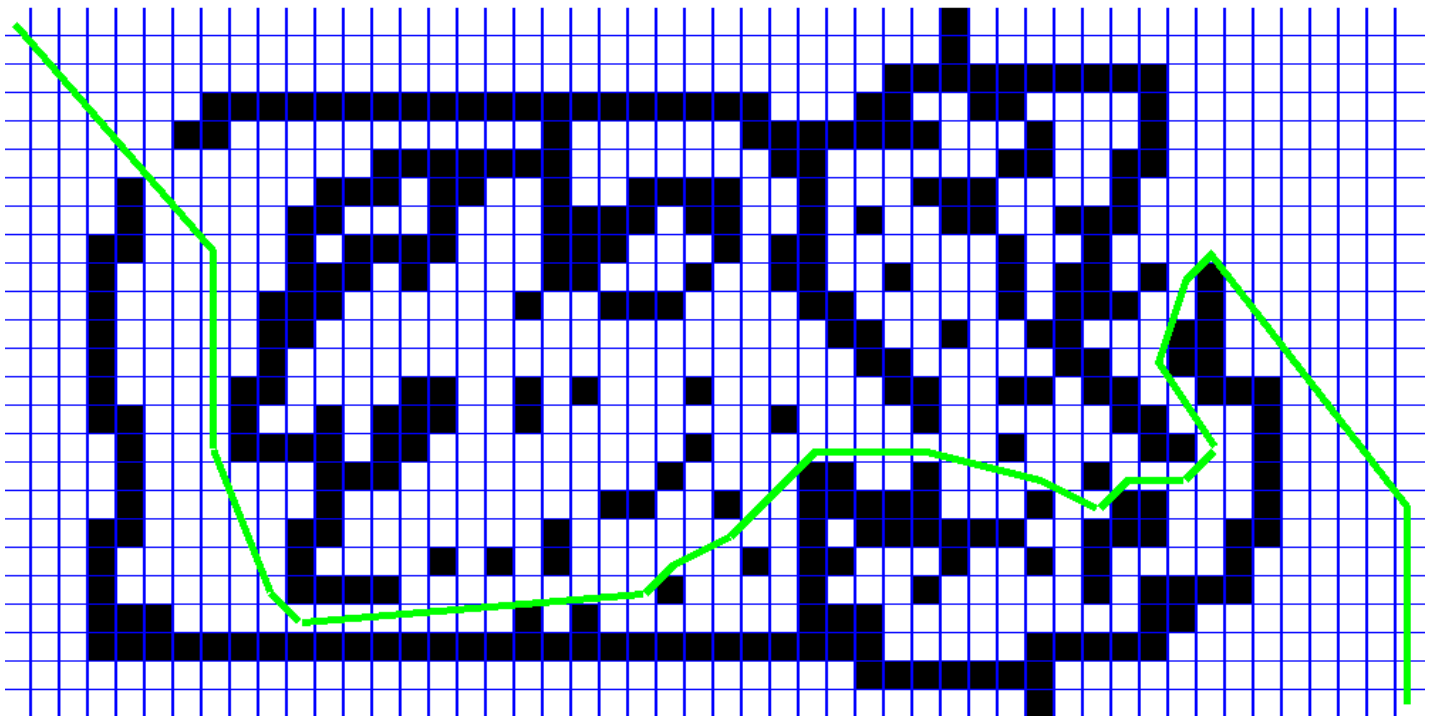


PRACA PRZEJŚCIOWA INŻYNIERSKA

PORÓWNANIE ALGORYTMÓW A* ORAZ THETA*



Damian Wysockiński, 286699

Prowadzący: dr inż. Andrzej Chmielniak

Spis treści

Wprowadzenie	3
A*	3
Theta*	3
Narzędzie programistyczne	4
Schematy blokowe	5
A*	5
Theta*	6
Porównanie długości wyznaczonych ścieżek.....	7
Konfiguracja #0.....	7
Konfiguracja #1.....	9
Konfiguracja #2.....	11
Konfiguracja #3.....	13
Konfiguracja #5.....	15
Konfiguracja #6.....	17
Zestawienie przybliżonych wyników	19
Wnioski	20

Wprowadzenie

A*

Algorytm wyznaczający najkrótszą ścieżkę między 2 węzłami w grafie pod warunkiem, że taka ścieżka istnieje. Kształt ścieżki jest zależny od wybranej heurystyki.

Algorytm realizuje minimalizację funkcji:

$f_{cost}(x) = g_{cost}(x) + h_{cost}(x)$, gdzie g_{cost} oznacza odległość po przebytej ścieżce od węzła początkowego do danego miejsca – jest zmienna, a h_{cost} oznacza niezmienną odległość obliczoną przez heurystykę między danym węzłem a węzłem będącego celem.

W swojej symulacji wykorzystałem 2 heurystyki: euklidesową i manhattańską.

Ścieżka powstaje w następujący sposób: każdy sprawdzony węzeł przez A* ma informacje o poprzednim węźle, z którego algorytm dotarł do niego. I na tej podstawie można odtworzyć najlepszą ścieżkę od węzła będącego celem do węzła będącego początkiem. Warto tu zaznaczyć, że w A* poprzednikiem węzła może być tylko węzeł leżący w sąsiedztwie danego węzła.

Theta*

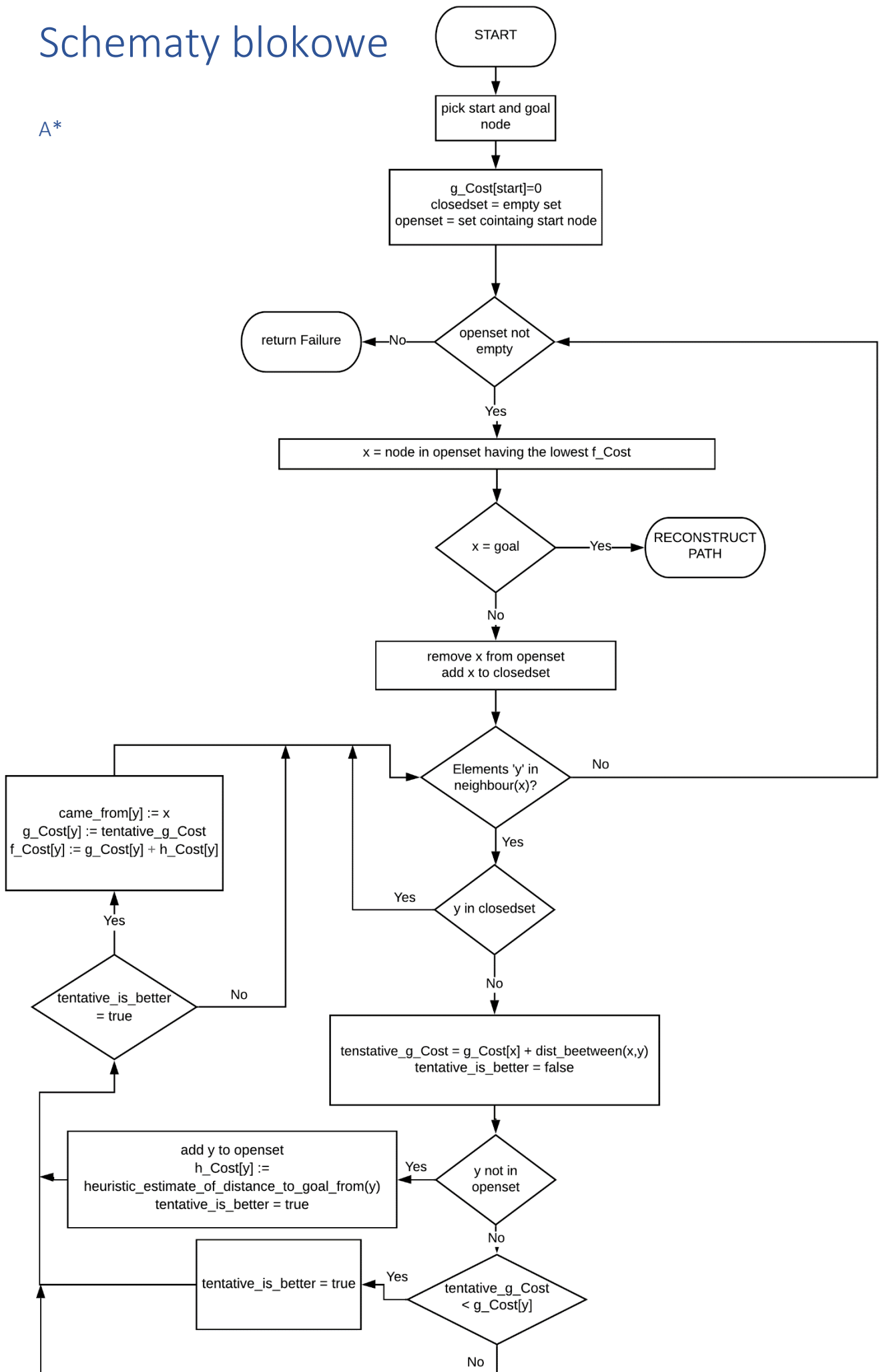
Theta* jest algorytmem bardzo zbliżonym do A*. Jedyna różnica polega na tym, że poprzednikiem danego węzła może być węzeł niebędący w najbliższym sąsiedztwie danego węzła. Dzieje się tak jeśli korzystniejsza jest droga od poprzednika poprzednika do danego węzła, z pominięciem poprzednika, niż droga przez każdy z tych węzłów po kolei. Przy analizowaniu Theta* na równomiernej siatce 2D można zauważyć jest to algorytm poruszający się po mapie pod dowolnym kątem. A* natomiast porusza się tylko w kierunkach będącymi wielokrotnościami kąta 45°.

