

## **ZASADY DZIAŁANIA PROGRAMU [GRY] ORAZ INSTRUKCJE, JAK NALEŻY KORZYSTAĆ Z PROGRAMU**

Niniejszy projekt, a właściwie program / aplikacja została napisana w języku C# .NET z wykorzystaniem WinForms [czyli Windows Forms] i przedstawia grę PolishBrickBreaker.

Niniejszy program, gra jest aplikacją okienkową [Windows Forms] i została wykonana z użyciem programu Microsoft Visual Studio 2022, jak i również programu graficznego Krita.

Kritę wykorzystano do zaprojektowania piłeczki [ball], która została zaimplementowana w programie Visual Studio, gdzie resztę elementów zaprojektowano z użyciem kolorów RGB [a dokładniej kolorów tęczy, czyli: red (czerwony), orange (pomarańczowy), yellow (żółty), green (zielony), blue (niebieski), indigo (indygo), violet (fioletowy)].

W aplikacji konsolowej stworzono 4 klasy, gdzie:

- => pierwsza klasa to 'public static class Score\_Wynik' i ma za zadanie liczenie punktów dla gracza po zbitiu każdej cegielki [brick – gdzie kolor każdej cegielki odnosi się także do innej punktacji]; ponadto niniejsza, pierwsza klasa zawiera w sobie 4 metody
- => druga klasa to 'public class Brick\_Cegielka' i ma za zadanie generowanie, kolorowanie oraz ustawianie w odpowiedni sposób [tak, aby się na siebie nie nakładały] cegielki, które będą później musiały zostać zbite; ponadto druga klasa zawiera w sobie 6 metod
- => trzecia klasa to 'public class Paddle\_Płytki' i ma za zadanie wygenerowanie, ustawienie, podzielenie na trzy części [ale nadal wyświetlanie jako jednego elementu, gdzie lewa część będzie odbijać piłeczkę w lewo, środkowa w górę, zaś prawa w prawą stronę], nadanie interakcji, a więc ustawienie odpowiednich klawiszy do ruchu płytki [paddle] horyzontalnie, na szerokości od lewej do prawej, jak i zablokowanie możliwości wyjścia poza planszę; ponadto niniejsza, trzecia prezentowana klasa zawiera w sobie 6 metod
- => czwarta klasa to 'public class Ball\_Pileczka' i odnosi się do piłeczki, która będzie odbijać się od płytki [paddle], jak i również od krawędzi, poruszać się na osi X i Y, przyspieszać, czy też i zbijać cegielki [bricks]; metoda czwarta zawiera w sobie 5 metod

Ponadto, pomimo stworzonych 4 klas bardzo ważną rolę pełni również klasa Form1, gdzie są wywoływane funkcje ze wspomnianych powyżej czterech klas, które później wchodzi w interakcję na formularzu / planszy [form] i gdzie zachodzi następnie rozgrywka.

Aplikacja konsolowa, gra PolishBrickBreaker ma za zadanie nie tylko zapewnienie przyjemności z rozrywki, ale także zapewnienie wyzwania, a więc aby wygrać trzeba się nieco bardziej wysilić. Niniejsza gra generuje domyślnie 18 cegiełek [bricks], które trzeba zbić, aby wygrać, ale jeśli ktoś chciałby większe lub mniejsze wyzwanie to może zmienić ilość generowanych losowo cegiełek na formularzu / planszy [form].

## Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623 Z5-PolishBrickBreaker– projekt nr 5

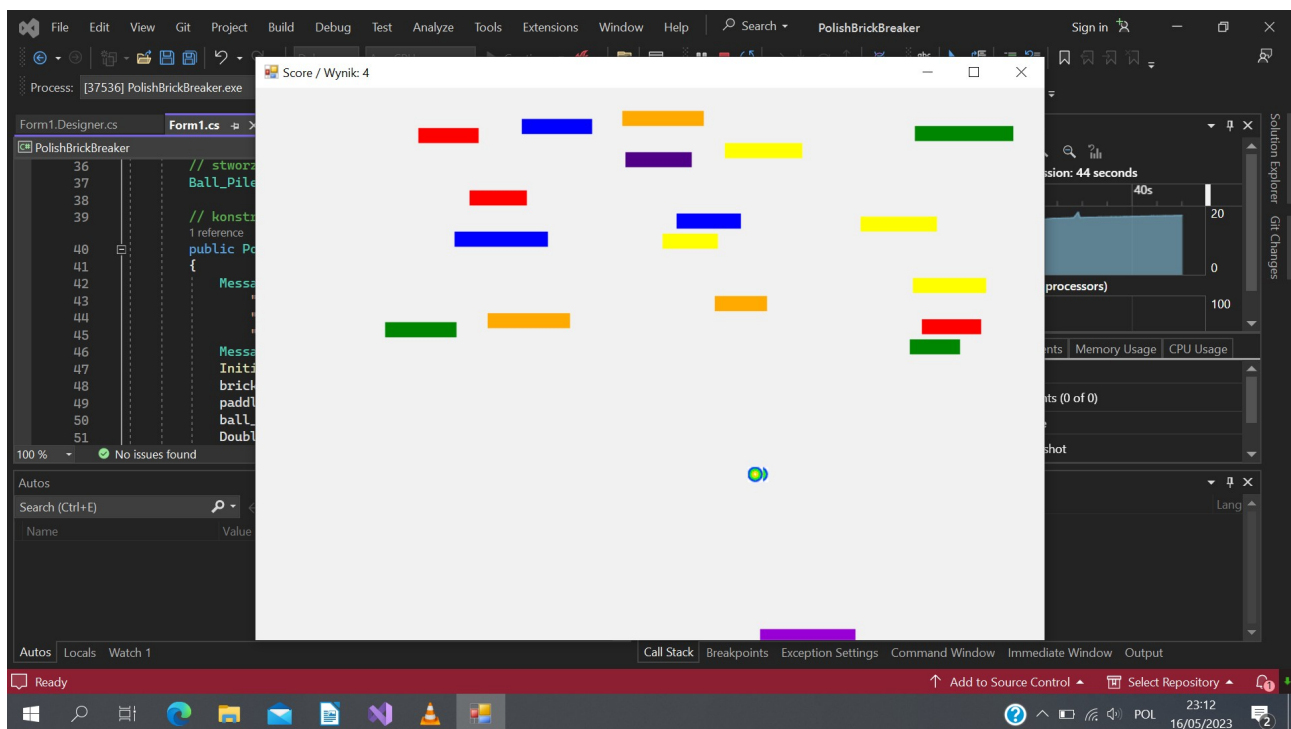
Przed rozpoczęciem gry jest wyświetlona informacja o nazwie gry, jej autorze oraz życzenie miłej rozrywki. Po przejściu przez pierwszą informację wyświetla następnie się kolejna, gdzie jest poinformowanie, iż ma się 3 piłeczki [czyli tzw. życia] oraz życzenie powodzenia w rozgrywce.

