

Gramatyki kształtu

1 Cel zajęć

Celem zajęć jest praktyczne zaznajomienie się z pojęciem gramatyki kształtu, wprowadzonym na [wykładzie](#). Pierwsza część zajęć ma na celu zaznajomienie się z możliwościami i funkcjonalnością środowiska, które wykorzystujemy, zaś kolejne ćwiczenia polegają na własnoręcznym wytworzeniu pewnych figur przy użyciu gramatyk kształtu.

2 Narzędzie

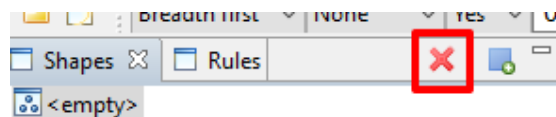
Na zajęciach wykorzystywać będziemy program SGI (Shape Grammar Interpreter), dostępny pod adresem <https://sourceforge.net/projects/sginterpreter/>.

2.1 Uwagi ogólne odnośnie działania

Pojedyncza gramatyka (kształty + reguły) stanowi dla SGI coś w rodzaju projektu. Przechowywana jest ona w postaci pliku .xml (jednego). Plik ten jest tworzony dopiero przy pierwszym zapisie jakiegokolwiek obiektu (kształtu/reguły) wchodzącego w skład projektu, więc w tym momencie SGI poprosi o podanie nazwy pliku. W jednym momencie „aktywna” może być tylko jedna gramatyka.

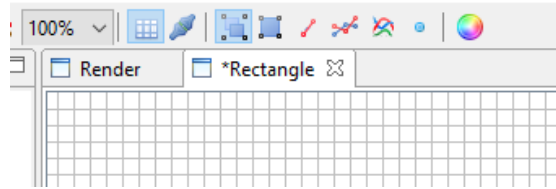
3 Ćwiczenie wprowadzające

Zacznijmy od stworzenia nowego projektu – File → New... → Grammar. W oknie po lewej widoczne są dostępne kształty (shapes) i produkcje (rules). Domyślnie tworzony jest jeden pusty kształt – <empty>. Edytując właściwości (*Properties*) kształtu możemy zmienić jego nazwę, np. na Square. Dodawać nowe kształty można poprzez File → New... → Shape, lub przy użyciu przycisku zaznaczonego na poniższym screenie



Tworzenie kształtu

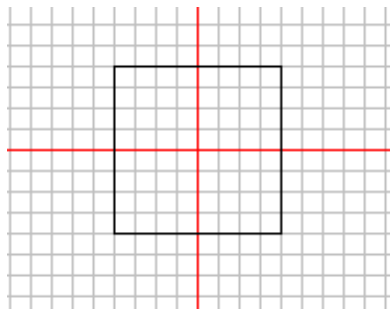
Aby stworzyć kształt, należy wykorzystać narzędzia kształtu, znajdujące się nad głównym panelem z kształtem.



Jako pierwszy kształt stworzymy kwadrat. W tym celu wybrać należy **PolyLine**, które służy do tworzenia linii łamanych. Aby stworzyć kształt zamknięty, należy klikać kolejno w miejscach, gdzie chcemy wstawić wierzchołki, a na końcu kliknąć dwukrotnie na wierzchołku początkowym (zdaje się że nie ma niestety opcji „domknięcia” linii, która uczyniłaby to automatycznie). Po stworzeniu kształt można modyfikować w dwóch trybach:

- Modify layout – tryb „wysokopoziomowy”, pozwala przenosić i skalować cały kształt
- Modify vertices – tryb „niskopoziomowy”, pozwala ustawiać położenie poszczególnych wierzchołków

Powinniśmy uzyskać coś takiego:



Położenie względem środka układu nie jest istotne, natomiast dla dalszej części ćwiczenia požądane jest, aby długości boków (wyrażone w ilości „kratek”) były parzyste.