Narzędzie programistyczne

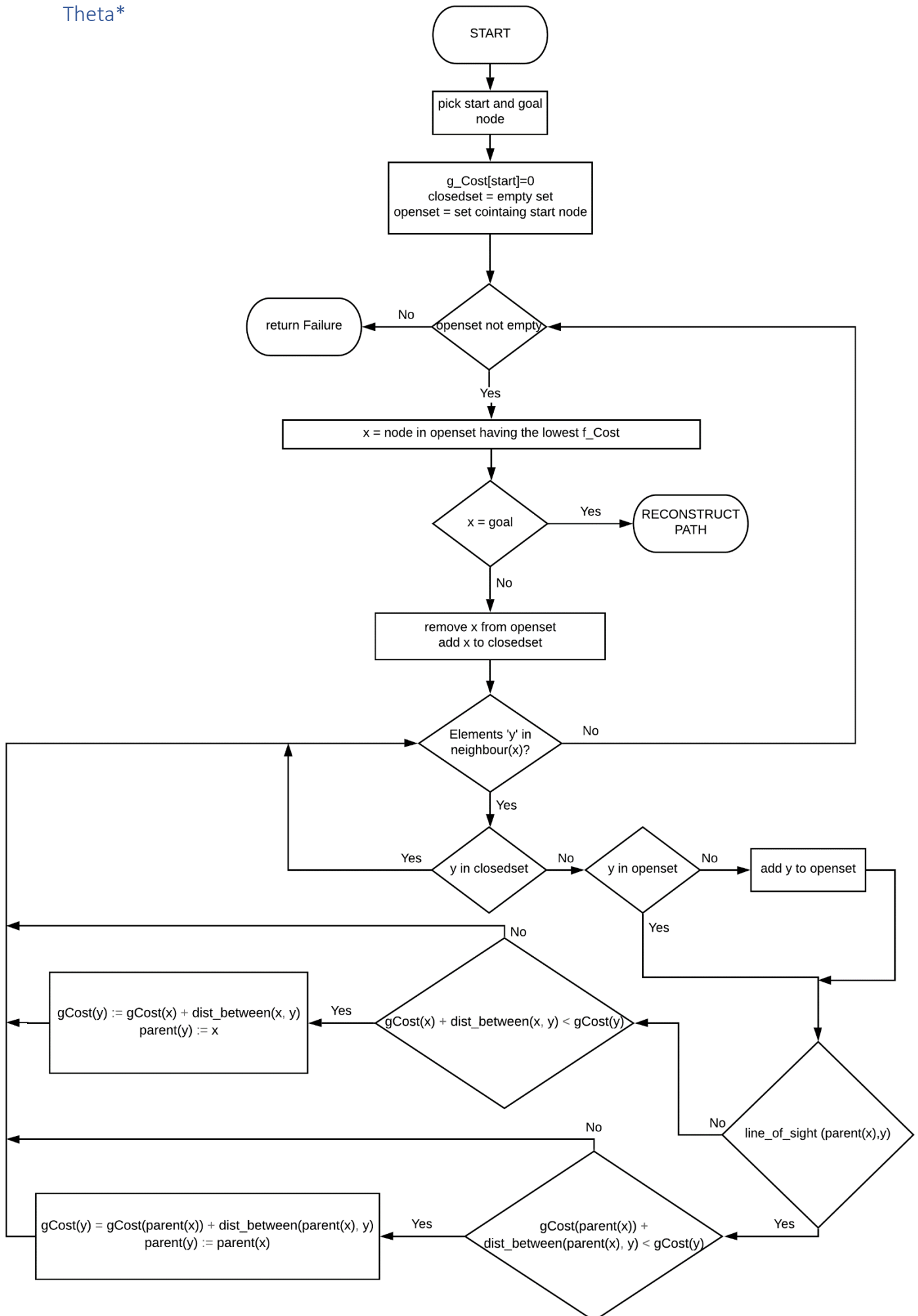
Projekt napisałem w języku C++ w środowisku Visual Studio. Do pokazania zachowania się algorytmu wykorzystałem bibliotekę SFML (Simple and Fast Multimedia Library). [Link do repozytorium](#)

