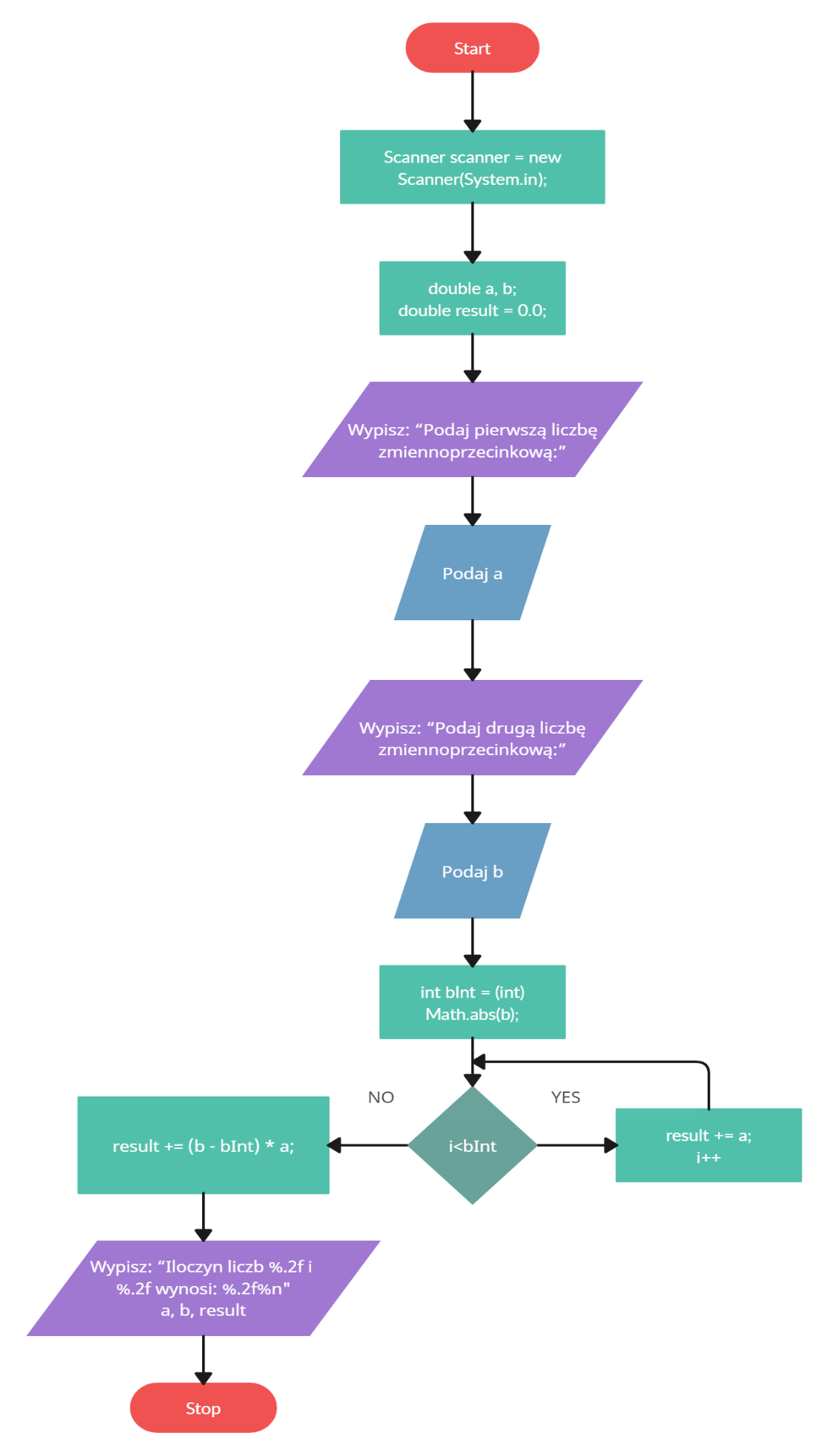
i. Dodanie wartości a do result.

b. Dodanie reszty iloczynu (b - bInt) \* a do result.

6. Wyświetlenie wyniku iloczynu z dwiema liczbami zmiennoprzecinkowymi w formacie "%.2f".

7. Zamknięcie programu.

b.) Schemat blokowy programu:



c.) Zastosowane funkcje i procedury w programie:

public static void main(String[] args): Jest to główna funkcja programu, w której rozpoczyna się jego wykonywanie. W tej funkcji znajduje się kod do interakcji z użytkownikiem, wprowadzania danych i wykonywania obliczeń.