Pomimo posiadania 3 piłeczek gra zawiera w sobie spore utrudnienie, gdzie wykorzystanie trzech piłeczek jest względnie trudne ze względu na wyświetlające się informacje po utracie pierwszej i drugiej piłeczki, a więc jest wymagane szybkie wyłączenie informacji okienkowej, aby kontynuować grę [trudność tutaj polega na tym, iż gra nie jest celowo zatrzymana, lecz dalej trwa, a żeby kontynuować grę to trzeba szybko wyłączyć powiadomienia, które wyskoczą nagle po utracie pierwszej lub drugiej piłeczki]. Jeśli na początku utraci się piłeczki to jeszcze można sobie z tym poradzić, ale im później to jest tym trudniej.

Kontynuując poprzednią myśl, iż im dalej trwa gra, tym jest trudniejsza polega ona na tym, iż po uzyskaniu odpowiedniej ilości punktów prędkość piłeczki wzrasta. Domyślna prędkość piłeczki, z jaką się przemieszcza to 5 pikseli, ale w przypadku uzyskania 5 pkt to już prędkość wzrasta o 1 piksel, a więc do początkowej prędkości jest dodawane przyśpieszenie [ czyli przy 5 pkt prędkość będzie wynosić 6 pikseli (  $5 \text{ px} + 1 \text{ px} = 6 \text{ px}$  ) ]. Kolejne przyśpieszenie wynosi 3 przy uzyskaniu 10 pkt, a więc 8 pikseli [  $5 \text{ px} + 3 \text{ px} = 8 \text{ px}$  ]. W grze zostało zainicjowane 10 różnych przyśpieszeń, z czego najwyższe przyśpieszenie wynosi 11 pikseli, czyli w przypadku uzyskania 50 punktów lub więcej prędkość piłeczki wynosi 16 pikseli [  $5 \text{ px} + 11 \text{ px} = 16 \text{ px}$  ].

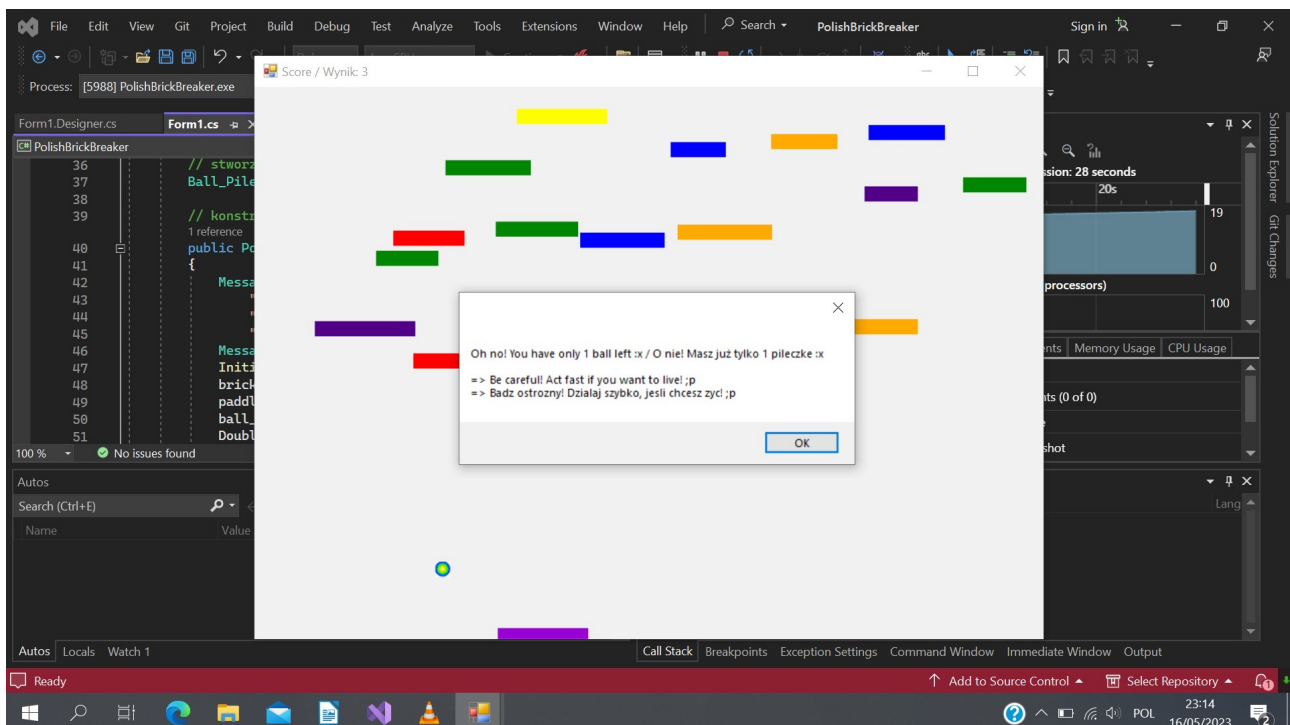
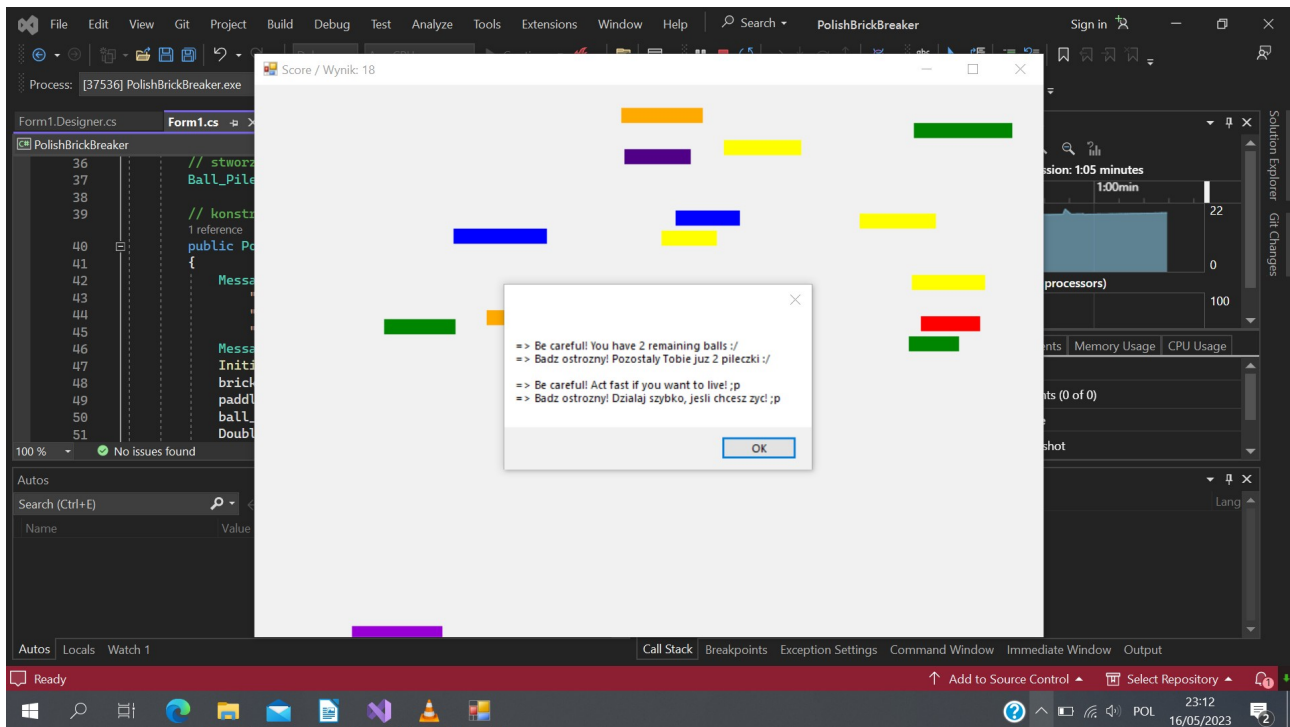
W celu ułatwienia rozgrywki prędkość płytki [paddle], którą odbija się piłeczkę wzrasta także przy osiągnięciu czwartego progu przyśpieszenia, czyli w przypadku 20 punktów. Podczas, gdy prędkość piłeczki wynosi 11 pikseli [  $5 \text{ px} + 6 \text{ px} = 11 \text{ px}$  ] to prędkość płytki wynosi 6 pikseli [ podczas gdy domyślna prędkość płytki (paddle) to 5 pikseli ].

