

**Projekt z przedmiotu Programowanie**

**Student: Bartosz Ochędowski**

**Nr Albumu: 59300**

**Prowadzący: Ewa Żesławska**

1. ***Opis założeń projektu***

Aplikacja wyznaczająca możliwe trajektorie przejść w zadanym grafie (C#)

Zaprojektowanie aplikacji do poszukiwania optymalnej trajektorii przejść w grafie skierowanym bądź nie skierowanym, który nie jest grafem ważonym. W zadaniu tym należy wykorzystać algorytm BFS, DFS, A\*, metodę branch and bound. Aplikacja powinna umożliwiać wyświetlenie w postaci graficznej zestawu wyszukanych trajektorii.

Pliki wejściowe zadane będą w Trivial Graph Format, który opisano w dalszej części projektu. Grafy nieskierowane będą przedstawiane jako graf skierowany z podwójnymi krawędziami (od wierzchołka i do wierzchołka). Etykietami wierzchołków będą współrzędne, separowane przecinkiem bez dodatkowej spacji.

Grafy będą rysowane w przestrzeni 1024x768 z możliwymi współrzędnymi (0..102, 0..76)

1. ***Specyfikacja wymagań.***

*Wymagania funkcjonalne:*

- Program wczytuje podany graf z pliku

- Program znajduje drogę w danym grafie z zastosowaniem określonego algorytmu

- Program zapisuje jako plik graficzny reprezentację ścieżki w grafie

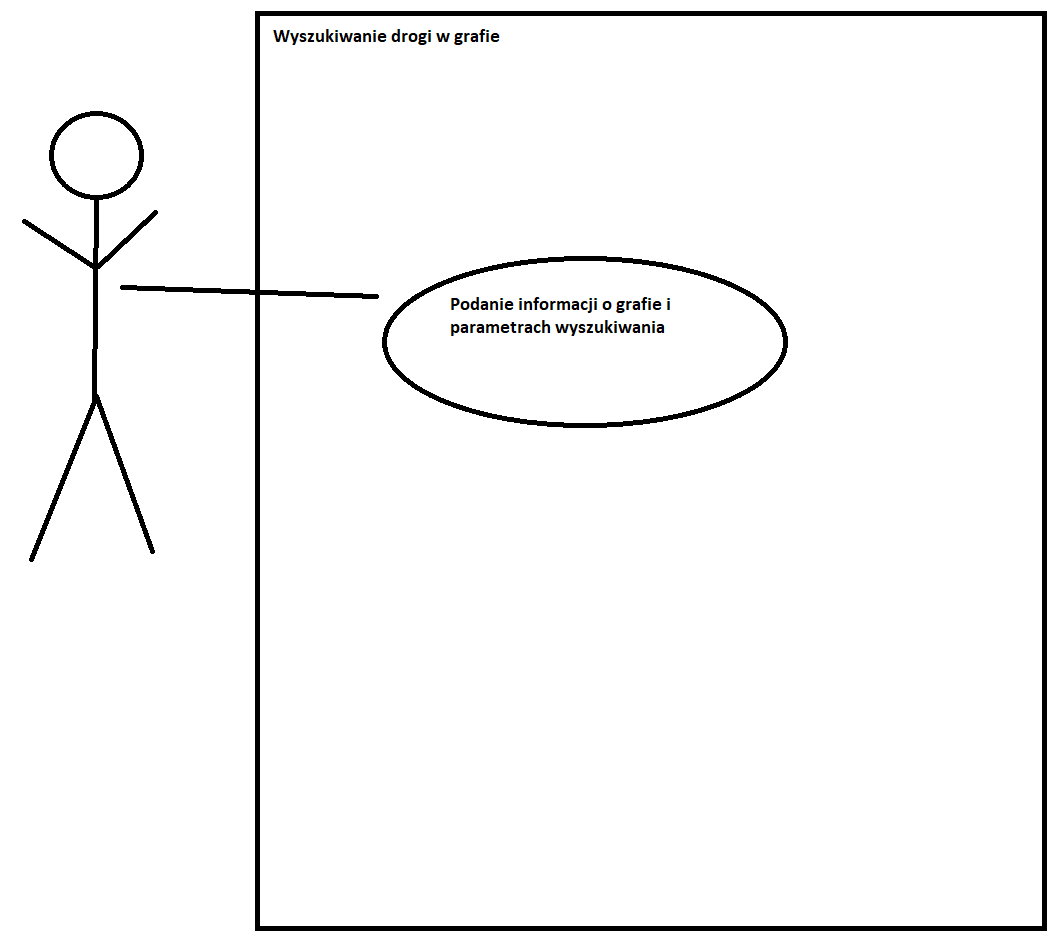
*Wymagania niefunkcjonalne:*

- Istnieje droga w podanym grafie

- Plik z podanym grafem jest poprawny i zgodny z formatem wejściowym

- Środowiskiem uruchomieniowym jest Windows co najmniej 32-bit z .NET Frameworkiem 4.0

1. ***Diagram przypadków użycia z wyróżnieniem aktorów projektu, oraz przypadków użycia odzwierciedlających wymagania funkcjonalne***



1. ***Harmonogram realizacji projektu (diagram Gantta***)



1. ***Opis techniczny projektu***

Program został wykonany przy użyciu języka C# i frameworku .NET Framework. Program na wejściu (tj. w linii poleceń) spodziewa się kolejno:

1. Nazwę pliku zawierającego opis grafu
2. Nazwę algorytmu do wyszukiwania ścieżki
3. Wierzchołek początkowy
4. Wierzchołek końcowy

*Sposób opisu grafu przy użyciu pliku tekstowego jest jak następuje:*

Wierzcholek\_grafu wspolrzedna\_x,wspolrzedna\_y  
Kolejny\_wierzcholek\_grafu wspolrzedna\_x,wspolrzedna\_y  
…  
#  
wierzcholek\_z\_ktorego\_wychodzi\_sciezka wierzcholek\_do\_ktorego\_wpada\_sciezka

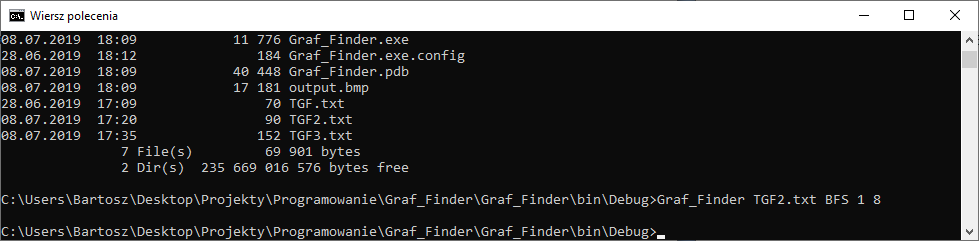
Aby stworzyć graf nieskierowany, wystarczy zduplikować połączenia między wierzchołkami, gdyż każdy graf nieskierowany jest szczególnym przypadkiem grafu skierowanego.

