

Wymagania funkcjonalne

Menu główne

Program powinien wyświetlać następujące opcje:

- 1) Dodawanie
- 2) Odejmowanie
- 3) Mnożenie
- 4) Dzielenie
- 5) Potęgowanie
- 6) Pierwiastkowanie
- 7) Wyświetlanie historii
- 8) Czyszczenie historii
- 9) Zakończenie programu

Operacje matematyczne

Dla każdej operacji program powinien:

- poprosić użytkownika o odpowiednie liczby,
- wykonać obliczenia,
- wyświetlić wynik na ekranie,

zapisać działanie i wynik do pliku historia.txt w formacie:

a + b = wynik

Obsługa wyjątków

Program powinien uwzględniać i obsługiwać następujące błędy:

Rodzaj błędu	Typ wyjątku	Opis
Błędne dane wejściowe	InputMismatchException	Wprowadzono tekst zamiast liczby
Dzielenie przez zero	ArithmeticException	Próba dzielenia przez 0
Pierwiastkowanie liczby ujemnej	IllegalArgumentException	Nie można obliczyć $\sqrt[n]{x}$ z liczby < 0

Błędy plików

IOException

Problem z zapisem lub
odczytem pliku

Historia operacji

Program powinien:

- zapisywać każdą wykonaną operację w pliku historia.txt,
 - umożliwiać wyświetlenie całej historii (czytanie z pliku linia po linii),
 - umożliwiać czyszczenie historii, czyli całkowite usunięcie zawartości pliku bez jego kasowania.
-

Biblioteki do zaimportowania

```
import java.io.*;    // Operacje na plikach (File, FileReader, FileWriter, BufferedReader,
                    // BufferedWriter)
import java.util.*;  // Klasa Scanner, obsługa wyjątków InputMismatchException
import java.lang.Math; // Operacje matematyczne (pow, sqrt)
```

Metody matematyczne

Operacja	Metoda	Przykład
Potęgowanie	Math.pow(a, b)	Math.pow(2, 3) → 8
Pierwiastkowanie	Math.sqrt(a)	Math.sqrt(9) → 3

Czyszczenie historii (pliku historia.txt)

Po wybraniu opcji 8 – Czyszczenie historii, program powinien:

- opróżnić zawartość pliku historia.txt,

wyświetlić komunikat:

Historia została wyczyszczona.

- - pozostawić plik istniejącym, ale pustym,
 - obsłużyć ewentualne błędy przy pomocy try-catch.
-

Sposób 1 – Nadpisanie pustym tekstem (zalecany, najprostszy)

Ten sposób jest najłatwiejszy i najczęściej stosowany.

Polega na otwarciu pliku w trybie nadpisywania i zapisaniu pustego ciągu znaków.

```
else if (wybor == 8) {  
    try {  
        BufferedWriter w = new BufferedWriter(new FileWriter("historia.txt"));  
        w.write(""); // Nadpisanie pliku pustą zawartością  
        w.close();  
        System.out.println("Historia została wyczyszczona.");  
    } catch (IOException e) {  
        System.out.println("Błąd podczas czyszczenia pliku!");  
    }  
}
```

Zasada działania programu

- Program działa w pętli,
- Każda operacja zapisuje wynik do pliku historia.txt,
- Historia może być odczytana (opcja 7) lub wyczyszczona (opcja 8),
- Program kończy działanie po wybraniu opcji 9 – Zakończenie programu,
- Wszystkie błędy użytkownika (np. dzielenie przez 0, złe dane) są obsługiwane w blokach try-catch.