Scanner scanner = new Scanner(System.in): Inicjalizuje obiekt Scanner do odczytu danych z konsoli. Używany jest do wczytania liczb zmiennoprzecinkowych od użytkownika.

double a, b;: Deklaracja zmiennych a i b, które przechowują wprowadzone od użytkownika liczby zmiennoprzecinkowe.

double result = 0.0;: Zmienna result przechowuje wynik iloczynu, początkowo ustawiony na 0.0.

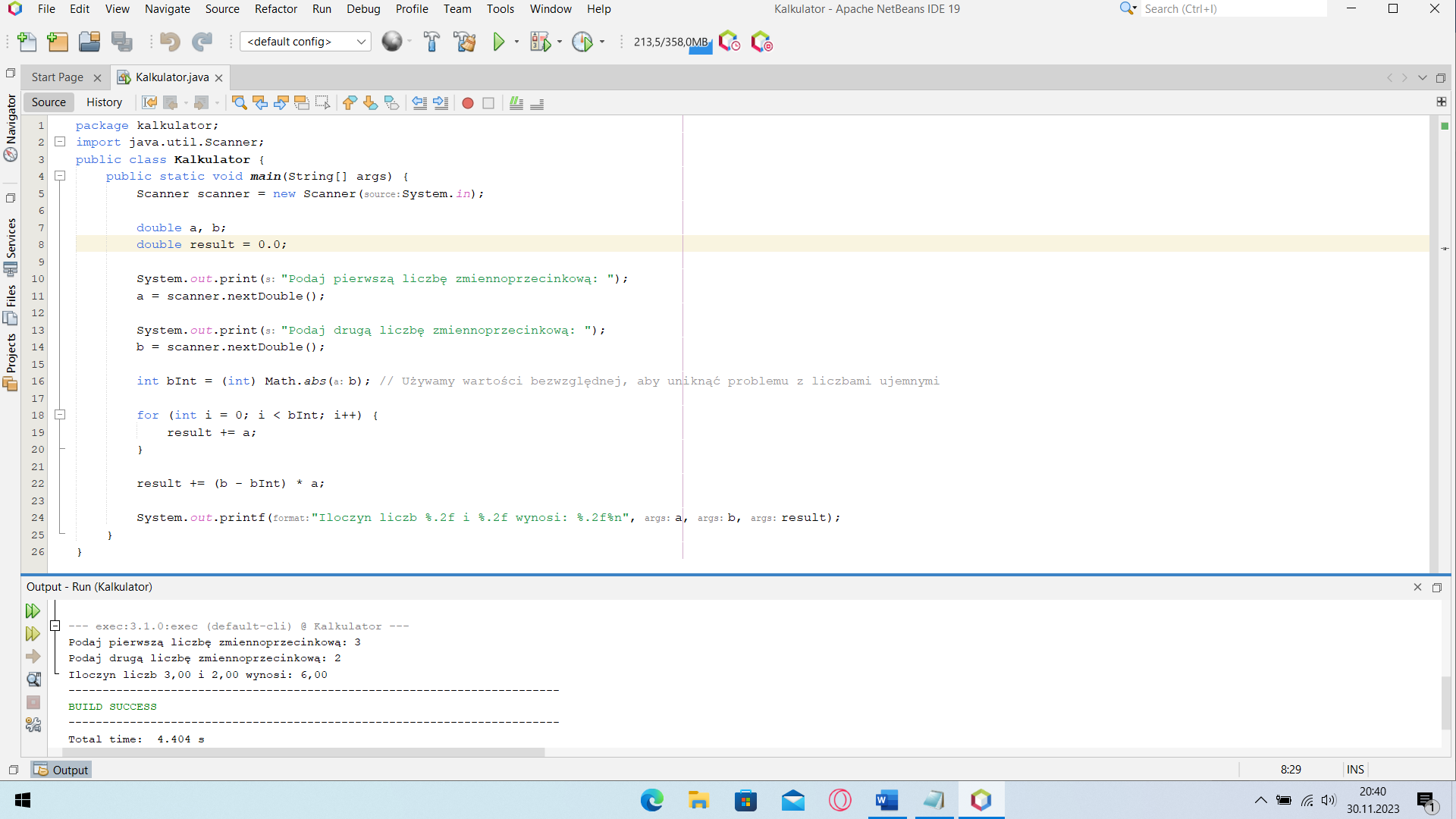
int bInt = (int) Math.abs(b);: Deklaracja zmiennej bInt, która przechowuje wartość bezwzględną z b, używając funkcji Math.abs(). Zapobiega to problemom związanych z liczbami ujemnymi.

