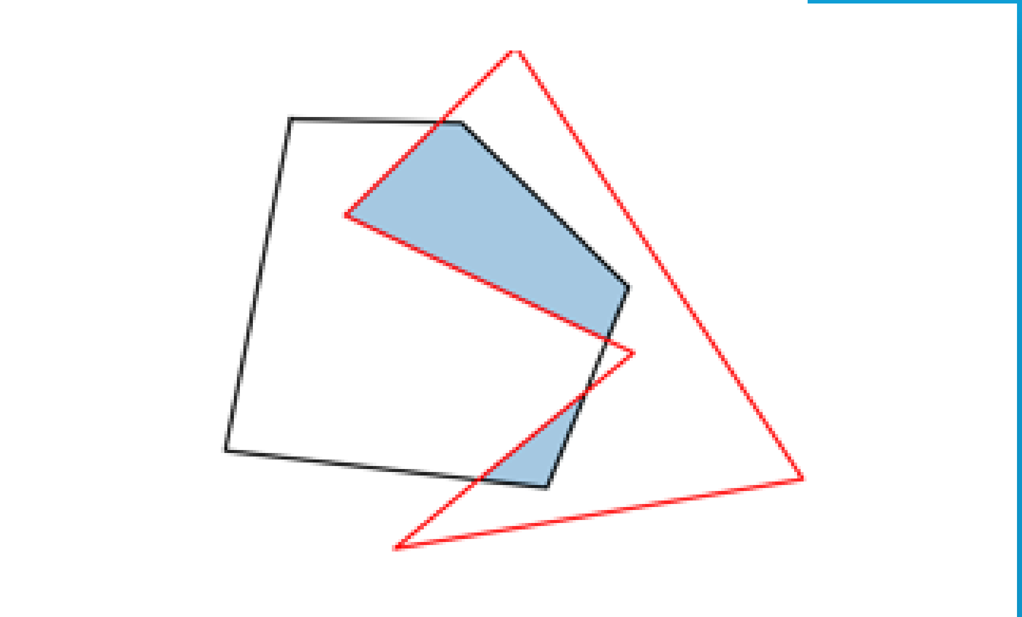
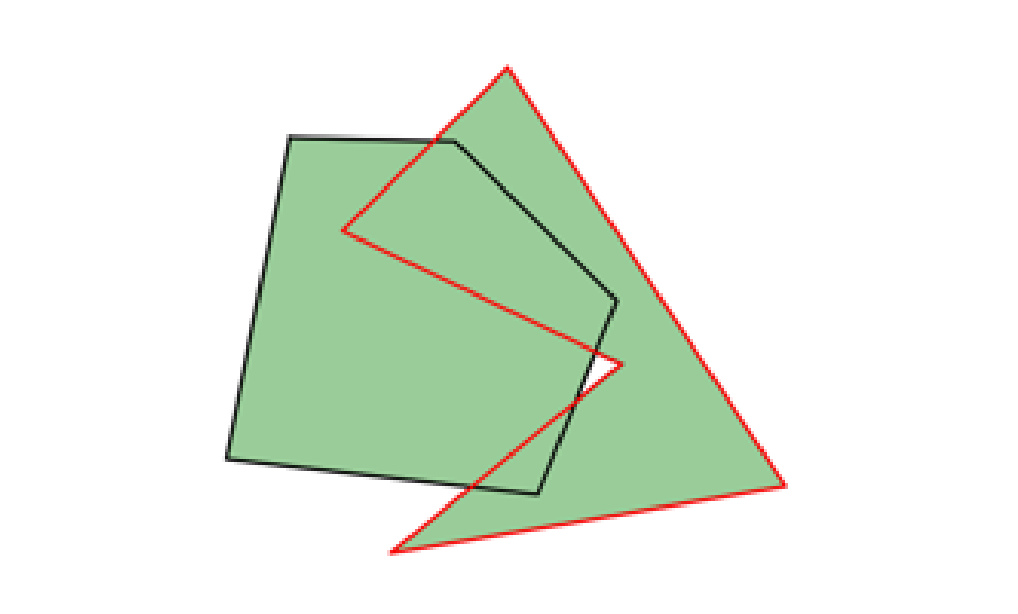
Algorytm Greinera-Hormanna

Obliczanie sumy i iloczynu dwóch wielokątów

Błażej Żuk

Iloczyn i suma dwóch wielokątów





- Zadanie polega na wyznaczeniu sumy oraz iloczynu dwóch wielokątów.
- Iloczynem wielokątów jest zbiór wszystkich punktów należących do obu wielokątów.
- Sumą wielokątów jest zbiór wszystkich punktów należących do któregokolwiek wielokąta.