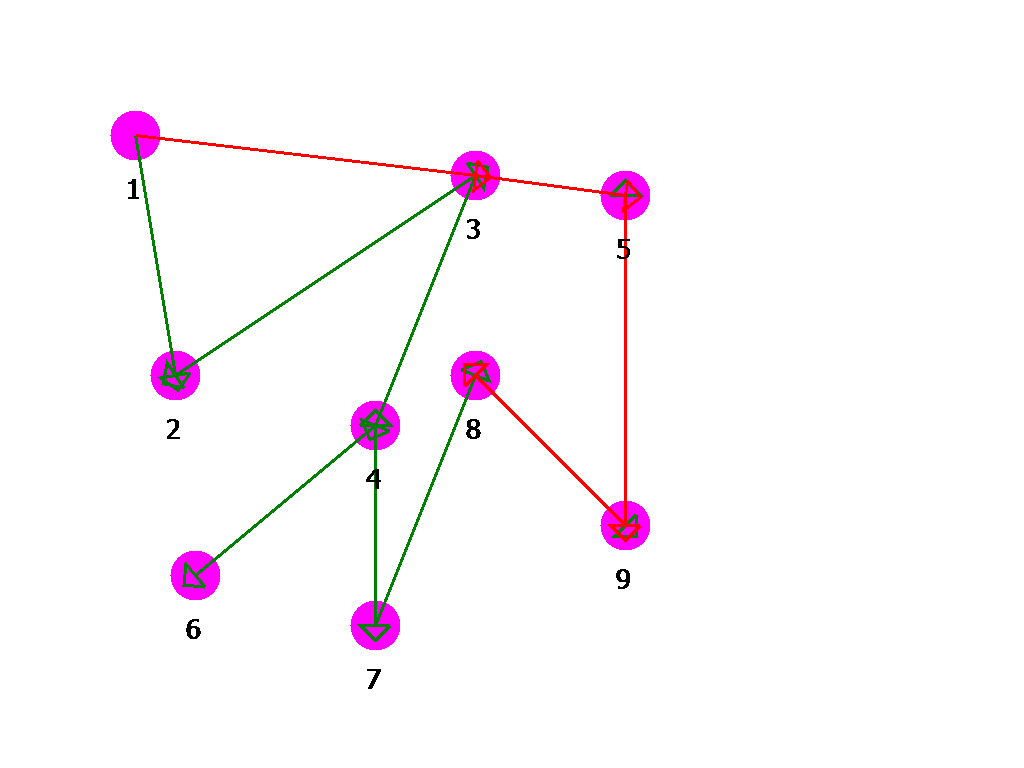
Dalej wyboru algorytmu dokonuje się poprzez wpisanie odpowiedniego łańcucha tekstowego. Możliwe to

* **DFS** – dla algorytmu DFS
* **BFS** – dla algorytmu BFS
* **Astar** – dla algorytmu A\*
* **B&B** – dla algorytmu Branch and Bound

Algorytm branch and bound wykorzystuje algorytm DFS jako heurystykę, przy podobnej strategii odwiedzania węzłów (odwiedza kolejne, jak wyszczególniono w pliku tekstowym).