Schematy blokowe

A*

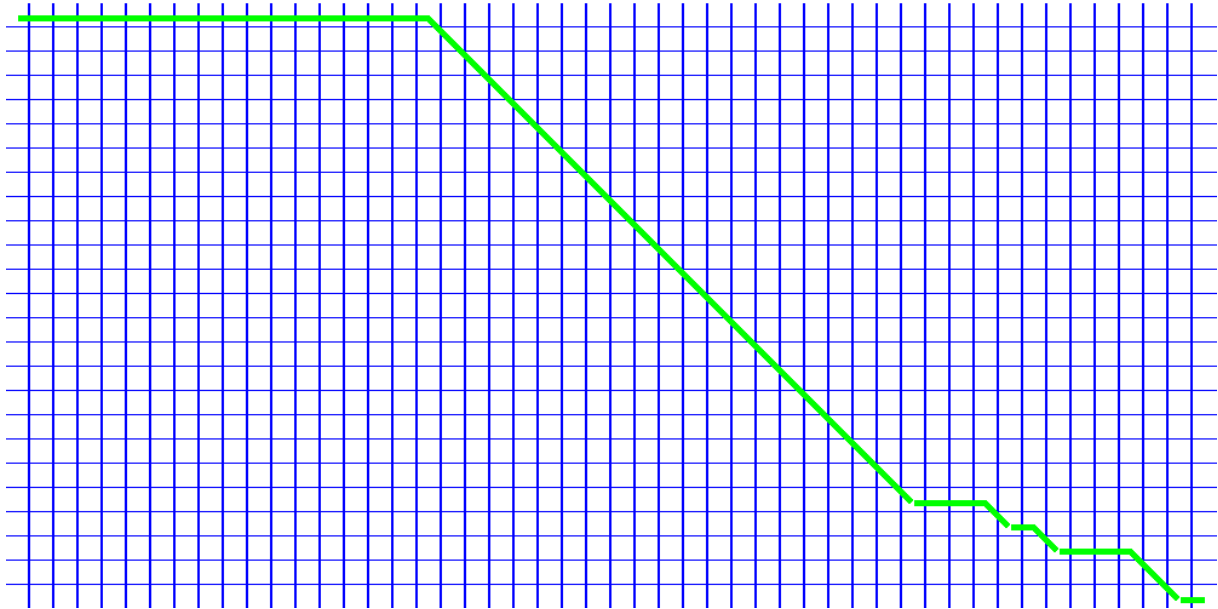


Theta*

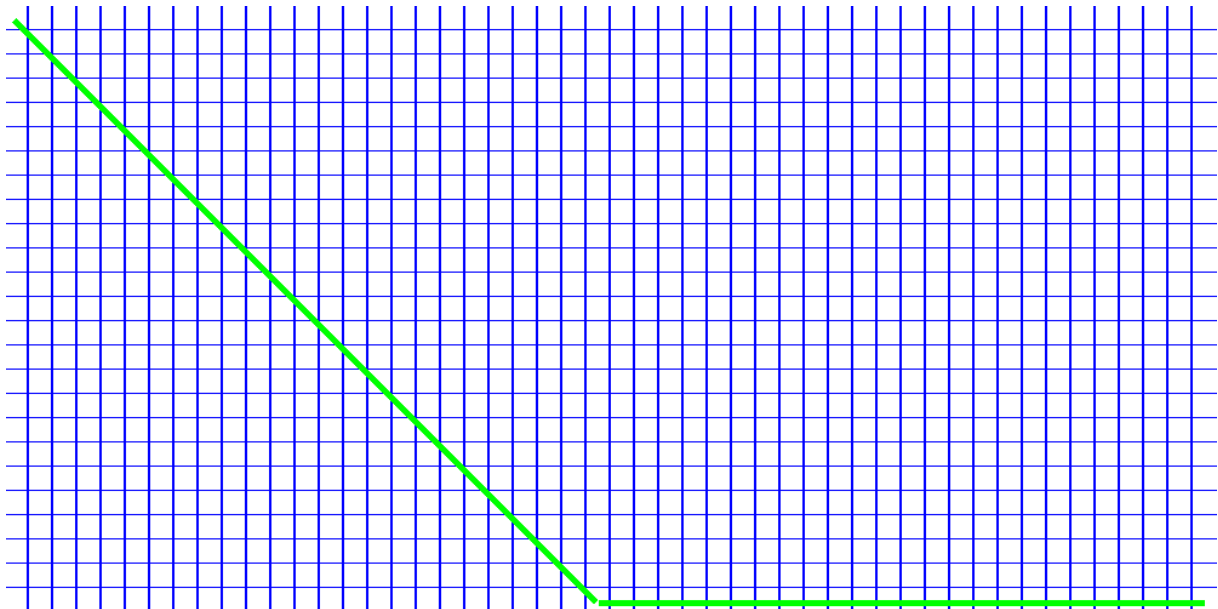


Porównanie długości wyznaczonych ścieżek

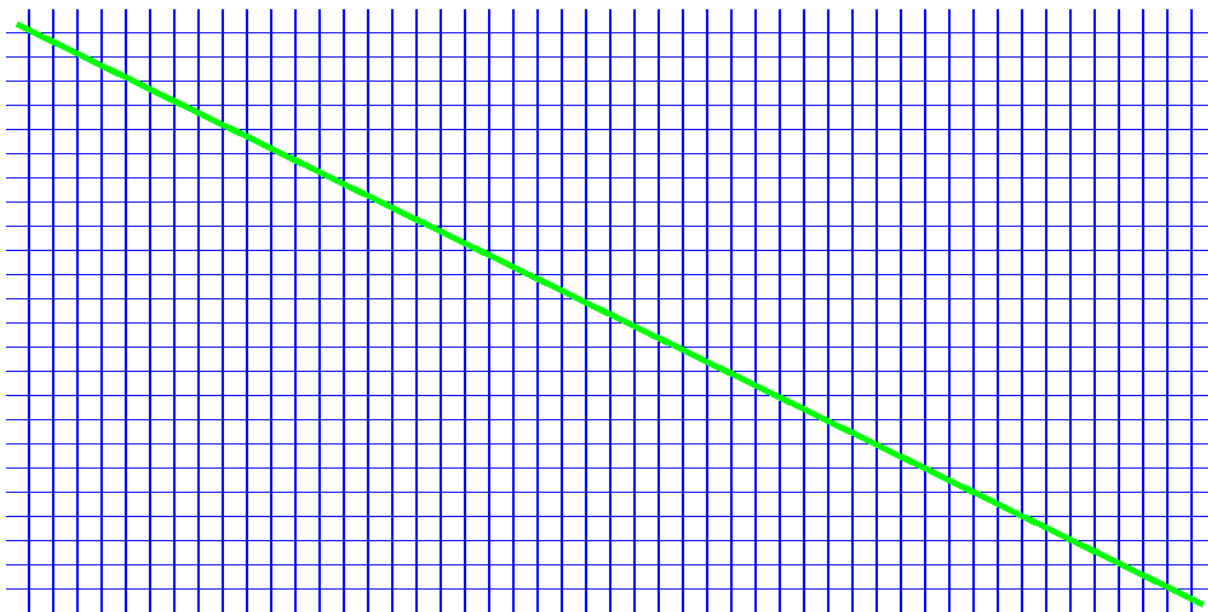
Konfiguracja #0



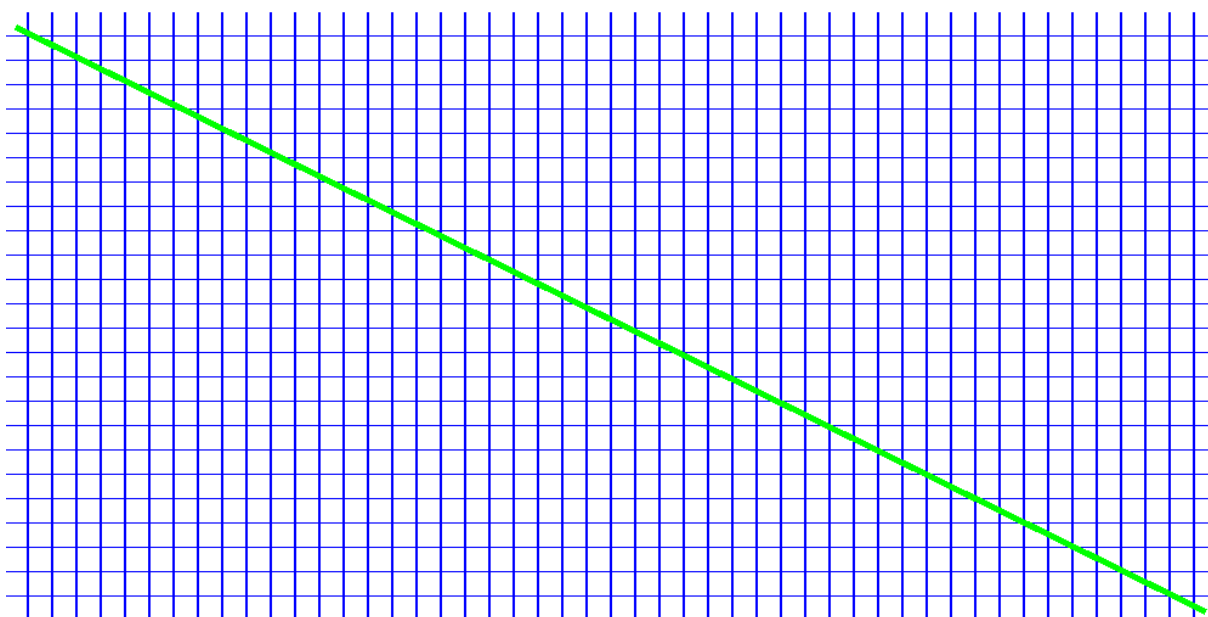
A h. euklidesowa, dł. ścieżki 58.9412*



A h. manhattańska, dł. ścieżki 58.9412*