Założenia

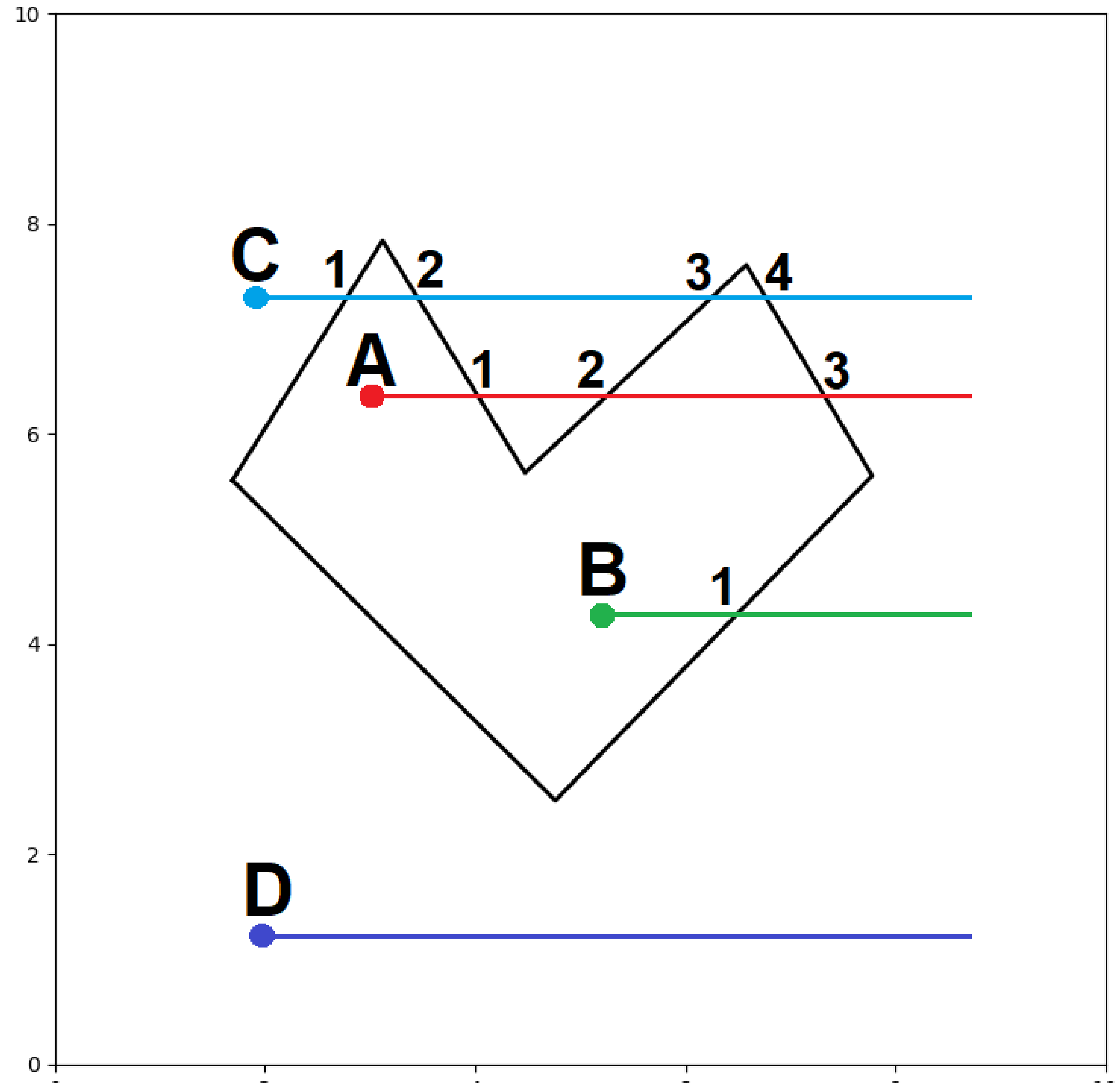
- Każdy wielokąt jest prosty (nie posiada dziur oraz nie przecina samego siebie),
- Krawędzie wielokątów nie nachodzą na siebie,
- Wierzchołek jednego wielokąta nie leży na krawędzi drugiego wielokąta,
- Żadna para wierzchołków nie posiada takiej samej współrzędnej y.

Położenie punktu względem wielokąta

Metoda „ray-casting” służy do sprawdzania, czy dany punkt leży wewnątrz wielokąta.

Od badanego punktu rzutowany jest poziomy promień w prawo (do wystarczająco dużej wartości x). Zliczana jest liczba przecięć tego promienia z krawędziami wielokąta.

Nieparzysta liczba oznacza, że punkt leży wewnątrz wielokąta, a parzysta oznacza, że leży na zewnątrz.



Algorytm Greinera-Hormanna

Algorytm składa się z trzech etapów:

1. Wyznaczenie przecięć wielokątów,
2. Oznaczenie przecięć jako „punkty wejścia” lub „punkty wyjścia”,
3. Konstrukcja iloczynu lub sumy.