1. ***Prezentacja warstwy użytkowej projektu***

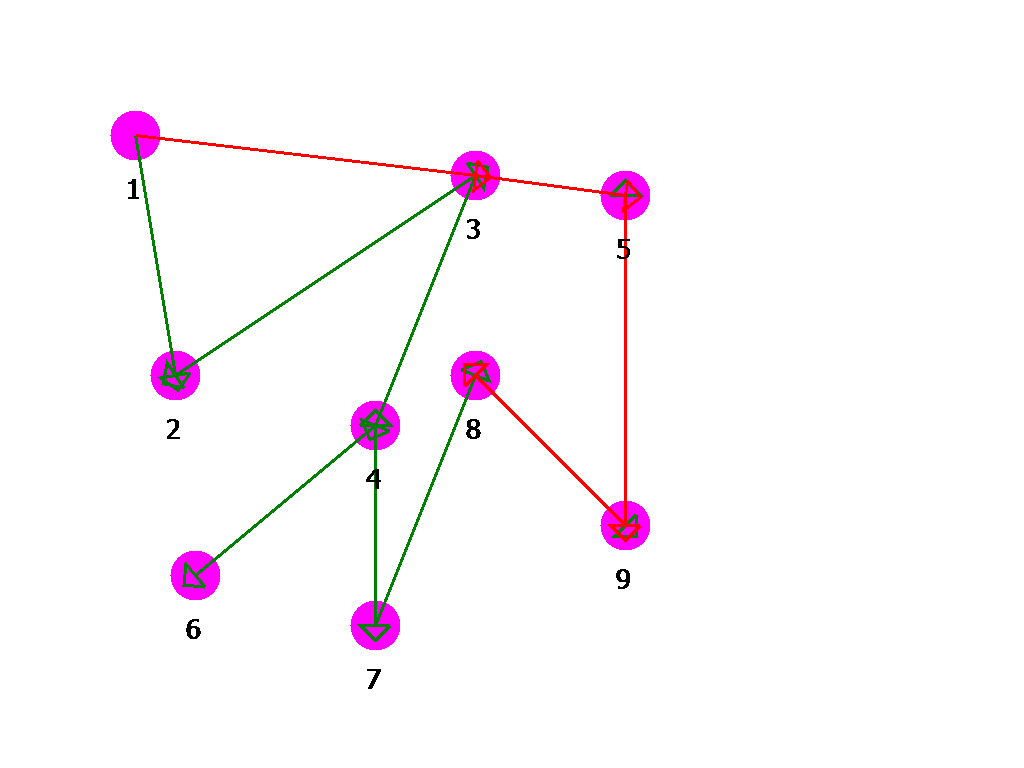