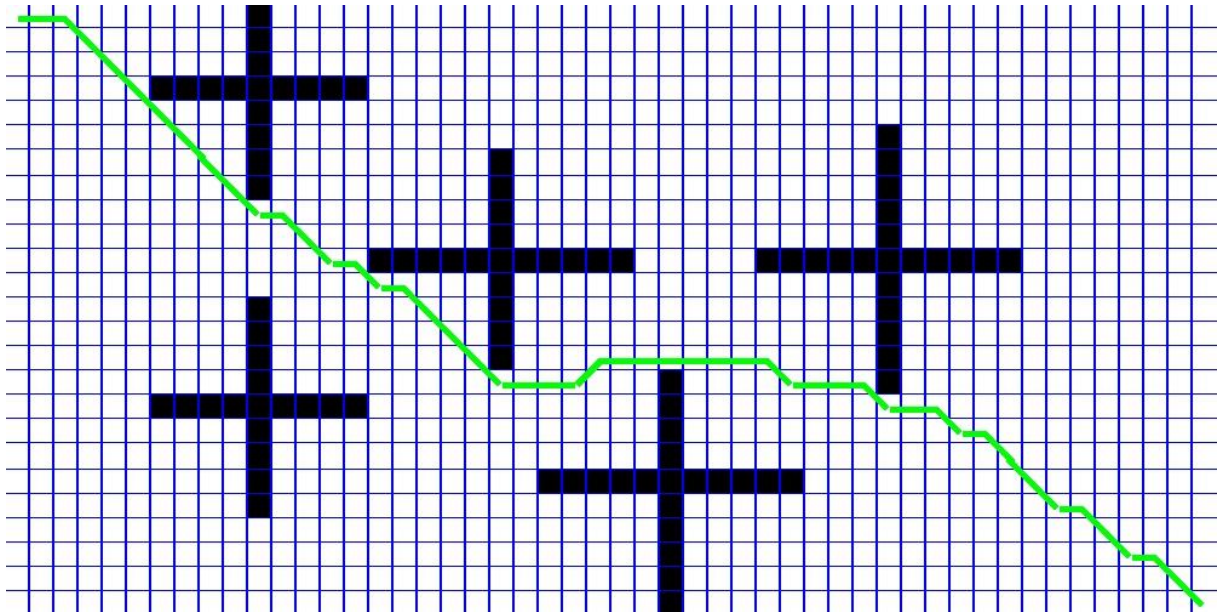


Theta h. euklidesowa, dł. ścieżki 54.5619*

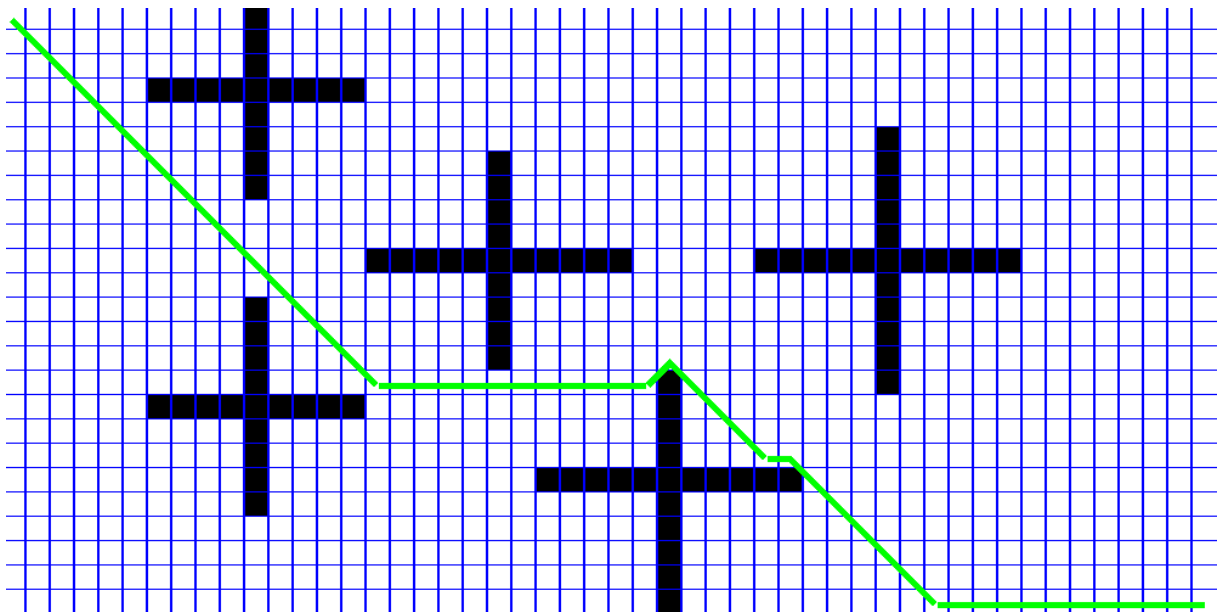


Theta h. manhattańska, dł. ścieżki 54.5619*

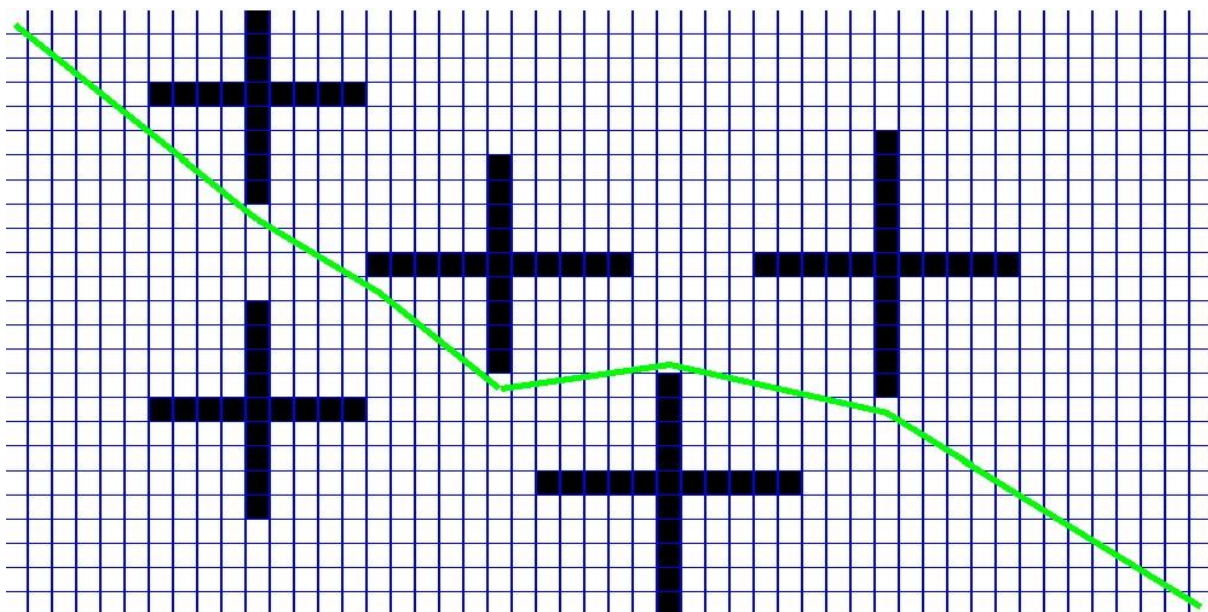
Konfiguracja #1



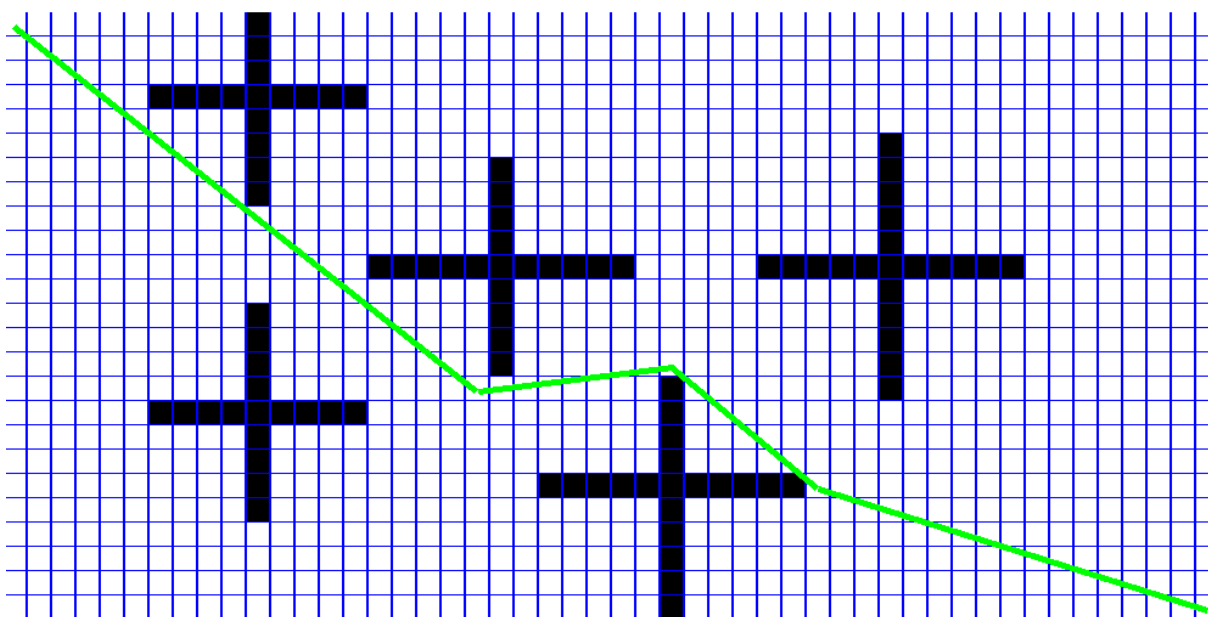
A h. euklidesowa, dł. ścieżki 59.7696*