Poniższe zrzuty ekranu przedstawiają rozgrywkę:

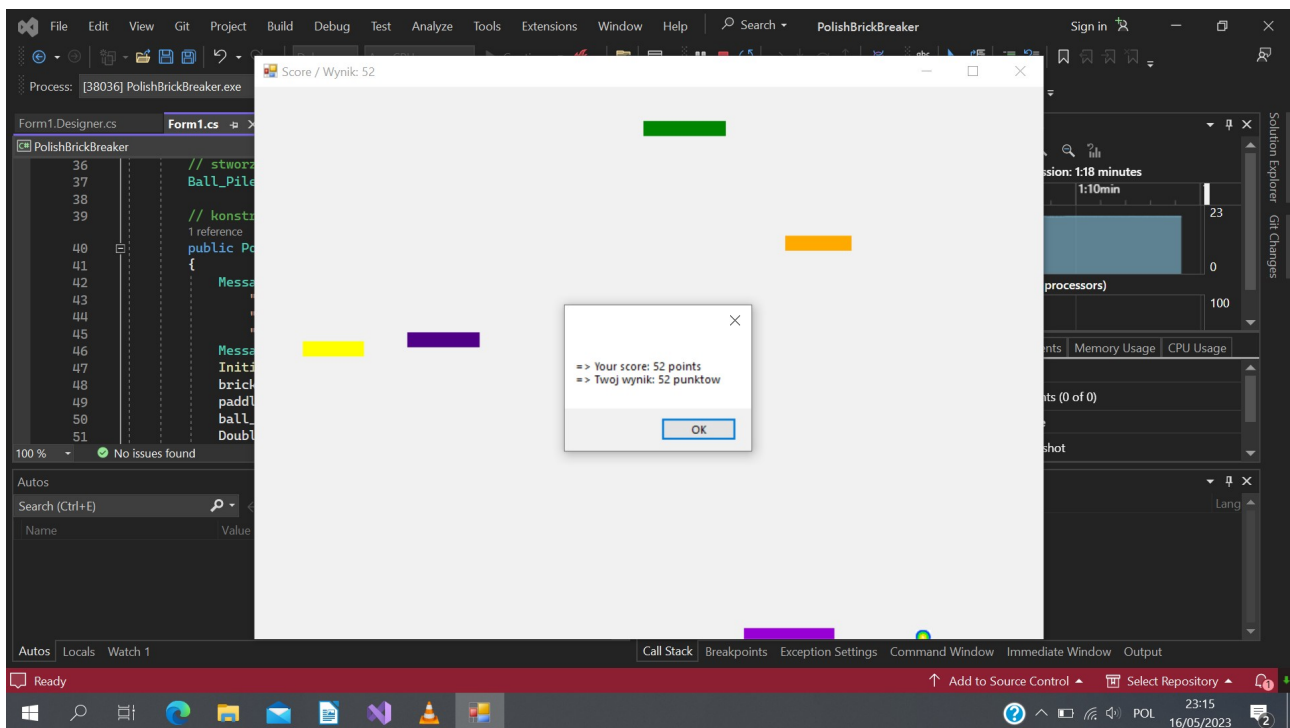
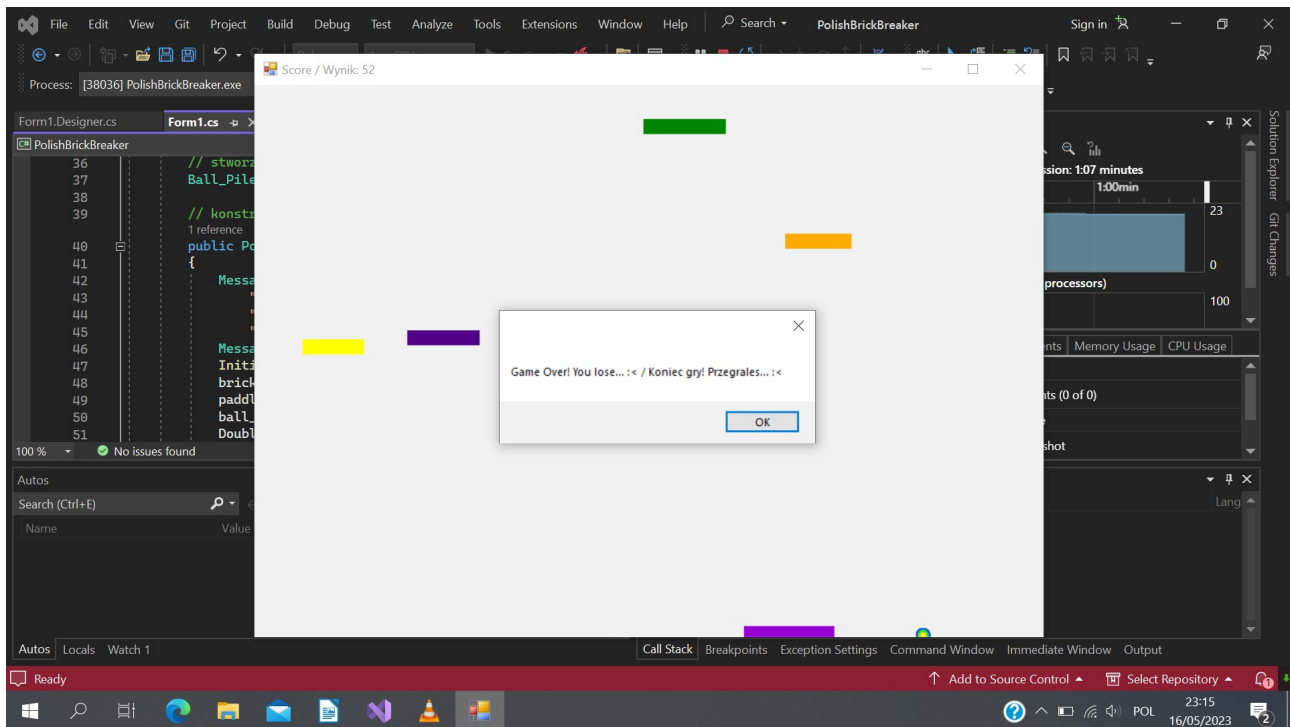


## Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623

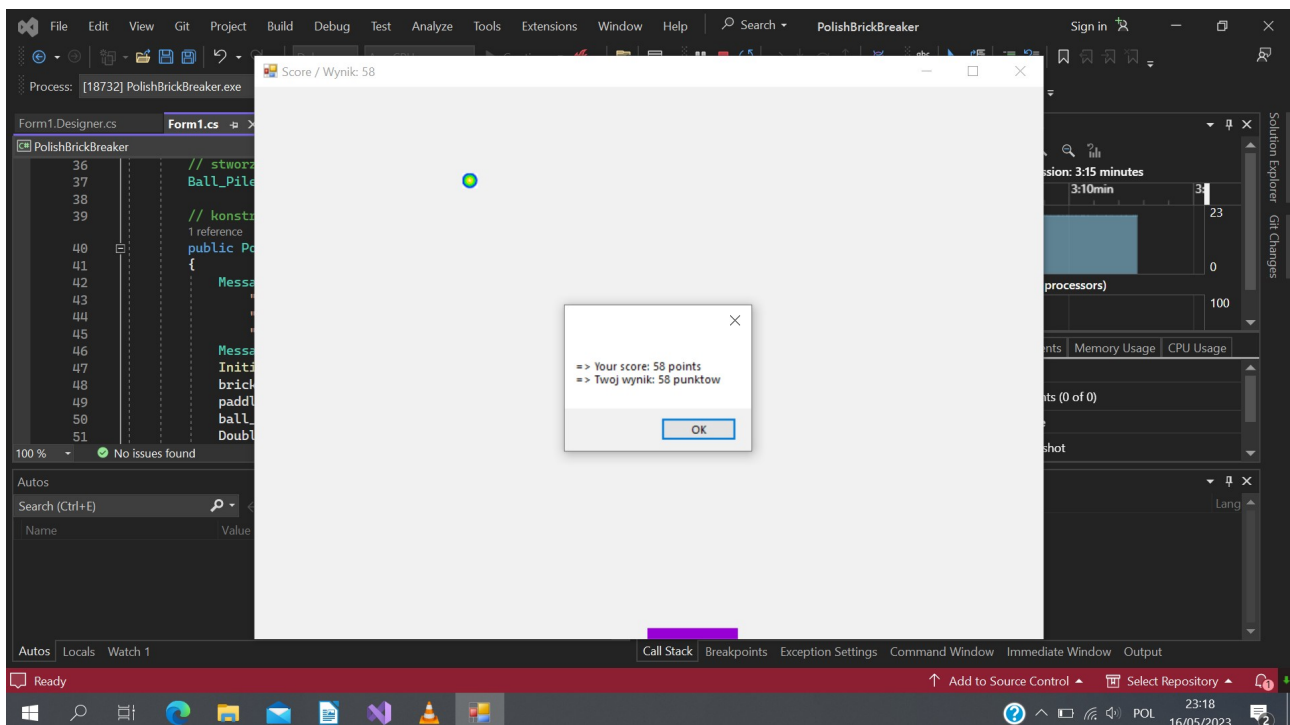
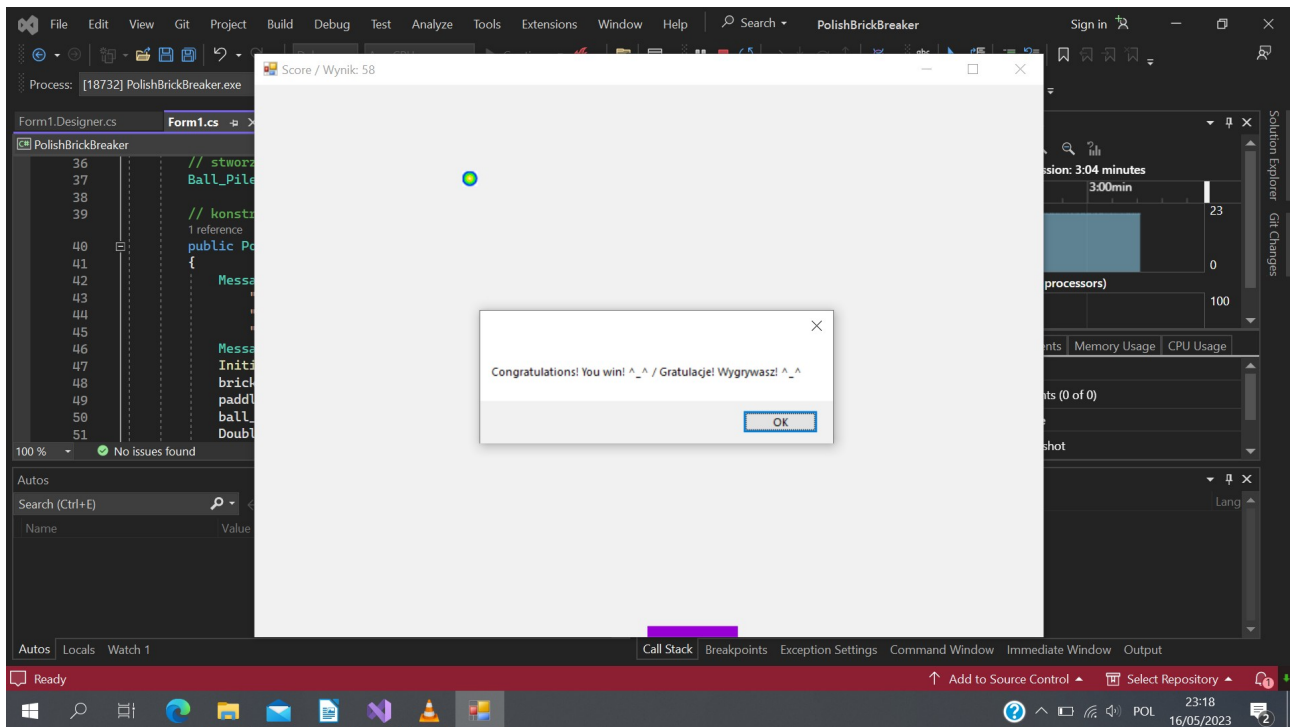
### Z5-PolishBrickBreaker– projekt nr 5



## Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623 Z5-PolishBrickBreaker– projekt nr 5



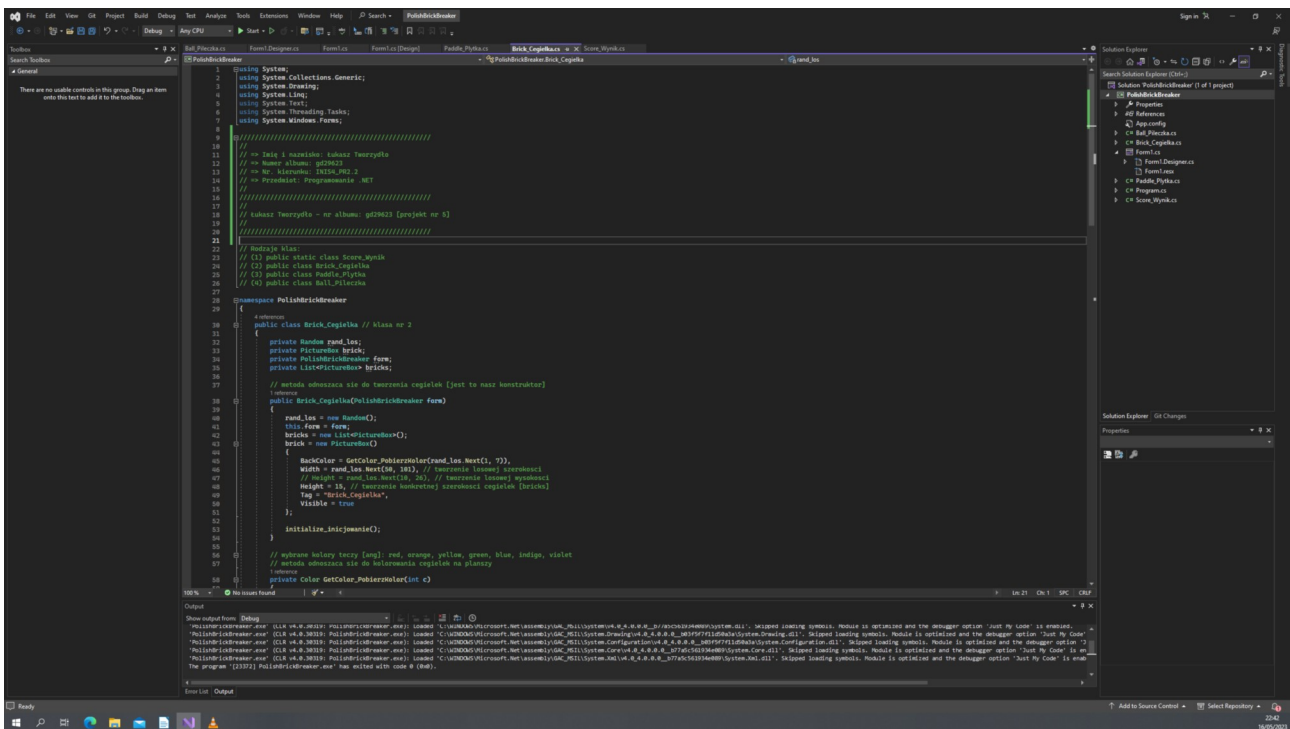
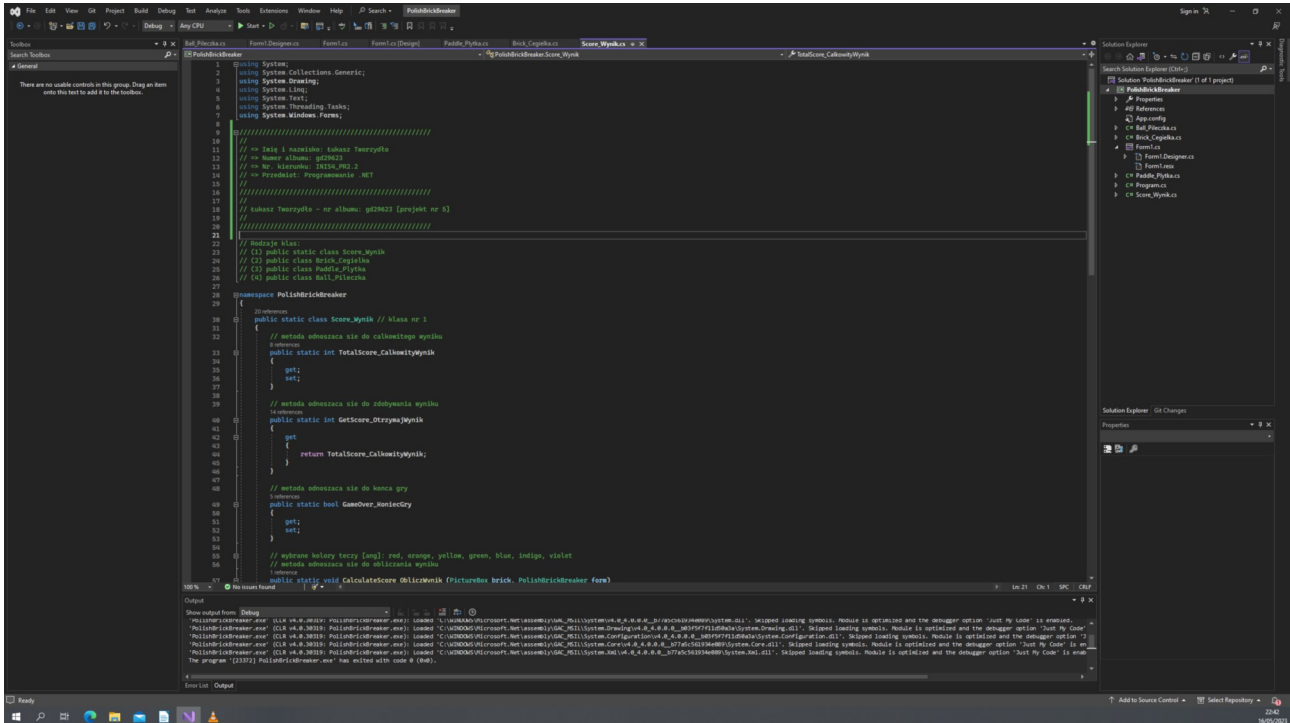
## Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623 Z5-PolishBrickBreaker– projekt nr 5



# Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623

## Z5-PolishBrickBreaker– projekt nr 5

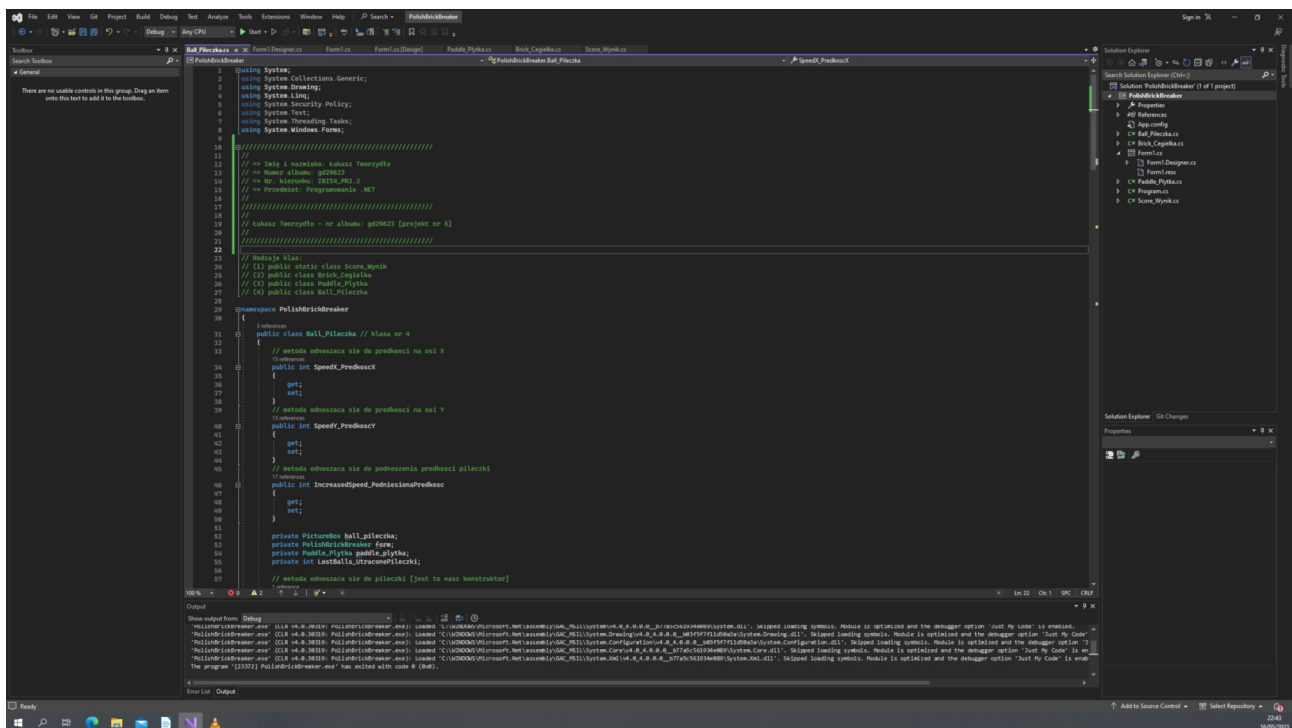
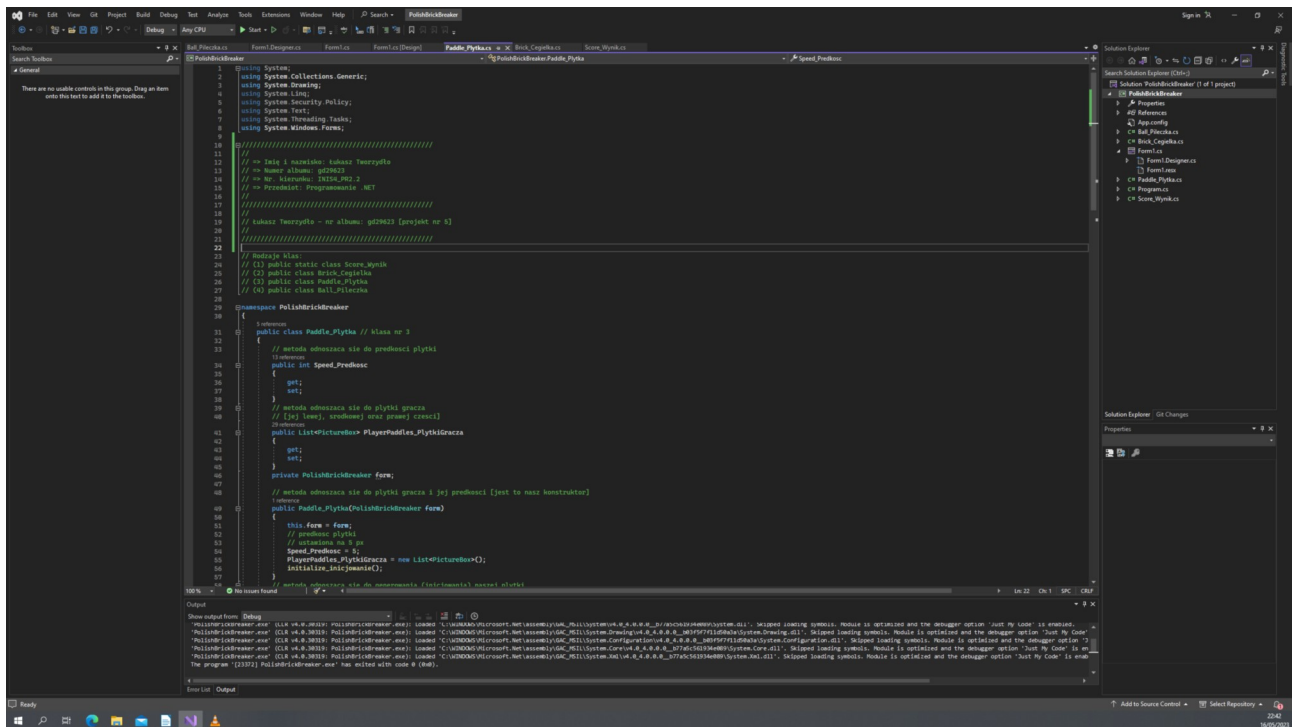
Natomiast kolejne, poniższe zrzuty przedstawiają fragmenty klas:





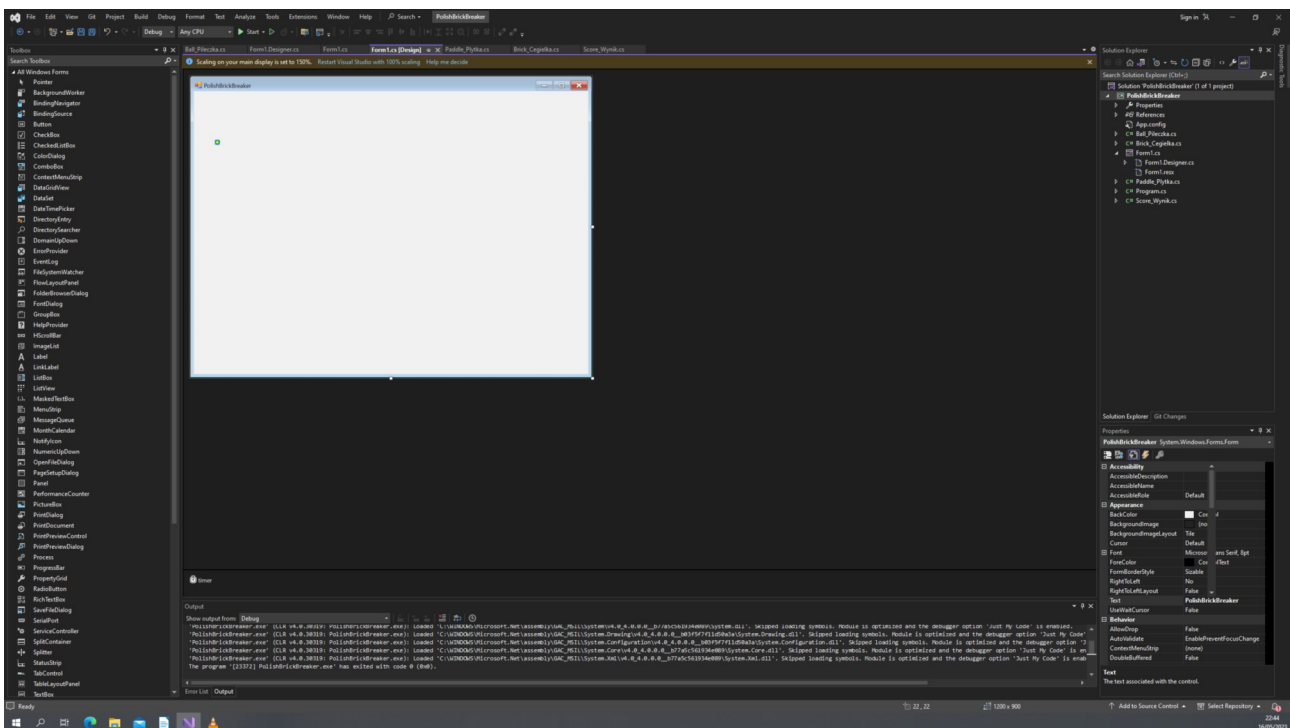
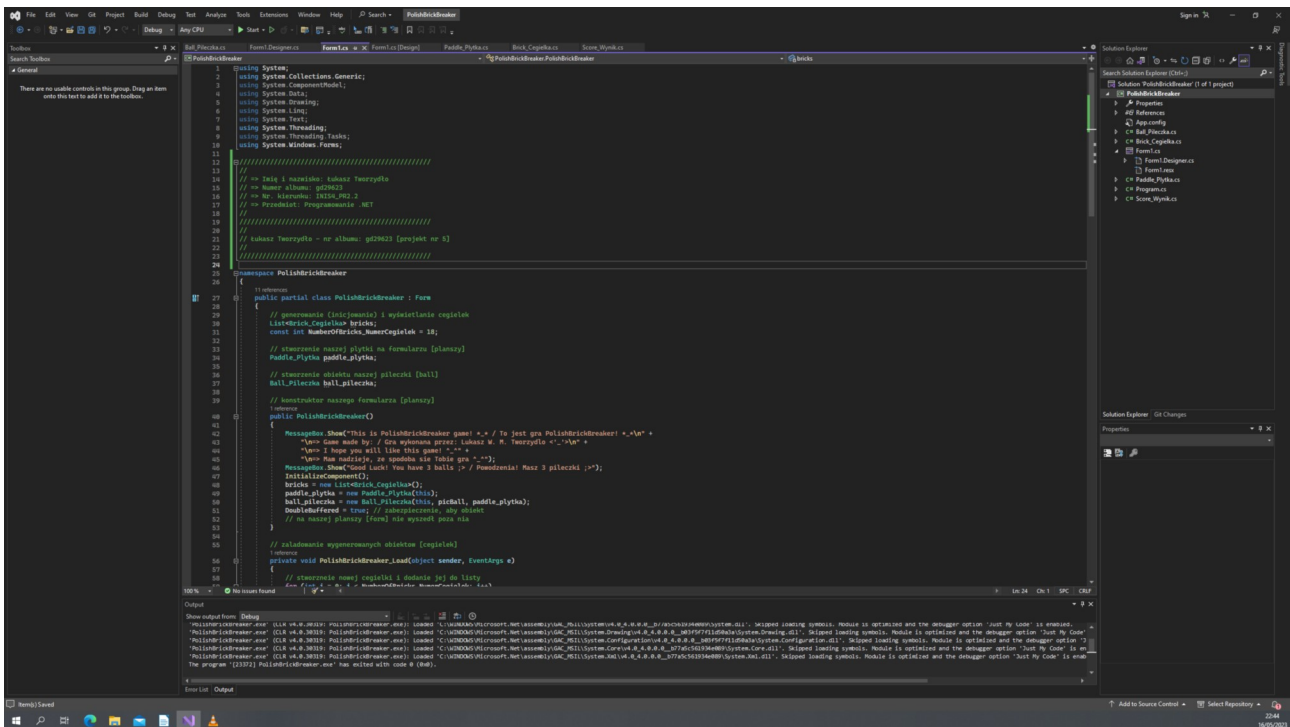
# Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623