Wynikiem algorytmu jest zbiór wielokątów tworzących iloczyn/sumę.

Wyznaczenie przecięć wielokątów

W celu wyznaczenia przecięć można użyć prostego algorytmu wyczerpującego – sprawdzamy każdą krawędź pierwszego wielokąta ze wszystkimi krawędziami drugiego wielokąta.

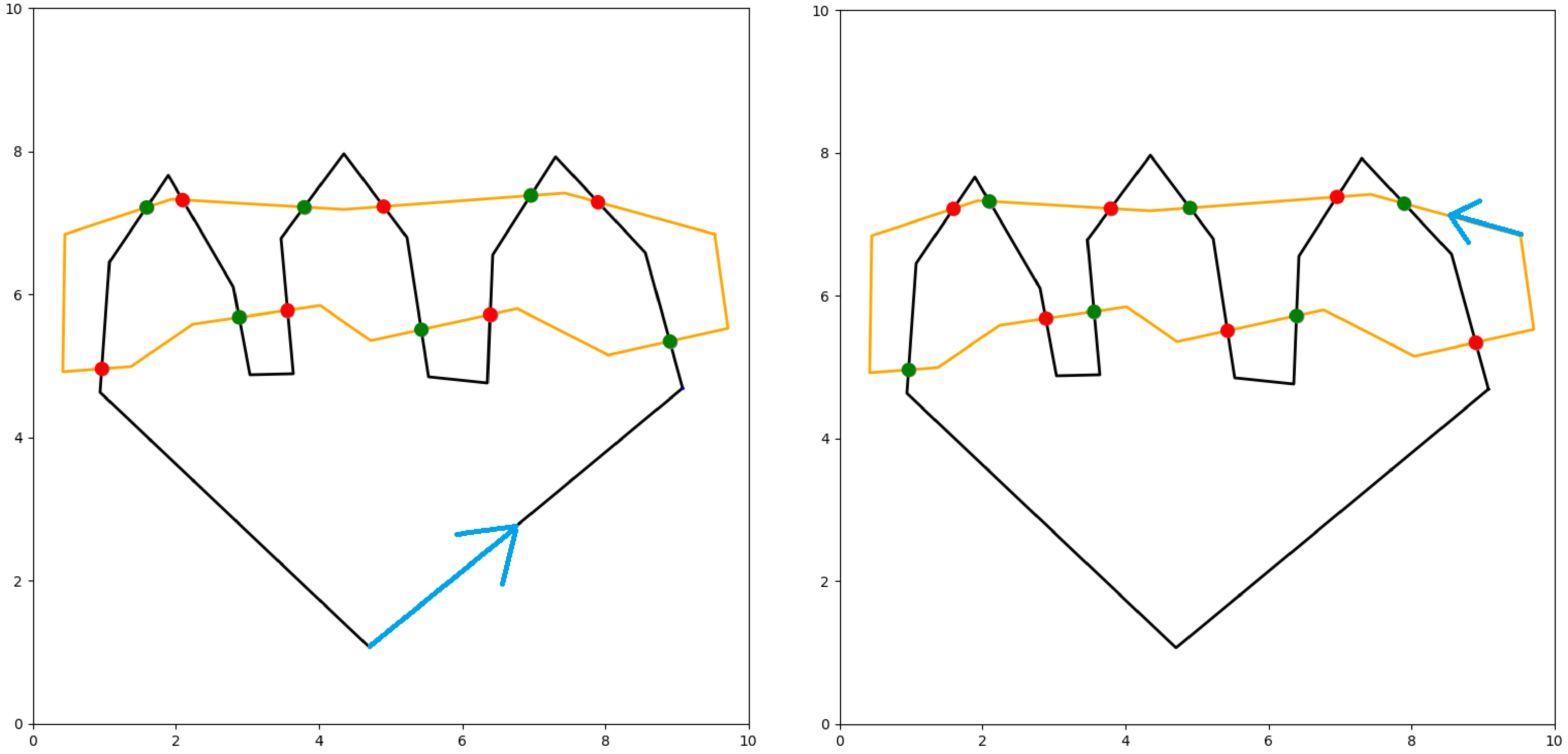
Po znalezieniu przecięć wstawiamy je w odpowiednim miejscu do struktury przechowującej wielokąt.

Oznaczenie przecięć jako „punkty wejścia” lub „punkty wyjścia”

Dla każdego punktu przecięcia w danym wielokącie określane jest, czy jest to "punkt wejścia" do drugiego wielokąta, czy "punkt wyjścia".

W tym celu przechodzimy po wszystkich wierzchołkach wielokąta - jeśli w danym momencie znajdujemy się na zewnątrz drugiego wielokąta, to wiemy, że następne przecięcie będzie wejściem.