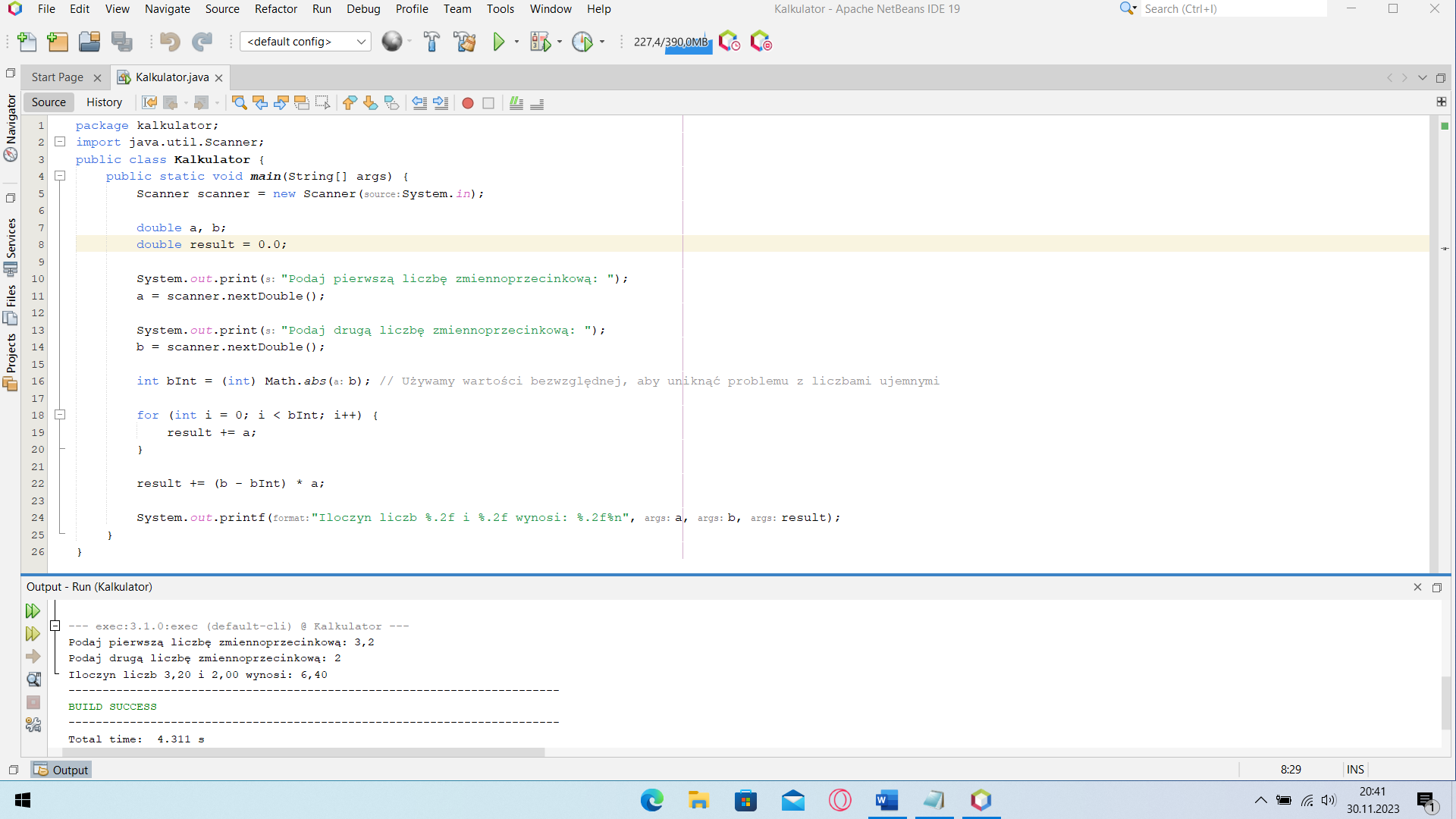
for (int i = 0; i < bInt; i++) { result += a; }: Pętla for, która wykonuje dodawanie wartości a do result bInt razy. Odpowiada to za część całkowitą iloczynu.