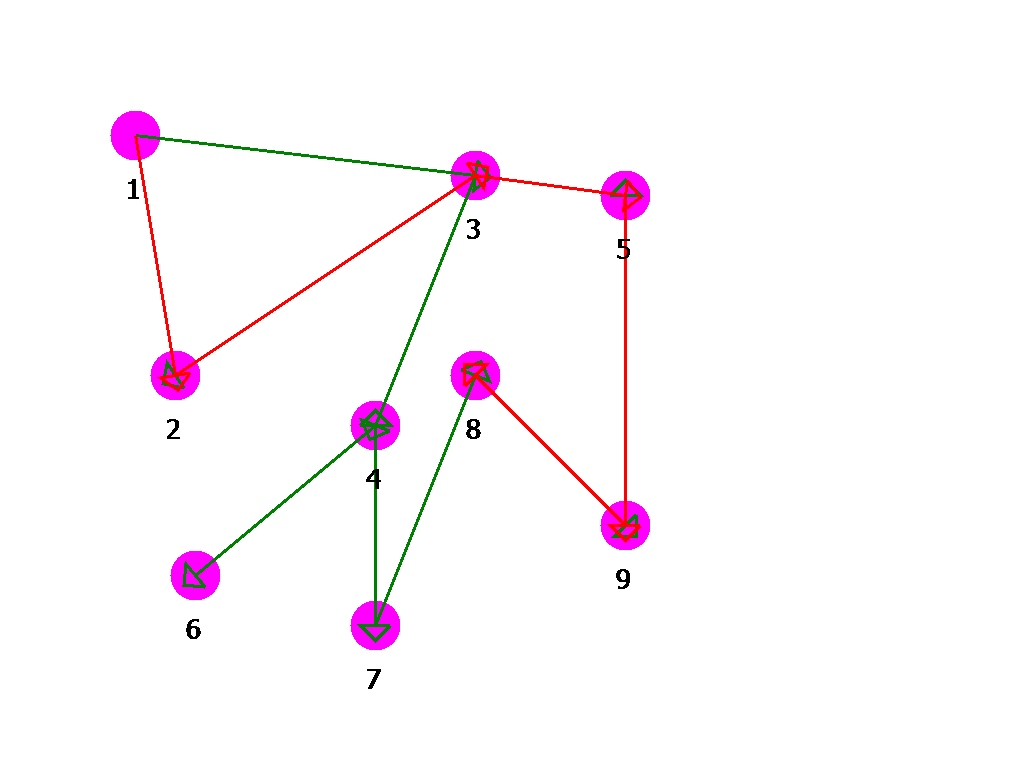
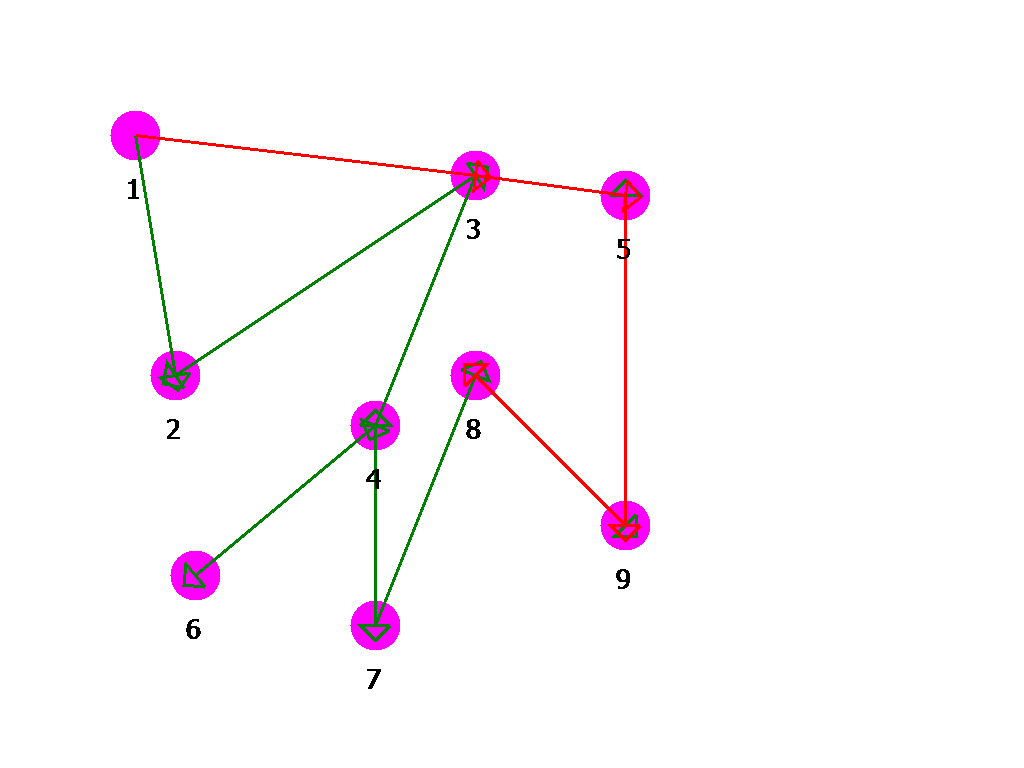