Analogicznie, jeśli jesteśmy wewnątrz drugiego wielokąta, to następne przecięcie będzie punktem wyjścia.



0
2
4
6
8
10
0
2
4
6
8
10
0
2
4
6
8
10
0
2
4
6
8
10

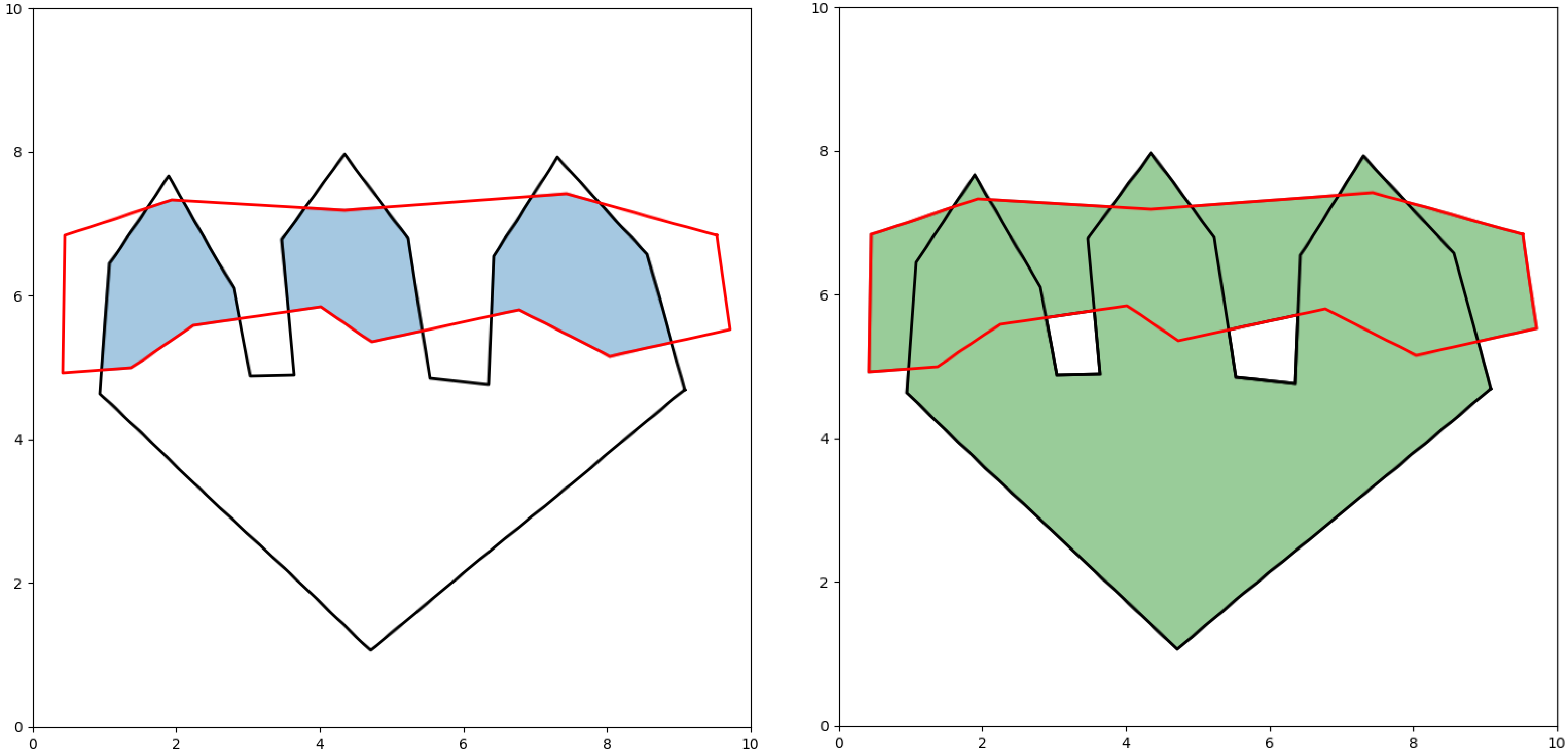
Konstrukcja iloczynu i sumy

Zaczynając od dowolnego punktu przecięcia, przechodzimy wzdłuż wielokąta zgodnie z regułą:

- dla iloczynu: przy wejściu idziemy do przodu, przy wyjściu do tyłu,
- dla sumy: przy wejściu idziemy do tyłu, przy wyjściu do przodu.

Po osiągnięciu kolejnego punktu przecięcia przechodzimy na drugi wielokąt.

Proces powtarzany jest, aż wszystkie przecięcia zostaną odwiedzone.