A h. manhattańska, dł. ścieżki 59.7696*

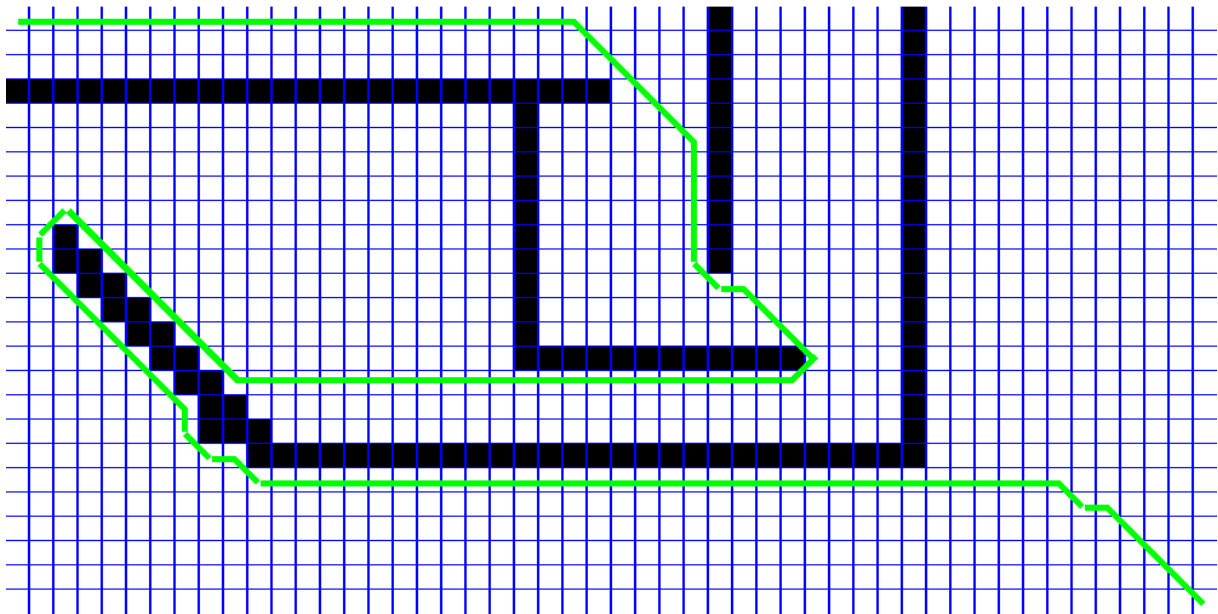


Theta h. euklidesowa, dł. ścieżki 56.5953*

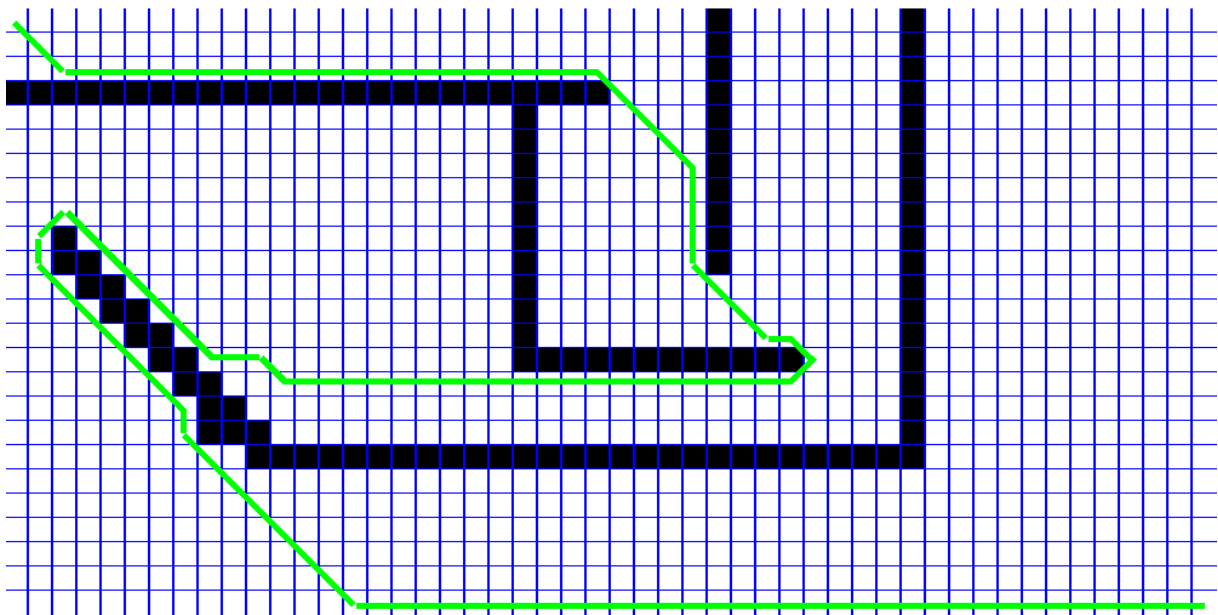


Theta h. manhattańska, dł. ścieżki 56.843*

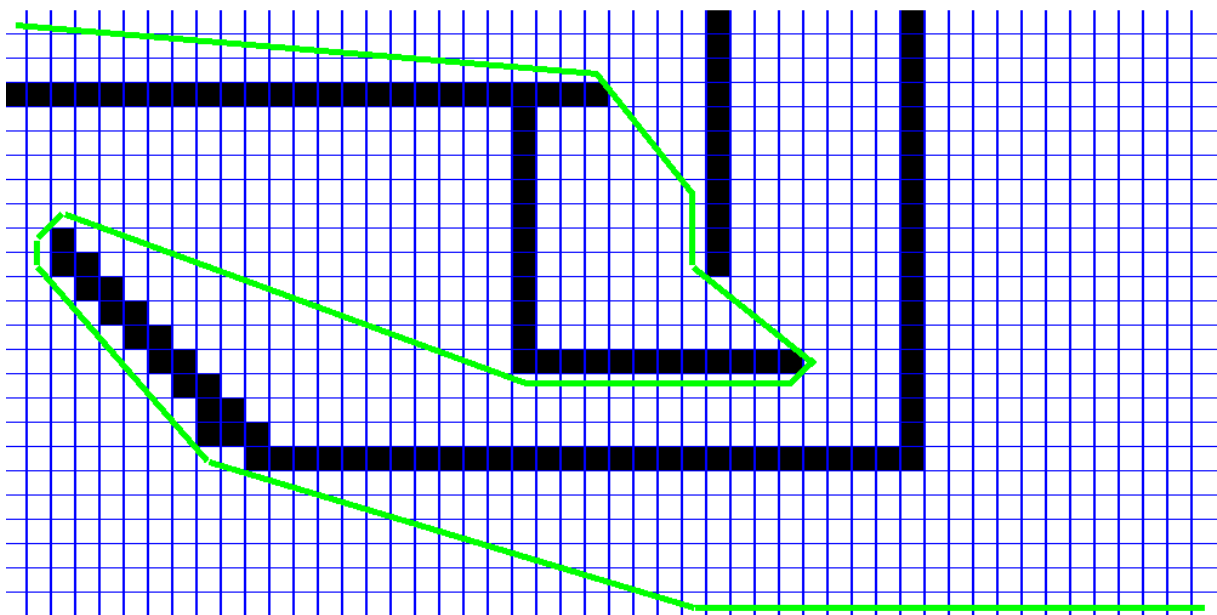
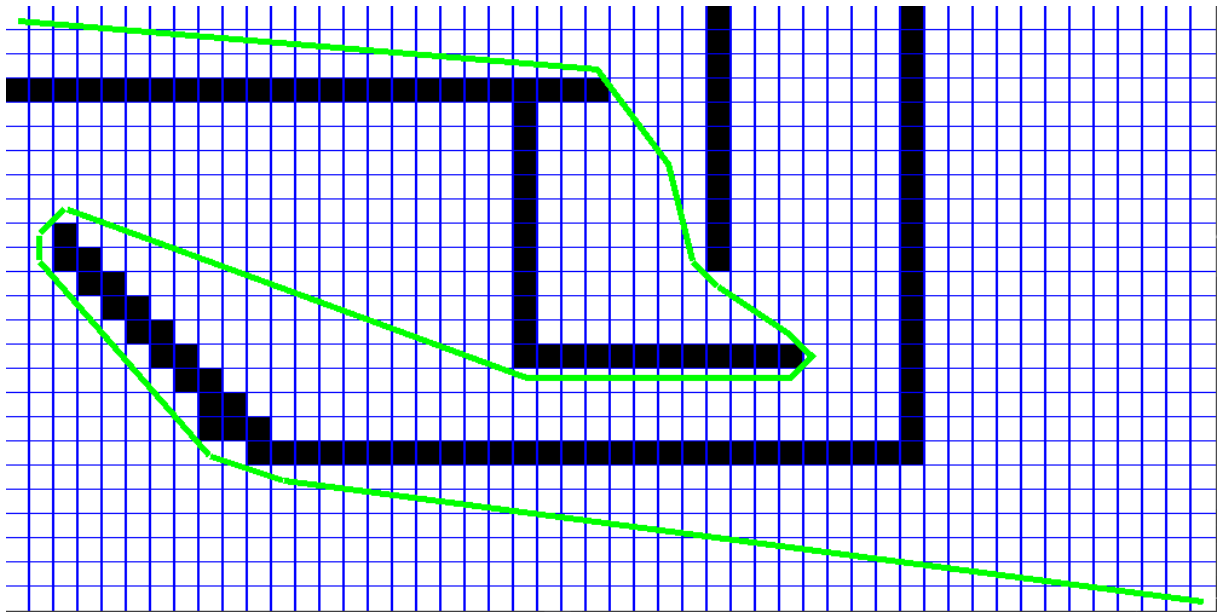
Konfiguracja #2