1. ***Raporty z testów – powinny być przeprowadzone testy jednostkowe.***

Sporządziłem trzy rózne grafy i sprawdziłem każdy algorytm dla kazdego z nich(ze spodziewanymi róznicami miedzy DFS a BFS czy A\*).

Przykład dla pliku TGF3.txt:



1. ***Materiały źródłowe – wskazanie literatury i materiałów źródłowych wykorzystanych przy realizacji projektu.***

trivial graph format:

<https://en.wikipedia.org/wiki/Trivial_Graph_Format>

foreach c#:

<https://docs.microsoft.com/pl-pl/dotnet/csharp/language-reference/keywords/foreach-in>

depth first serach:

<https://www.google.com/search?q=depth+first+search&oq=depth+first+search&aqs=chrome..69i57j0l5.1854j0j8&sourceid=chrome&ie=UTF-8>

how to parse text files c#:

<https://stackoverflow.com/questions/858756/how-to-parse-a-text-file-with-c-sharp>

list generic c#:

<https://docs.microsoft.com/pl-pl/dotnet/api/system.collections.generic.list-1?view=netframework-4.8>

C# hashmap:

<https://stackoverflow.com/questions/1273139/c-sharp-java-hashmap-equivalent>

Depth first serach:

<https://pl.wikipedia.org/wiki/Przeszukiwanie_w_g%C5%82%C4%85b>

How to use dictionary c#

<https://www.tutorialsteacher.com/csharp/csharp-dictionary>

c# clone a list:

<https://stackoverflow.com/questions/222598/how-do-i-clone-a-generic-list-in-c>

How to convert int to string c#:

<https://stackoverflow.com/questions/3081916/convert-int-to-string>

CS0246 C# The type or namespace name could not be found (are you missing a using directive or an assembly reference?):

<https://stackoverflow.com/questions/11854308/why-am-i-getting-error-cs0246-the-type-or-namespace-name-could-not-be-found>