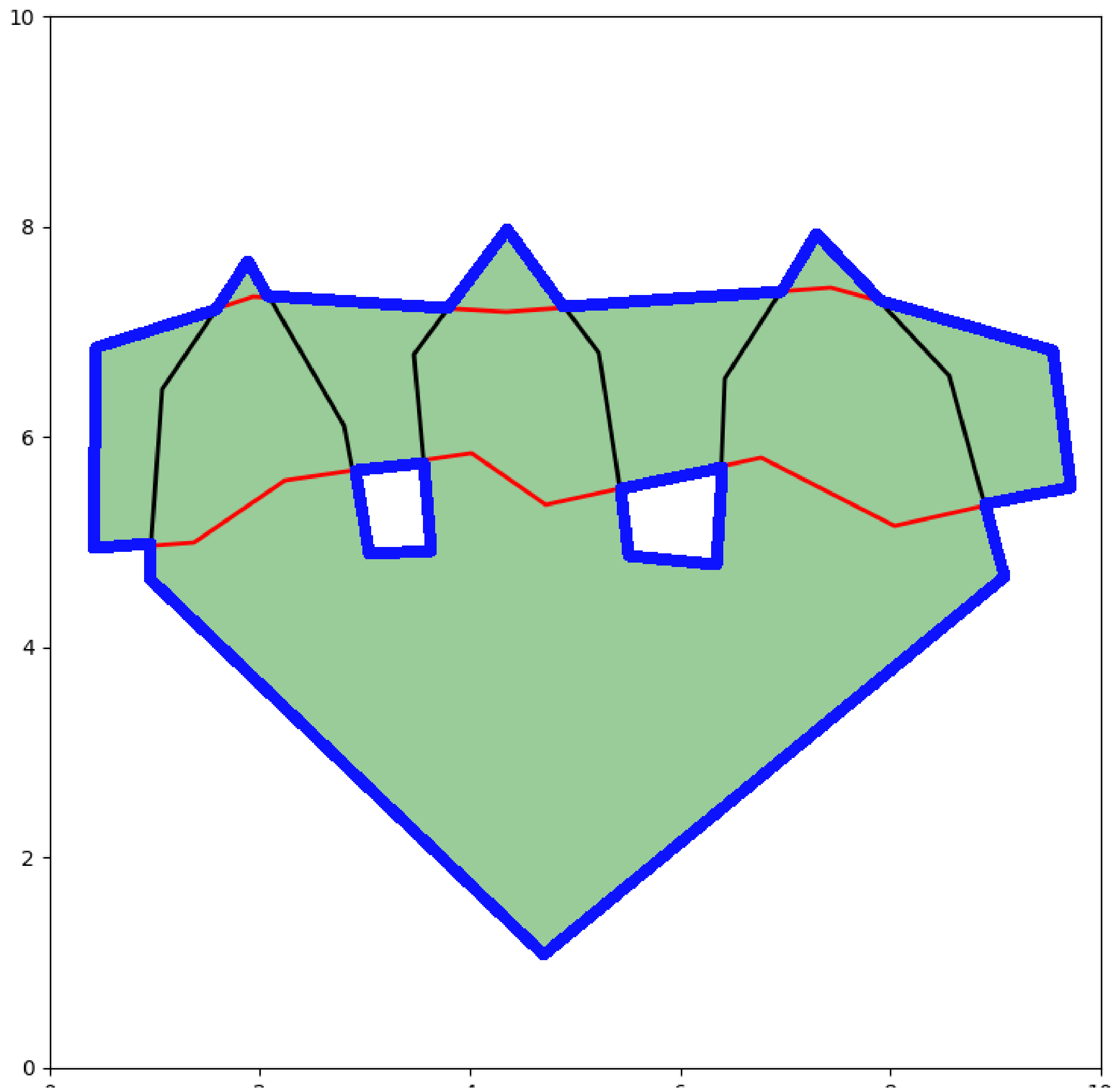
10
8
6
4
2
0
0
2
4
6
8
10
10
8
6
4
2
0
0
2
4
6
8
10

Obsługa dziur w sumie

W niektórych przypadkach suma wielokątów będzie posiadała dziury. W takich przypadkach zwróconym wynikiem algorytmu będzie wielokąt „zewnętrzny”, a także wielokąty „wewnętrzne” – tworzące dziury.

W celu poprawnego wyznaczenia obszaru sumy należy odjąć od obszaru wielokąta zewnętrznego obszary wielokątów wewnętrznych.

Aby sprawdzić, które wielokąty są wewnętrzne, a które zewnętrzne, można użyć metody ray-castingu.