A h. euklidesowa, dł. ścieżki 132.841*

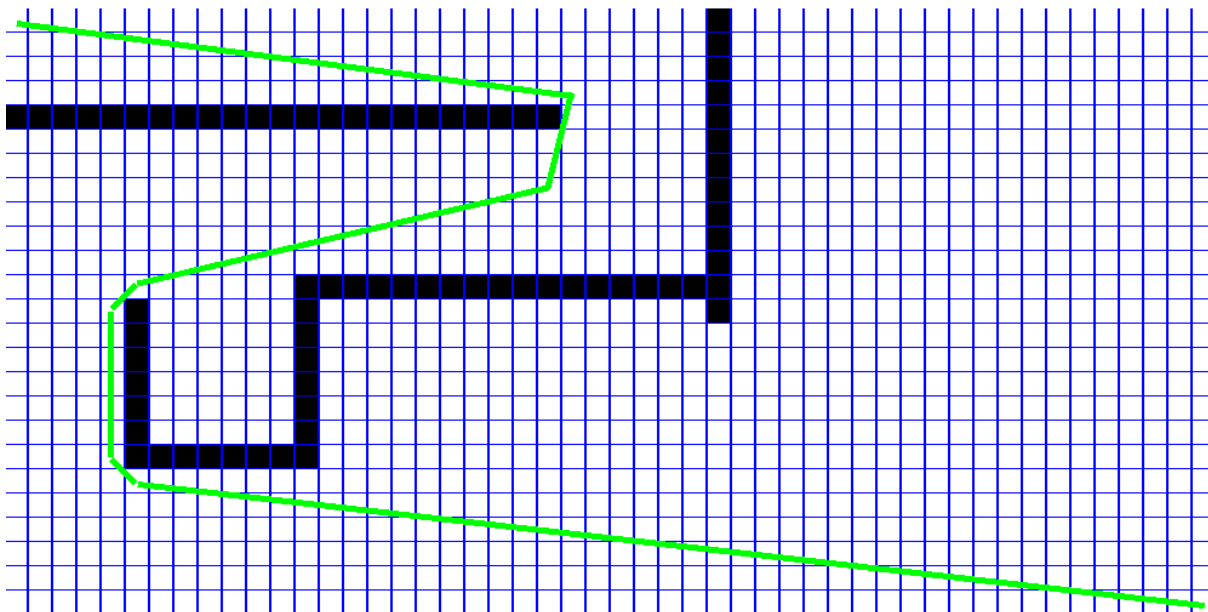


A h. manhattańska, dł. ścieżki 132.255*

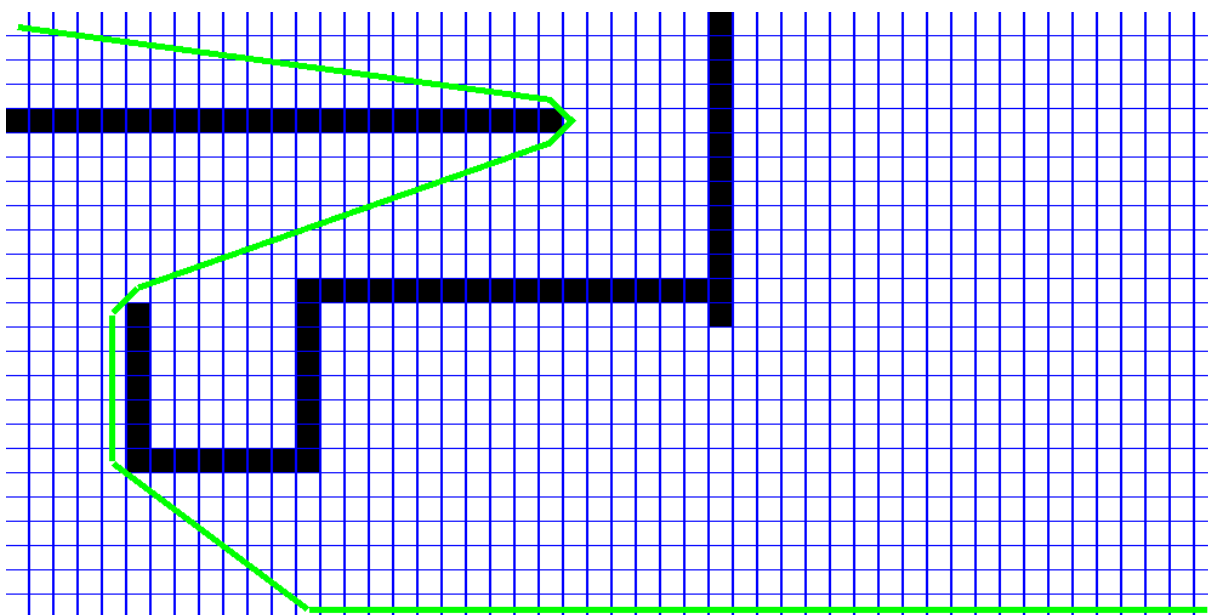


[illegible][illegible]

A* h. manhattańska, dł. ścieżki 100.456



Theta h. euklidesowa, dł. ścieżki 97.8938*

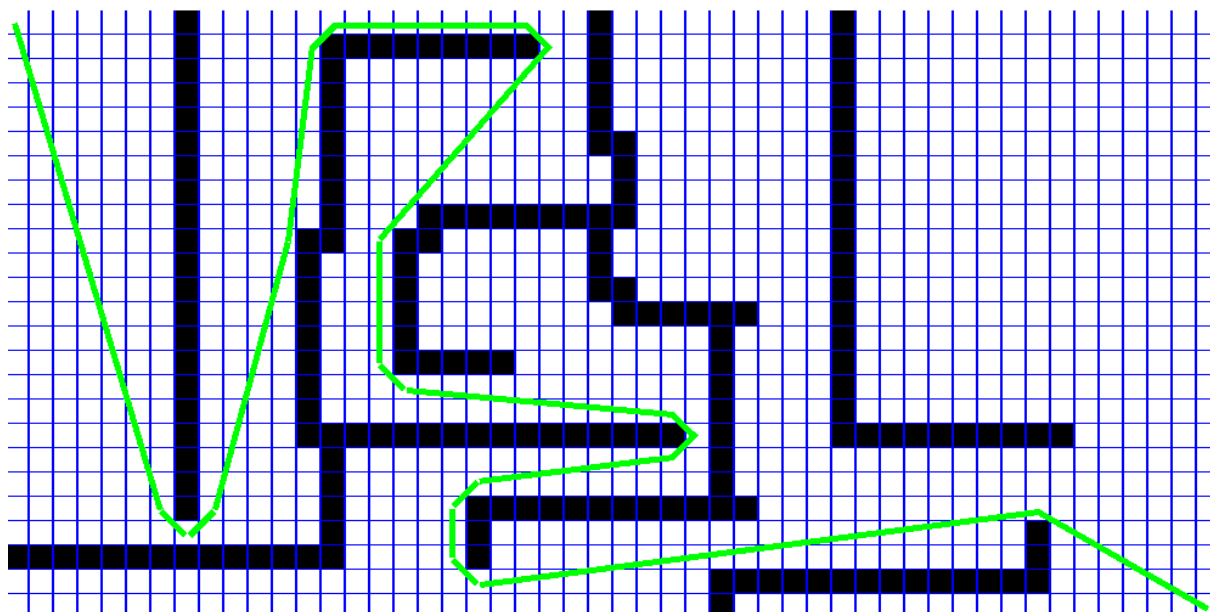


Theta h. manhattańska, dł. ścieżki 97.474*

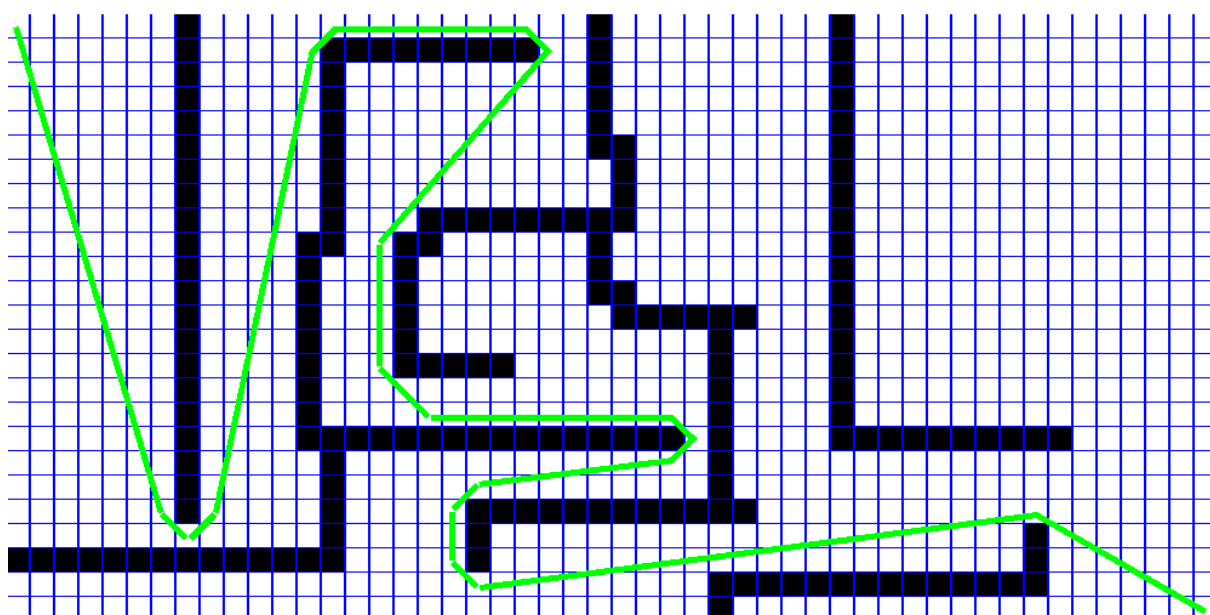
A 2D grid world environment. The grid is composed of blue lines on a white background. Black squares represent obstacles or walls. A green line represents a path starting from the top-left corner, moving right, then down, then right again, and finally down to the bottom-right corner, avoiding the obstacles.

