

Akademia

Styczeń 2024

Górnictwo-Hutniczo



AGH

**AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE**

KALKULATOR NAUKOWY DLA INFORMATYKÓW/ELEKTRONIKÓW

Autor: Maciej Dawczak

Elektronika i Telekomunikacja

2021/2022

Spis Treści:

1. Wstęp
2. Wymagania Systemowe
3. Funkcjonalność i działanie programu
4. Klasa Calc
5. Testy
6. Bibliografia

1.Wstęp

Projekt ma na celu stworzenie aplikacji kalkulatora, który będzie obsługiwał podstawowe operacje matematyczne(dodawanie,odejmowanie,mnożenie,dzielenie,potęgowanie,pierwiastkowanie).Użytkownik będzie mógł przełączać się na różne systemy liczbowe: binarny, ósemkowy, szesnastkowy oraz dziesiętny.

2.Wymagania systemowe

Program został zbudowany z myślą o systemie windows, podczas przygotowywania buildu w programie Cmake wybrano x64.

Struktura programu zwizualizowana za pomocą funkcji tree w systemie Linux

```
maciej@CSCO-W-PF443CCN:/mnt/c/Users/mdawczak/source/repos/KALK$ tree -L 2
```

```
├── Calc_test
│   ├── Calc_test.vcxproj
│   ├── Calc_test.vcxproj.user
│   ├── packages.config
│   ├── pch.cpp
│   ├── pch.h
│   └── test.cpp
├── Sample-Test1
│   ├── Sample-Test1.vcxproj
│   ├── Sample-Test1.vcxproj.user
│   ├── packages.config
│   ├── pch.cpp
│   ├── pch.h
│   └── test.cpp
├── src
├── build
│   ├── ALL_BUILD.vcxproj
│   ├── ALL_BUILD.vcxproj.filters
│   ├── ALL_BUILD.vcxproj.user
│   ├── CMakeCache.txt
│   ├── CMakeFiles
│   ├── Calc.lib
│   ├── Debug
│   ├── KALK.dir
│   ├── KALK.sln
│   ├── KALK.vcxproj
│   ├── KALK.vcxproj.filters
│   ├── KALK.vcxproj.user
│   ├── ZERO_CHECK.vcxproj
│   ├── ZERO_CHECK.vcxproj.filters
│   ├── cmake_install.cmake
│   ├── CMakeLists.txt
│   ├── packages
│   └── CMakeFiles
├── lib_test
│   ├── lib_test.vcxproj
│   ├── lib_test.vcxproj.user
│   ├── packages.config
│   ├── pch.cpp
│   ├── pch.h
│   └── test.cpp
└── KALK
    ├── CMakeLists.txt
    ├── KALK.cpp
    ├── CMakeLists.txt
    ├── Calc.cpp
    └── Calc.h
```

3.Funkcjonalność i działanie programu

Na początku działania programu użytkownik zostaje powitany wiadomością „Hello World!” oraz poproszony o wybór operacji którą chce przeprowadzić

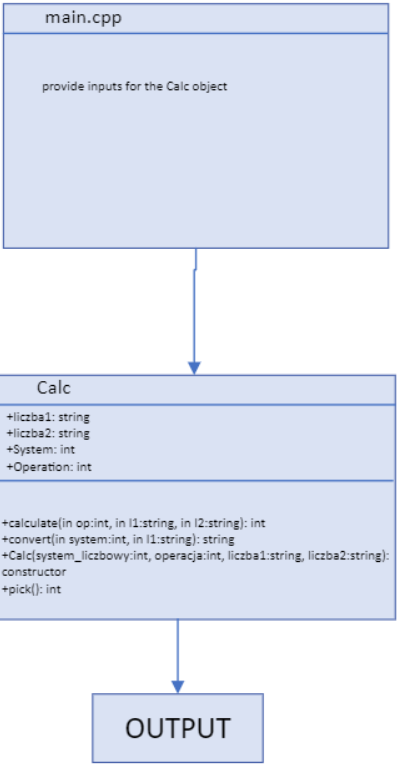
```
Hello, World!  
1: Konwersja  
2: +  
3: -  
4: *  
5: /
```

Operacja	Kod operacji
Konwersja	1
Dodawanie	2
Odejmowanie	3
Mnożenie	4
Dzielenie	5

Następnie użytkownik zostaje poproszony o wybór systemu liczbowego

```
Wybierz system liczbowy:  
1: Dziesiętny  
2: Binarny  
3: Hexadecymalny
```

System	Kod systemu
Dziesiętny	1
Binarny	2
Hexadecymalny	3



4. Klasa Cals

4.1 Metody

a) Convert

Jest to prywatna metoda klasy Calc. Wywoływana jest przez metode Calc::pick() aby metoda Calc::calculate() otrzymywała za każdym razem liczby dziesiętne. Pisząc algorytmy konwersji w tej metodzie mocno inspirowałem się kodem wygenerowanym przez ChatGPT.

b) Calculate

Ta metoda przyjmuje numer operacji oraz dwie liczby dziesiętne w postaci std:string. Następnie wykonywana jest operacja odpowiadająca numerowi przekazanemu do funkcji oraz zwracany jest wynik w postaci int.

c) Pick

W tej metodzie na podstawie switch case wybierane jest typ konwersji, następnie wywołanie kolejno metod convert() oraz calculate(). Metoda pick() jest metodą publiczną.

d) Konstruktor klasy Calc

Konstruktor przyjmuje wartości przekazane do klasy i inicjalizuje nimi daną instancję.

4.2 Atrubuty

a) liczba1

b) liczba2

c) Operation

d) System

5. Testy

Do programu zostały podłączone i wykonane testy z biblioteki gtest. W ramach projektu napisano kilka podstawowych testów sprawdzających podstawowe funkcje programu.

Test	Duration	Traits
▲ ✓ Sample-Test1 (10)	1 ms	
▲ ✓ <Empty Namespace> (10)	1 ms	
▲ ✓ CalcTest (9)	1 ms	
✓ BinAddition	< 1 ms	
✓ BinDivide	< 1 ms	
✓ BinMultiply	< 1 ms	
✓ DecimalAddition	< 1 ms	
✓ DecimalDivide	< 1 ms	
✓ DecimalMultiply	1 ms	
✓ HexAddition	< 1 ms	
✓ HexDivide	< 1 ms	
✓ HexMultiply	< 1 ms	
▲ ✓ TestCaseName (1)	< 1 ms	
✓ TestName	< 1 ms	

6. Opis

Program kalkulatora, napisany w języku C++, dostarcza funkcji do wykonywania operacji matematycznych oraz konwersji między różnymi systemami liczbowymi (dwójkowym, dziesiętnym, szesnastkowym).

Program działa w pętli, a więc możliwe jest wielokrotne wykonywanie obliczeń. Aby zakończyć działanie programu należy wcisnąć kombinację klawiszy Ctrl+C

7. Opis realizacji

Projekt był realizowany na dwóch urządzeniach i aktualizowany za pomocą systemu kontroli wersji Git. Oba urządzenia posiadają system operacyjny Windows 11 64 bitowy oraz WSL z Ubuntu.

Pisanie kodu źródłowego odbyło się w dwóch środowiskach, CLion oraz Visual Studio 2022.

8.Bibilografia

- [1] <https://www.programiz.com/cpp-programming/examples/binary-decimal-convert>
- [2] <https://www.geeksforgeeks.org/program-for-hexadecimal-to-decimal>
- [3] <https://chat.openai.com/>