Struktury danych

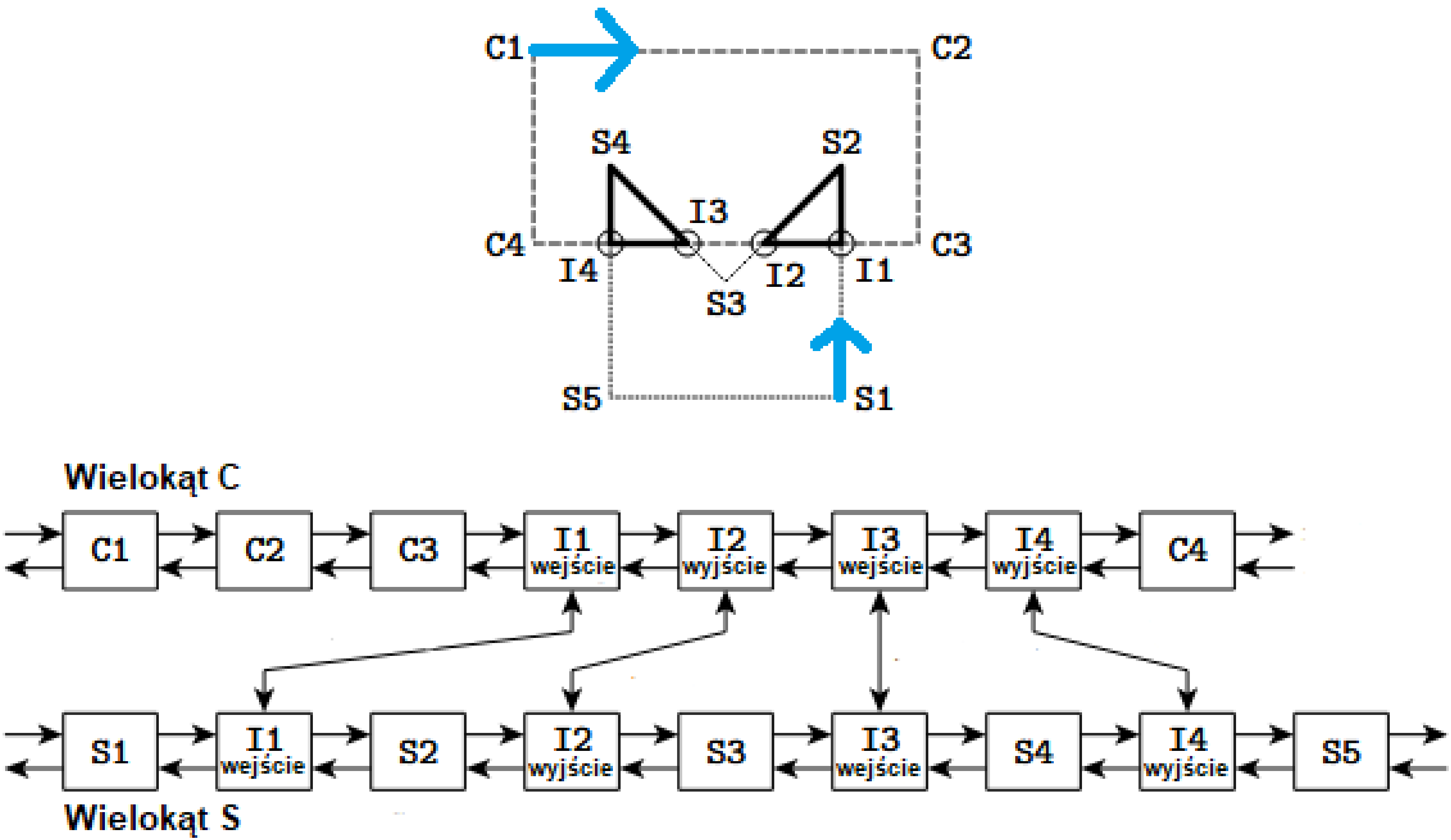
Wielokąt jest reprezentowany przez dwukierunkową listę cykliczną, której elementami są oryginalne wierzchołki wielokątów oraz punkty przecięcia.

Wierzchołki są przechowywane w strukturze posiadającej następujące atrybuty:

- współrzędne wierzchołka,
- indeks (dla oryginalnych wierzchołków wielokąta)
- wskaźniki na poprzedni i następny wierzchołek wielokąta,
- flagi: czy wierzchołek jest punktem przecięcia, czy jest wejściem, czy jest wyjściem,
- wskaźnik na odpowiadający punkt przecięcia w drugim wielokącie (dla punktów przecięcia),
- parametr α (dla punktów przecięcia).