A 2D grid world environment. The grid is 20x20 units. A green path starts at (0, 19) and ends at (19, 0). Black obstacles are located at (1, 19), (2, 19), (3, 19), (4, 19), (5, 19), (6, 19), (7, 19), (8, 19), (9, 19), (10, 19), (11, 19), (12, 19), (13, 19), (14, 19), (15, 19), (16, 19), (17, 19), (18, 19), (19, 19), (0, 18), (1, 18), (2, 18), (3, 18), (4, 18), (5, 18), (6, 18), (7, 18), (8, 18), (9, 18), (10, 18), (11, 18), (12, 18), (13, 18), (14, 18), (15, 18), (16, 18), (17, 18), (18, 18), (19, 18), (0, 17), (1, 17), (2, 17), (3, 17), (4, 17), (5, 17), (6, 17), (7, 17), (8, 17), (9, 17), (10, 17), (11, 17), (12, 17), (13, 17), (14, 17), (15, 17), (16, 17), (17, 17), (18, 17), (19, 17), (0, 16), (1, 16), (2, 16), (3, 16), (4, 16), (5, 16), (6, 16), (7, 16), (8, 16), (9, 16), (10, 16), (11, 16), (12, 16), (13, 16), (14, 16), (15, 16), (16, 16), (17, 16), (18, 16), (19, 16), (0, 15), (1, 15), (2, 15), (3, 15), (4, 15), (5, 15), (6, 15), (7, 15), (8, 15), (9, 15), (10, 15), (11, 15), (12, 15), (13, 15), (14, 15), (15, 15), (16, 15), (17, 15), (18, 15), (19, 15), (0, 14), (1, 14), (2, 14), (3, 14), (4, 14), (5, 14), (6, 14), (7, 14), (8, 14), (9, 14), (10, 14), (11, 14), (12, 14), (13, 14), (14, 14), (15, 14), (16, 14), (17, 14), (18, 14), (19, 14), (0, 13), (1, 13), (2, 13), (3, 13), (4, 13), (5, 13), (6, 13), (7, 13), (8, 13), (9, 13), (10, 13), (11, 13), (12, 13), (13, 13), (14, 13), (15, 13), (16, 13), (17, 13), (18, 13), (19, 13), (0, 12), (1, 12), (2, 12), (3, 12), (4, 12), (5, 12), (6, 12), (7, 12), (8, 12), (9, 12), (10, 12), (11, 12), (12, 12), (13, 12), (14, 12), (15, 12), (16, 12), (17, 12), (18, 12), (19, 12), (0, 11), (1, 11), (2, 11), (3, 11), (4, 11), (5, 11), (6, 11), (7, 11), (8, 11), (9, 11), (10, 11), (11, 11), (12, 11), (13, 11), (14, 11), (15, 11), (16, 11), (17, 11), (18, 11), (19, 11), (0, 10), (1, 10), (2, 10), (3, 10), (4, 10), (5, 10), (6, 10), (7, 10), (8, 10), (9, 10), (10, 10), (11, 10), (12, 10), (13, 10), (14, 10), (15, 10), (16, 10), (17, 10), (18, 10), (19, 10), (0, 9), (1, 9), (2, 9), (3, 9), (4, 9), (5, 9), (6, 9), (7, 9), (8, 9), (9, 9), (10, 9), (11, 9), (12, 9), (13, 9), (14, 9), (15, 9), (16, 9), (17, 9), (18, 9), (19, 9), (0, 8), (1, 8), (2, 8), (3, 8), (4, 8), (5, 8), (6, 8), (7, 8), (8, 8), (9, 8), (10, 8), (11, 8), (12, 8), (13, 8), (14, 8), (15, 8), (16, 8), (17, 8), (18, 8), (19, 8), (0, 7), (1, 7), (2, 7), (3, 7), (4, 7), (5, 7), (6, 7), (7, 7), (8, 7), (9, 7), (10, 7), (11, 7), (12, 7), (13, 7), (14, 7), (15, 7), (16, 7), (17, 7), (18, 7), (19, 7), (0, 6), (1, 6), (2, 6), (3, 6), (4, 6), (5, 6), (6, 6), (7, 6), (8, 6), (9, 6), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6), (16, 6), (17, 6), (18, 6), (19, 6), (0, 5), (1, 5), (2, 5), (3, 5), (4, 5), (5, 5), (6, 5), (7, 5), (8, 5), (9, 5), (10, 5), (11, 5), (12, 5), (13, 5), (14, 5), (15, 5), (16, 5), (17, 5), (18, 5), (19, 5), (0, 4), (1, 4), (2, 4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 4), (11, 4), (12, 4), (13, 4), (14, 4), (15, 4), (16, 4), (17, 4), (18, 4), (19, 4), (0, 3), (1, 3), (2, 3), (3, 3), (4, 3), (5, 3), (6, 3), (7, 3), (8, 3), (9, 3), (10, 3), (11, 3), (12, 3), (13, 3), (14, 3), (15, 3), (16, 3), (17, 3), (18, 3), (19, 3), (0, 2), (1, 2), (2, 2), (3, 2), (4, 2), (5, 2), (6, 2), (7, 2), (8, 2), (9, 2), (10, 2), (11, 2), (12, 2), (13, 2), (14, 2), (15, 2), (16, 2), (17, 2), (18, 2), (19, 2), (0, 1), (1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (9, 1), (10, 1), (11, 1), (12, 1), (13, 1), (14, 1), (15, 1), (16, 1), (17, 1), (18, 1), (19, 1), (0, 0), (1, 0), (2, 0), (3, 0), (4, 0), (5, 0), (6, 0), (7, 0), (8, 0), (9, 0), (10, 0), (11, 0), (12, 0), (13, 0), (14, 0), (15, 0), (16, 0), (17, 0), (18, 0), (19, 0).