## Z5-PolishBrickBreaker– projekt nr 5



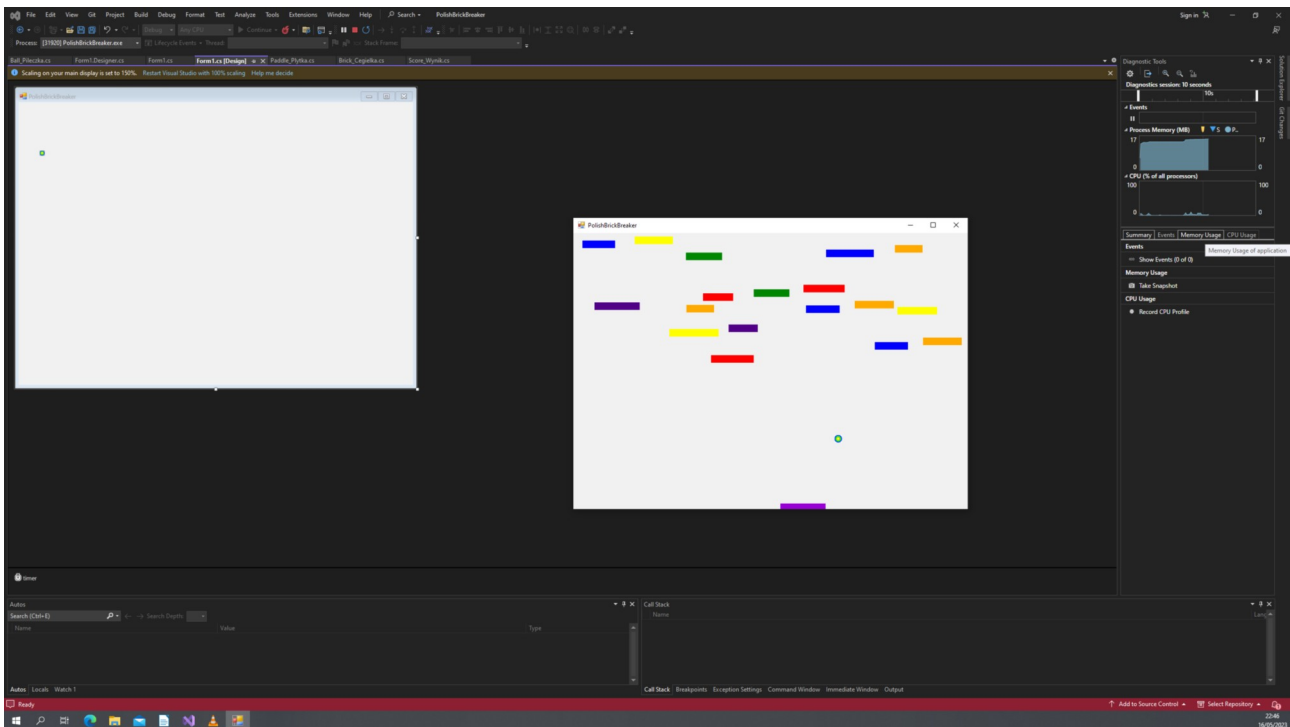
# Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623

## Z5-PolishBrickBreaker– projekt nr 5





# Programowanie .NET – INIS4\_PR2.2 – Łukasz Tworzydło – gd29623 Z5-PolishBrickBreaker– projekt nr 5



Prezentacja działania programu, gry, aplikacji okienkowej w C#. NET opisał Łukasz W. Tworzydło.

Dane studenta:

=> Imię i nazwisko: Łukasz Tworzydło

=> Numer albumu: gd29623

=> Nr. kierunku: INIS4\_PR2.2

=> Przedmiot: Programowanie .NET