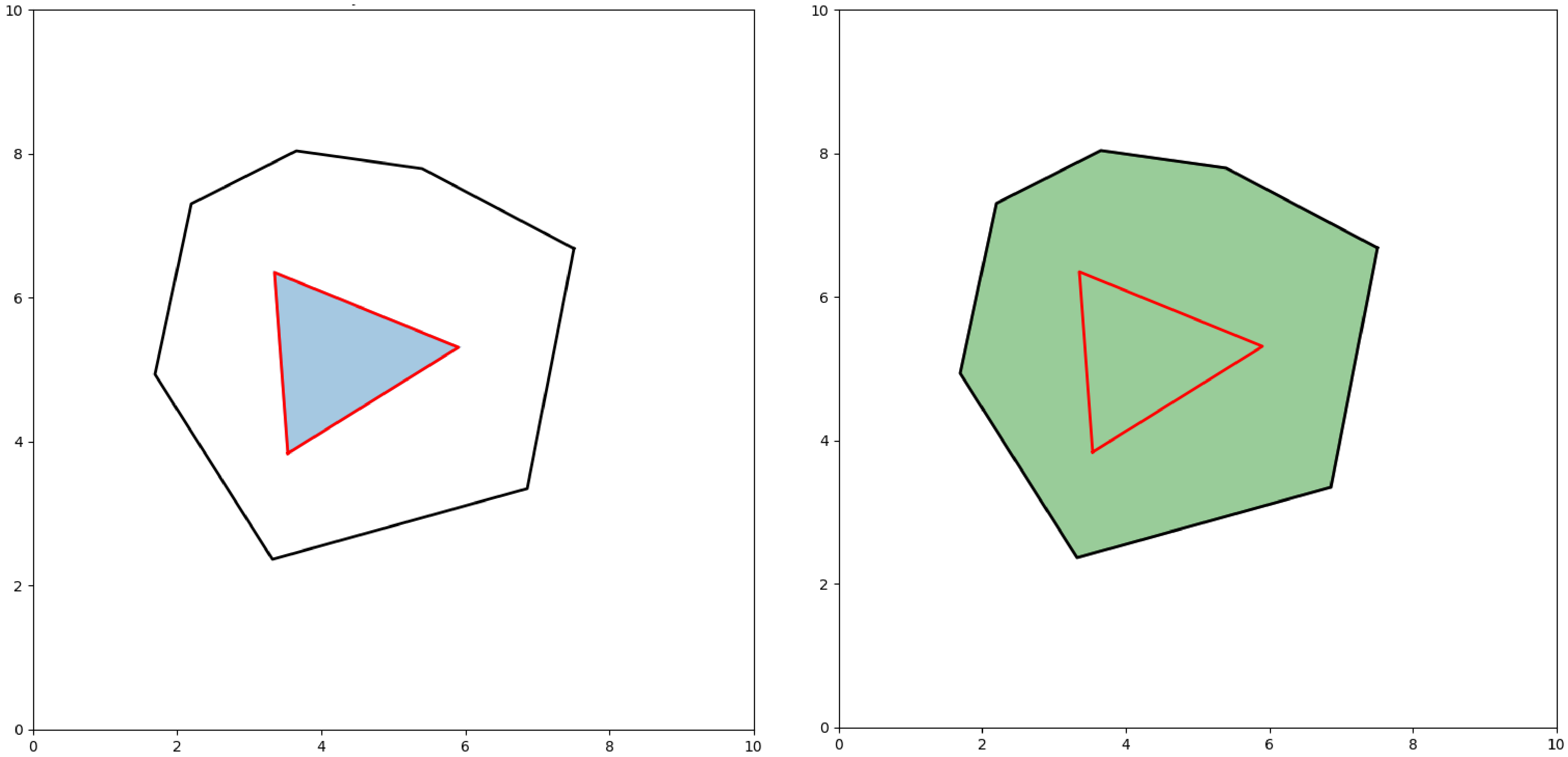
Parametr α określa położenie punktu przecięcia na danej krawędzi („jak daleko" na tej krawędzi się znajduje).

Parametr ten jest potrzebny aby wstawić przecięcia do wielokątów w odpowiedniej kolejności.