A* h. manhattańska, dł. ścieżki 134.498

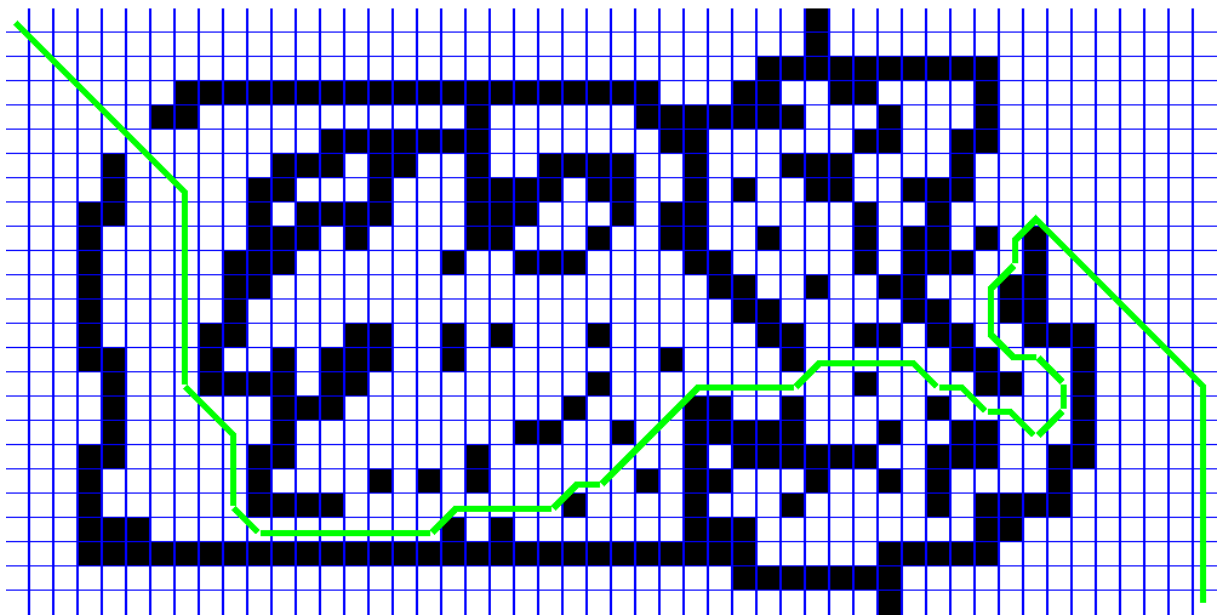


Theta h. euklidesowa, dł. ścieżki 129.067*

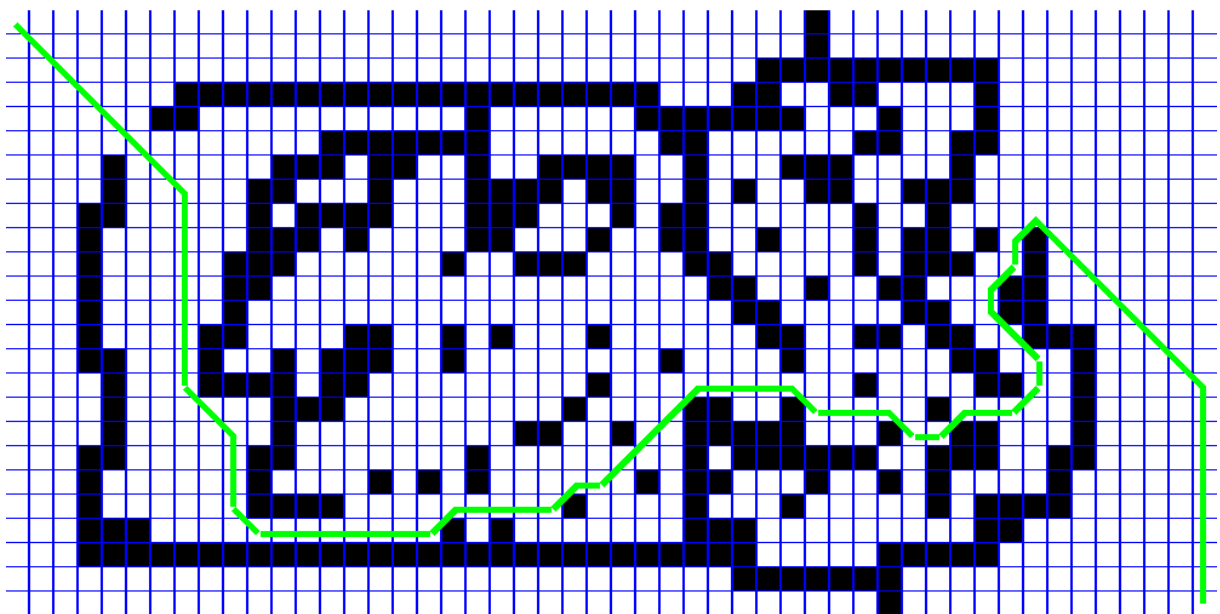


Theta h. manhattańska, dł. ścieżki 129.389*

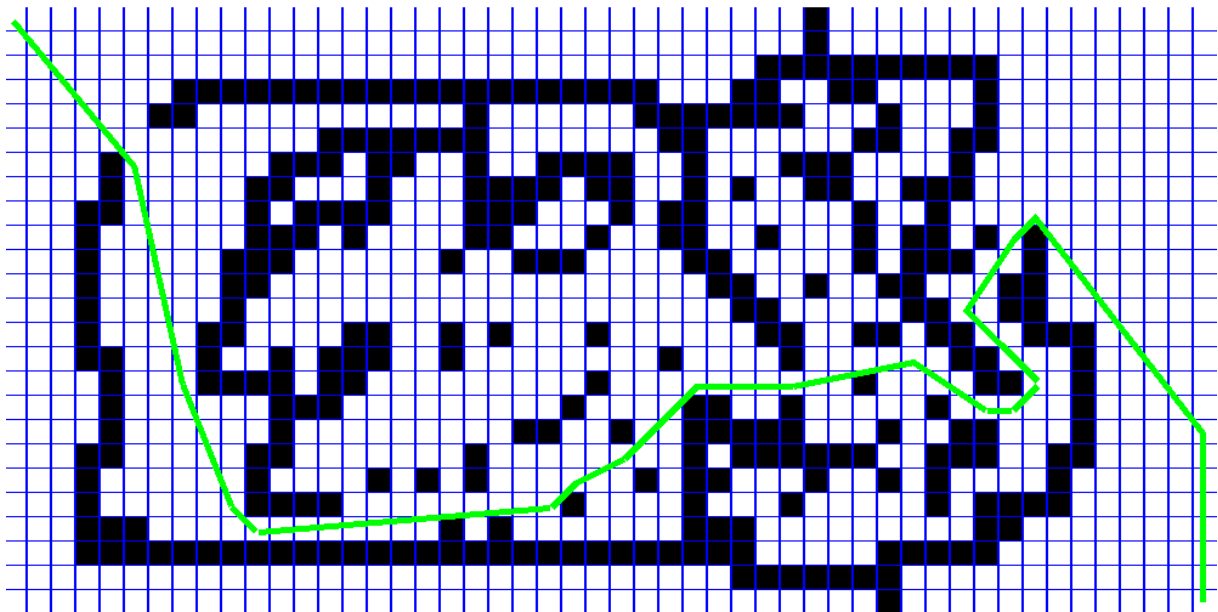
Konfiguracja #6



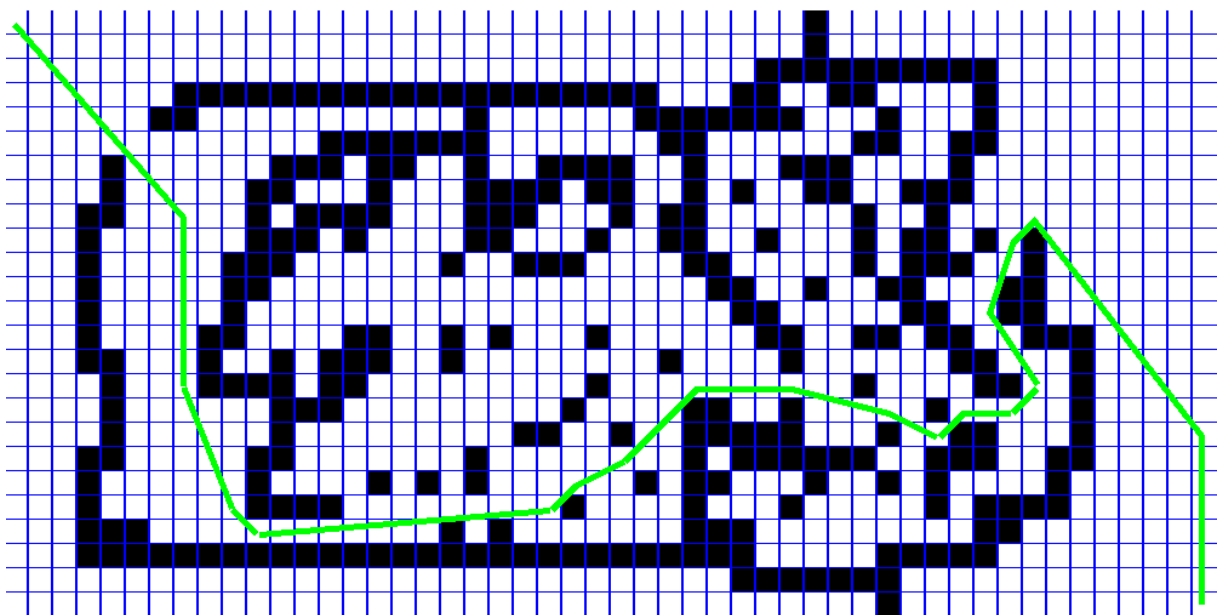
A h. euklidesowa, dł. ścieżki 92.2549*



A h. manhattańska, dł. ścieżki 88.8406*



Theta h. euklidesowa, dł. ścieżki 86.5466*



Theta h. manhattańska, dł. ścieżki 86.1354*

Zestawienie przybliżonych wyników

(dokładne wyniki w [/SPRAWOZDANIE/dane_podsumowanie.xlsx](#))

mapa	A*						Theta*					
	h. euklidesowa			h. manhattańska			h. euklidesowa			h. manhattańska		
#0	58,94	236,3	235,8	58,94	17,7	17,2	54,56	235,6	233,8	54,56	19,8	19,5
#1	59,77	166,1	165,3	59,77	41,5	41,0	56,60	147,3	146,3	56,84	41,4	40,7
#2	132,84	109,2	108,6	132,26	62,3	59,3	126,84	116,3	114,9	127,48	68,1	66,7
#3	101,04	199,2	108,6	100,46	36,1	34,9	97,90	185,5	183,7	97,47	40,3	39,4
#5	134,50	49,2	47,4	134,50	45,7	43,3	129,07	48,4	47,7	129,39	48,7	48,1
#6	92,25	101,0	98,2	88,84	87,3	86,1	86,55	108,0	105,4	86,14	92,8	91,6
	dł. ścieżki	czas (ms)	czas (ms prio.)	dł. ścieżki	czas (ms)	czas (ms prio.)	dł. ścieżki	czas (ms)	czas (ms prio.)	dł. ścieżki	czas (ms)	czas (ms prio.)

- Dla A* wykorzystanie heurystyki manhattańskiej zamiast euklidesowej przyspiesza algorytm około 1.7 razy
- Dla Theta* wykorzystanie heurystyki manhattańskiej zamiast euklidesowej przyspiesza algorytm około 1.4 razy
- Dla A* heurystyka manhattańska wyznacza krótsze ścieżki o średnio 0.8%
- W przypadku Theta* wybór heurystyki ma pomijalny wpływ (0.045%) na długość ścieżki
- Theta* wyznacza znacznie krótsze ścieżki od A*: średnio 5,4% dla heurystyki euklidesowej i 4.5% dla manhattańskiej
- Ustawienie najwyższego priorytetu na aplikację i zabicie nieistotnych procesów przyspiesza algorytm średnio o 1,76%

Wnioski

1. Theta* wyznacza wyraźnie krótsze ścieżki w porównaniu do A*
2. Dla A* minimalnie krótsze ścieżki daje wykorzystywanie heurystyki manhattańskiej
3. Dla Theta* wybór heurystyki nie ma większego wpływu na długość ścieżki
4. Dla obu algorytmów heurystyka manhattańska wyraźnie skraca czas obliczeń
5. Ustawienie najwyższego priorytetu dla aplikacji oraz zabijanie niepotrzebnych procesów daje niewielki wpływ na szybkość obliczeń
6. Do zadań, gdzie ważna jest szybkość obliczeń, a nie dokładność, wybrałbym A* z heurystyką manhattańską
7. Do zadań, gdzie istotna jest precyzja i szybkość, wybrałbym Theta* z heurystyką manhattańską

Kod źródłowy:

- https://github.com/damianski794/Przejsciewka_A_Star

Bibliografia:

- <https://arxiv.org/ftp/arxiv/papers/1401/1401.3843.pdf>
- https://pl.wikipedia.org/wiki/Algorytm_A*
- https://en.wikipedia.org/wiki/Theta*
- <https://gamedev.stackexchange.com/questions/75158/line-of-sight-on-a-2d-grid>