C# draw a circle on a bitmap:

<https://stackoverflow.com/questions/16727802/how-to-create-circle-with-bitmap-c-sharp>

System.drawing c#:

<https://docs.microsoft.com/pl-pl/dotnet/api/system.drawing?view=netframework-4.8>

System.drawing 4.6 .net:

<https://docs.microsoft.com/pl-pl/dotnet/api/system.drawing?view=netframework-4.6>

The type or namespace name does not exist in the namespace System.Drawing:

<https://stackoverflow.com/questions/8553136/system-drawing-namespace-not-found-under-console-application>

How to draw text c#:

<https://docs.microsoft.com/pl-pl/dotnet/api/system.drawing.graphics.drawstring?view=netframework-4.8>

How to add text on drawing c#:

<https://stackoverflow.com/questions/6311545/c-sharp-write-text-on-bitmap>

Depth-first search:

<https://pl.wikipedia.org/wiki/Przeszukiwanie_w_g%C5%82%C4%85b>

DrawLine does'nt work c#:

<https://stackoverflow.com/questions/13943387/draw-line-on-panel-not-showing-up>

BFS path serach:

<https://stackoverflow.com/questions/8922060/how-to-trace-the-path-in-a-breadth-first-search>

<https://en.wikipedia.org/wiki/Breadth-first_search>

Graphicspath c#:

<https://www.google.com/search?ei=GEQWXcXXDeHLrgTGxoroCg&q=graphicspath+c%23&oq=graphicspath+c%23&gs_l=psy-ab.3..0i203l2j0i22i30l4j0i22i10i30j0i22i30l2j0i22i10i30.1170.1786..1958...0.0..0.98.251.3......0....1..gws-wiz.......0i71j0.bWRihF9_oS4>

Queue c#;

<https://docs.microsoft.com/pl-pl/dotnet/api/system.collections.queue?view=netframework-4.8>

Linq how to reverse a list:

<https://stackoverflow.com/questions/6980608/c-sharp-trying-to-reverse-a-list>

Diagram gantta online:

<https://www.canva.com/pl_pl/wykresy/gantta-zarz%C4%85dzanie-projektami/>

Use case diagram:

<https://www.google.com/search?q=use+case+diagram&tbm=isch&source=iu&ictx=1&fir=jd55rBSxbvWPNM%253A%252Cx2OQ-_w1VOH-TM%252C%252Fm%252F0j3dgmx&vet=1&usg=AI4_-kTANvJNbTgAbB6sBUItVJWMAWkFKg&sa=X&ved=2ahUKEwjD38qj3qXjAhWitYsKHb2zDvQQ_B0wHnoECAUQAw#imgrc=Gt7fAaJv2Me-dM:&vet=1>

Branch and bound algorytm:

<https://www.ii.uni.wroc.pl/~prz/2011lato/ah/opracowania/met_podz_ogr.opr.pdf>

<http://www.cs.put.poznan.pl/mkasprzak/zp/ZP-wyklad5.pdf>

<https://en.wikipedia.org/wiki/Branch_and_bound>

Wymagania funkcjonalne i niefunkcjonalne:

<http://www.ujk.edu.pl/ifiz/pl/files/lectures/Inzynieria_oprogramowania/UJK-IO-FazaOkrWym.pdf>

Hashset c# return default if not found:

<https://docs.microsoft.com/pl-pl/dotnet/api/system.collections.generic.hashset-1.trygetvalue?view=netframework-4.8#System_Collections_Generic_HashSet_1_TryGetValue__0__0__>

C# infinity:

<http://www.blackwasp.co.uk/CSharpInfinity.aspx>

Operator overloading c#:

<https://www.tutorialspoint.com/csharp/csharp_operator_overloading>

A\*:

[https://pl.wikipedia.org/wiki/Algorytm\_A\*](https://pl.wikipedia.org/wiki/Algorytm_A*)

[https://en.wikipedia.org/wiki/A\*\_search\_algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm)

S