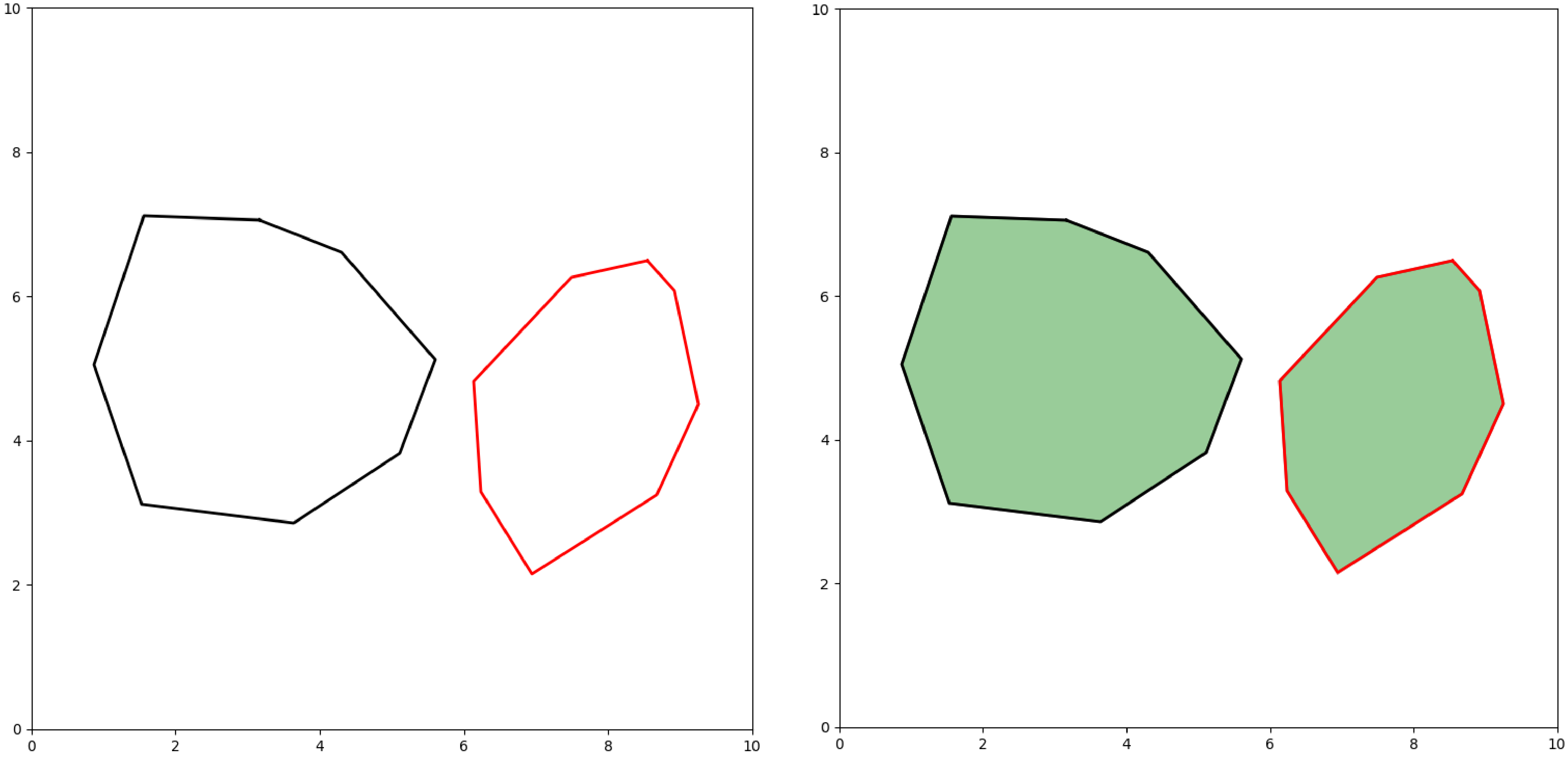

C1
C2
S4
S2
I3
C4
C3
I4
I2
I1
S3
S5
S1
Wielokąt C
C1
C2
C3
I1 wejście
I2 wyjście
I3 wejście
I4 wyjście
C4
S1
I1 wejście
S2
I2 wyjście
S3
I3 wejście
S4
I4 wyjście
S5
Wielokąt S

Przypadki specjalne

- Jeśli wielokąty się nie przecinają, sprawdzane jest położenie jednego wielokąta względem drugiego (za pomocą ray-castingu).
- Jeśli jeden wielokąt leży wewnątrz drugiego, zwracany jest odpowiedni wielokąt.
- Jeśli wielokąty są rozłączne, to w przypadku iloczynu zwracany jest zbiór pusty, a w przypadku sumy oba wielokąty.



10
8
6
4
2
0
0
2
4
6
8
10
10
8
6
4
2
0
0
2
4
6
8
10



0
2
4
6
8
10

Złożoność czasowa i pamięciowa

Złożoność czasowa:

- Wyznaczenie przecięć wielokątów: $O(n * m)$,
- Oznaczenie przecięć jako „punkty wejścia” lub „punkty wyjścia”: $O(n + m + p)$,
- Konstrukcja iloczynu i sumy: $O(n + m + p)$,
- Razem: $O(n * m)$.

Przy użyciu algorytmu zamiatania do znajdowania przecięć można osiągnąć złożoność $O((n + m + p) \log (n + m))$.

Złożoność pamięciowa: $O(n + m + p)$.

n – liczba krawędzi pierwszego wielokąta,

m – liczba krawędzi drugiego wielokąta,

p – liczba przecięć.

Bibliografia

1. Günther Greiner, Kai Hormann (1998). "Efficient clipping of arbitrary polygons". _ACM Transactions on Graphics_, 17(2), 71–83.
2. https://en.wikipedia.org/wiki/Point_in_polygon#Ray_casting_algorithm