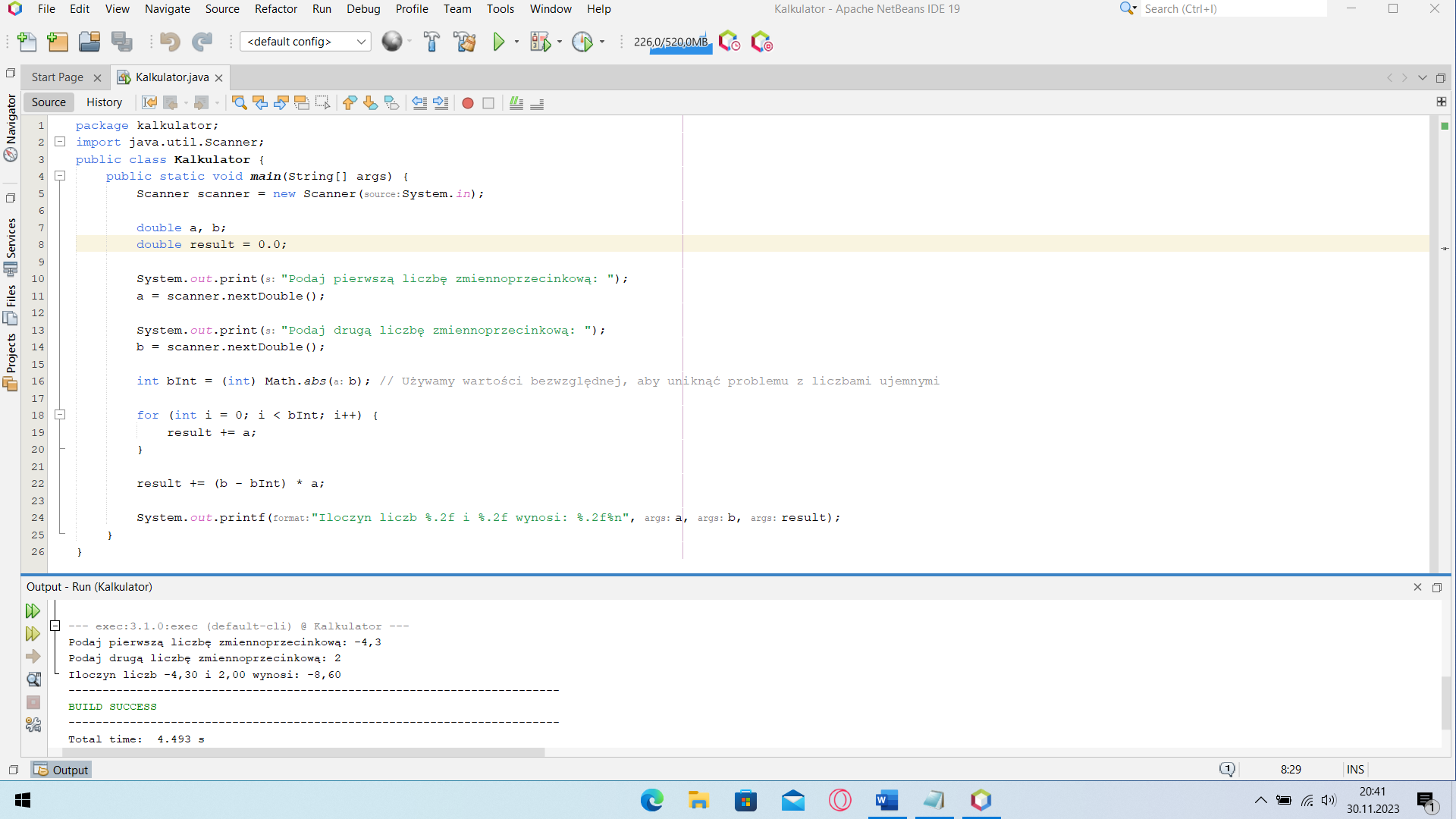
result += (b - bInt) \* a;: Dodaje resztę iloczynu, tj. (b - bInt) \* a, do result. Obsługuje część ułamkową iloczynu.

System.out.printf("Iloczyn liczb %.2f i %.2f wynosi: %.2f%n", a, b, result);: Wyświetla wynik iloczynu w ładny sposób, z dokładnością do dwóch miejsc po przecinku.

1. Testy na poprawność działania programu :







1. Wnioski:

Program wyżej przedstawiony poprawnie działa we wszystkich możliwych przypadkach oraz za każdym razem wyświetla poprawny wynik mnożenia. Kryterium mówiące, że nie można używać mnożenia podczas wykonywania takiego programu, zostało spełnione i zamiast mnożenia zostało użyte dodawanie, które było dozwolone.