

Platformy Programistyczne .NET i Java

Laboratorium 0

Aplikacja FizzBuzz

prowadzący: *Dr inż. Radosław Idzikowski, mgr inż. Michał Jaroszczuk*

1 Cel laboratorium

Celem laboratorium jest stworzenie pierwszej, prostej aplikacji w technologii .NET w celu przetestowania poprawności konfiguracji środowiska i wysłanie jej na zdalne repozytorium `github`. W ramach zajęć będzie do napisania aplikacja konsolowa w języku `C#`. Jako IDE zostanie użyty `Microsoft Visual Studio 2022` pod systemem operacyjnym `Microsoft Windows`. Praca będzie oceniana na bieżąco podczas zajęć.

2 Zadania

W ramach zajęć należy w zespołach wykonać następujące zadania:

1. Utworzyć projekt aplikacji konsolowej w języku `C#`
2. Napisać program `FizzBuzz` zgodnie z założeniami opisanymi w rozdziale 3.
3. Połączyć repozytorium `github` z `Microsoft Visual Studio 2022`, wypchnąć na nie napisaną aplikację i udostępnić repozytorium prowadzącemu.

Za wykonanie zadania nr 1 jest ocena dostateczna, za każde kolejne zadanie jest +1 do oceny. Na ocenę bardzo dobrą (5.0) należy wykonać wszystkie trzy zadania. Uzyskana ocena wlicza się do końcowej średniej z wagą 1. Link do repozytorium należy przesłać dopiero po oddaniu i ocenieniu pracy na laboratorium.

3 Opis zadań

W tej sekcji zostaną kolejno omówione wszystkie zadania do wykonania podczas laboratorium. Komputery w laboratorium zostały przygotowane do przeprowadzenia zajęć.

3.1 Zadanie 1

W ramach tego zadania będzie do wykonana nasza pierwsza podstawowa aplikacja w technologii .NET. W tym celu należy uruchomić `Microsoft Visual Studio 2022` oraz utworzyć nowy projekt `Console App` (Aplikacja konsolowa) zgodna z `Framework .NET 8.0 (LTS)`. **UWAGA!** Domyslnie od .NET 6.0 aplikacja zostanie utworzona z użyciem `top-level-statements` (instrukcje najwyższego poziomu), tzn. nie będzie **jawnie** zdefiniowana klasa `Program` oraz metoda `Main`. Po zaznaczeniu opcji, aby ich **nie** używać, otrzymamy poniższy kod:

```

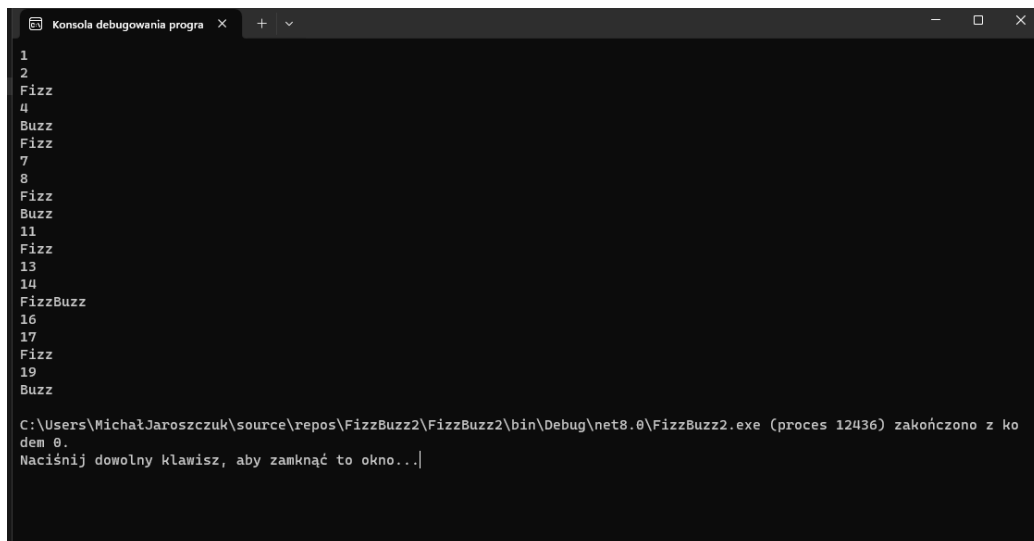
1 namespace ConsoleApp1
2 {
3     internal class Program
4     {
5         static void Main(string[] args)
6         {
7             Console.WriteLine("Hello, World!");
8         }
9     }
10 }

```

3.2 Zadanie 2

Zadanie FizzBuzz jest popularnym zadaniem rekrutacyjnym, mającym na celu sprawdzenie podstawowych umiejętności logicznego myślenia i poruszania się po IDE, kandydata na stanowisko programisty. W celu wykonania zadania należy:

1. Stworzyć osobną klasę FizzBuzz, zawierającą pole definiujące górny zakres liczb,
2. Stworzyć konstruktor klasy pozwalający na przyjęcie górnego zakresu liczb od użytkownika przy tworzeniu obiektu,
3. Stworzyć metodę pozwalającą na wyświetlenie w konsoli kolejnych liczb z podanego zakresu w osobnych liniach z założeniem, że:
 - Jeśli liczba jest podzielna przez 3, program wypisuje „Fizz”,
 - Jeśli liczba jest podzielna przez 5, program wypisuje „Buzz”
 - Jeśli liczba jest podzielna przez 3 i przez 5, program wypisuje „FizzBuzz”
 - Jeśli liczba nie jest podzielna ani przez 3 ani przez 5, program wypisuje samą liczbę.
4. Otrzymany rezultat powinien prezentować się następująco:



```

Konsola debugowania progra x + -
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
16
17
Fizz
19
Buzz

C:\Users\MichałJaroszczuk\source\repos\FizzBuzz2\FizzBuzz2\bin\Debug\net8.0\FizzBuzz2.exe (proces 12436) zakończono z ko
dem 0.
Naciśnij dowolny klawisz, aby zamknąć to okno...

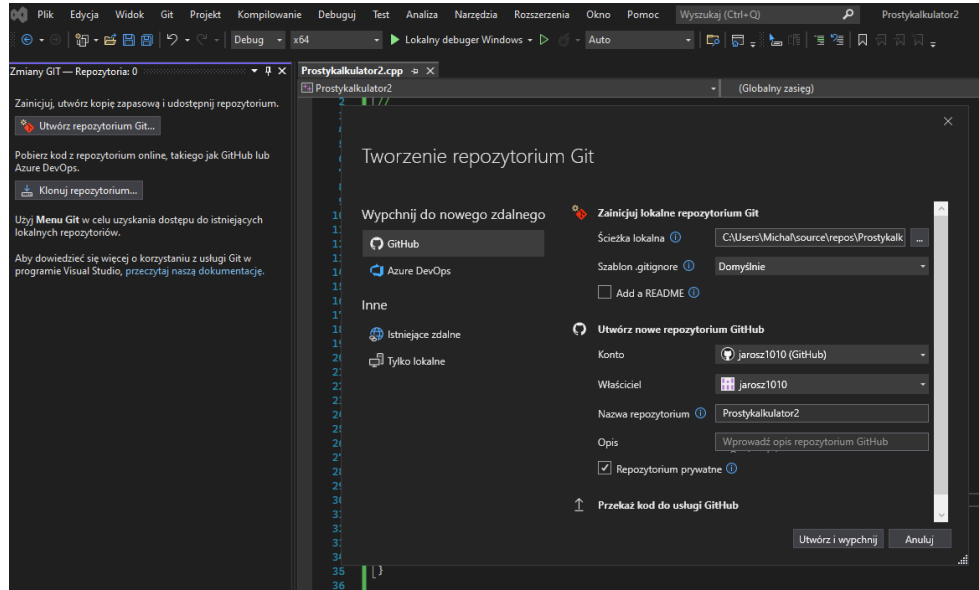
```

Rysunek 1: Wynik działania programu dla 20 liczb

3.3 Zadanie 3

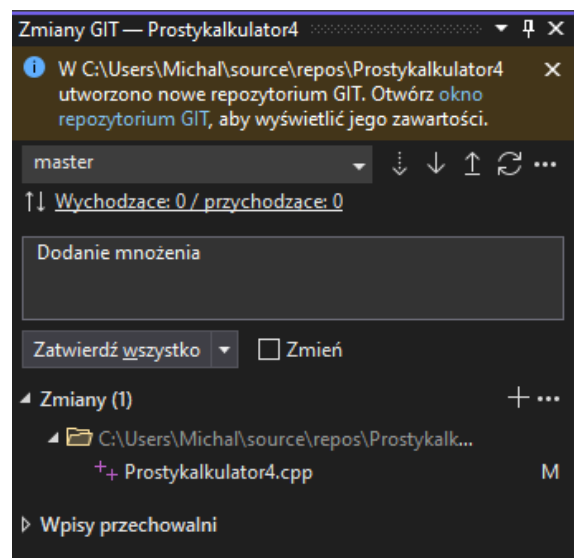
Microsoft Visual Studio 2022 pozwala na połączenie z zewnętrznym repozytorium github, dzięki czemu kontrola wersji może być obsługiwana bezpośrednio z poziomu IDE. W celu wypchnięcia programu na repozytorium zdalne należy:

1. Utworzyć lokalne repozytorium i zalogować się swoimi poświadczeniami do githuba,



Rysunek 2: Tworzenie zdalnego repozytorium GIT, z poziomu Visual Studio

2. Zatwierdzić zmiany z odpowiednim komentarzem (ang. commit),



Rysunek 3: Zatwierdzenie i wypchnięcie zmiany na zdalne repozytorium

3. Wypchnąć zmiany na repozytorium (ang. push).

Po wypchnięciu programu na zdalne repozytorium warto wylogować się z konta `git` w programie `Visual Studio` aby zabezpieczyć je przed dostępem niepowołanych osób (zwłaszcza podczas pracy na komputerach znajdujących się w laboratorium). Można zrobić to usuwając powiązanie z kontem `git` w `File` → `Account Settings`.