Tworzenie produkcji

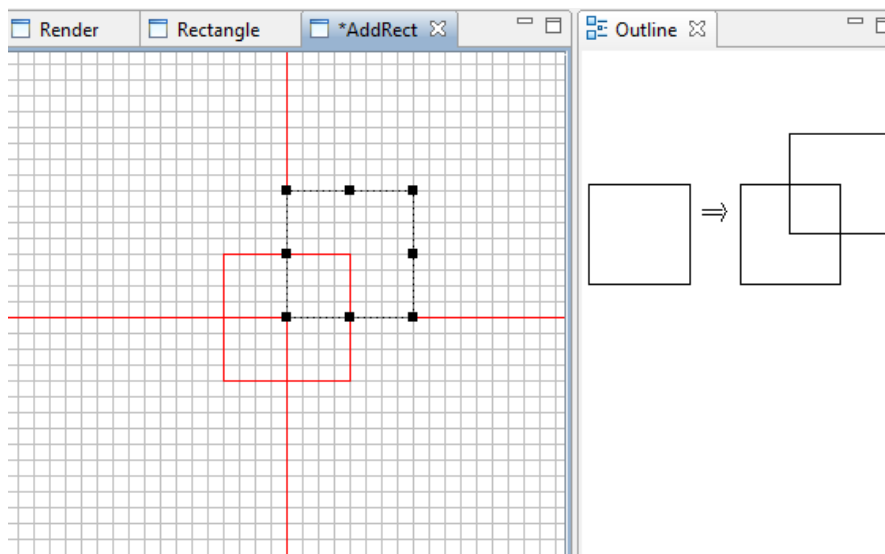
Stworzymy produkcję (regułę), która do istniejącego kwadratu dodawać będzie drugi, o tym samym rozmiarze, nieco przesunięty względem pierwszego. Aby dodać do gramatyki regułę, należy wybrać z menu głównego **File** → **New...** → **<Typ> Rule**, gdzie typ oznacza, w jaki sposób działa reguła:

- Addition – do istniejącego kształtu dokładany jest inny

- **Modification** – istniejący kształt jest modyfikowany (np. zmniejszany, przesuwany)
- **Substitution** – istniejący kształt jest zastępowany innym

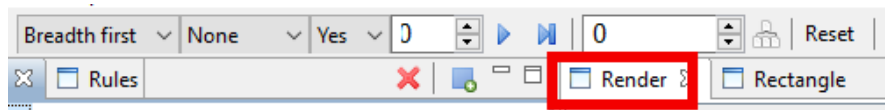
W pozostałej części ćwiczenia będziemy mieli okazję wykorzystać wszystkie rodzaje produkcji, na razie wybieramy **Addition**.

Produkcje w SGI są wyświetlane w pojedynczym widoku, w którym znajduje się zarówno lewa (zaznaczona na czerwono), jak i prawa (czarna) strona. Nie da się bezpośrednio edytować stron produkcji, należy dodać do nich istniejące kształty. Aby dodać kształt do którejś strony produkcji, należy z jego menu kontekstowego wybrać **Add to (left|right) side**. W naszej produkcji chcemy dodać kwadrat do kwadratu, więc umieszczamy kwadrat po stronach produkcji (w przypadku typu **addition** prawa strona jest dodawana do lewej w kształcie wynikowym). Następnie należy przesunąć jeden z kwadratów tak, aby uzyskać następujący układ:

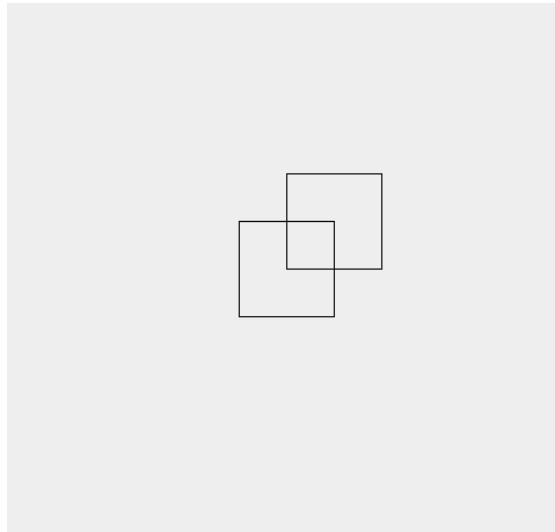


Zastosowanie produkcji gramatyki

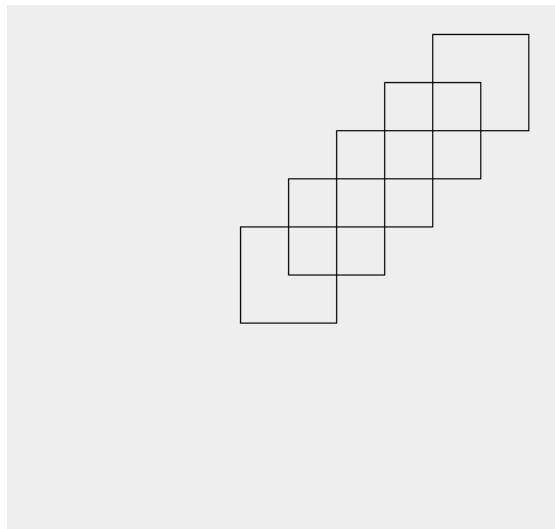
Aby zastosować gramatykę, najpierw należy ustawić kształt początkowy. W tym celu z menu kontekstowego kształtu **Square** należy wybrać **Set as Start Shape**. Następnie przełączyć należy się na zakładkę **Render**



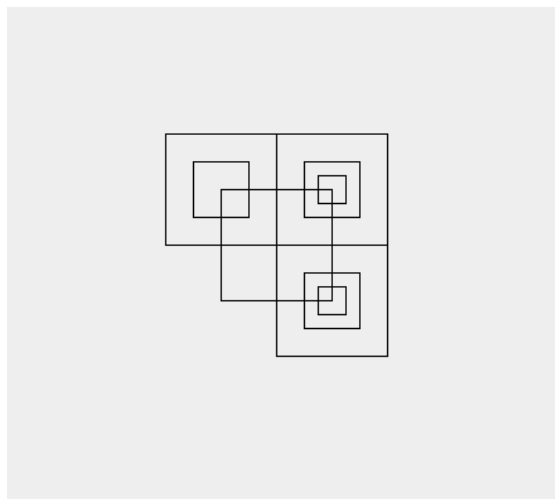
Początkowo w widoku renderingu wyświetlany jest wybrany kształt początkowy – w naszym przypadku kwadrat. Po jednokrotnym uruchomieniu polecenia **Render Next Step** powinniśmy otrzymać coś takiego:



Kolejne uruchomienia wykonują produkcję na kolejnych kwadratach:



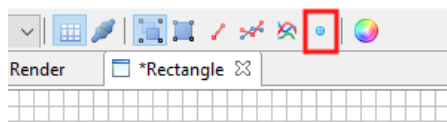
Rezultat jest oczekiwany, jednak mało interesujący. W szczególności, każda aplikacja produkcji tworzy, poza zwykłym, dodatkowym kwadratem, jeden mały, powstały w wyniku przecięcia dwóch dużych (tzw. *emergent shape* – kształt pasujący do lewej strony produkcji który powstał w wyniku zastosowania produkcji gramatyk). Dla niego również możliwe jest zastosowanie produkcji, jednak SGI domyślnie tego nie robi. Aby temu zaradzić, należy zmienić tryb renderingu z domyślnego „Breadth first” na „Subshape”. Rezultat przedstawia poniższy rysunek:



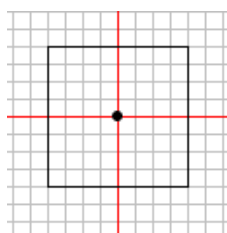
Markery

Czasem być może chcielibyśmy wyspecyfikować pewne dodatkowe ograniczenia na stosowalność jakiejś produkcji, które też mają charakter geometryczny, ale które ciężko wyrazić samym kształtem. Np. w powyższym przykładzie można by chcieć ograniczyć możliwość stosowania produkcji tak, by nie można było jej stosować do małych kwadratów powstałych w wyniku przecięcia dwóch dużych, ale żeby można było mimo to stosować ją w przypadku, gdy z połączenia czterech dużych kwadratów powstanie jeszcze większy.

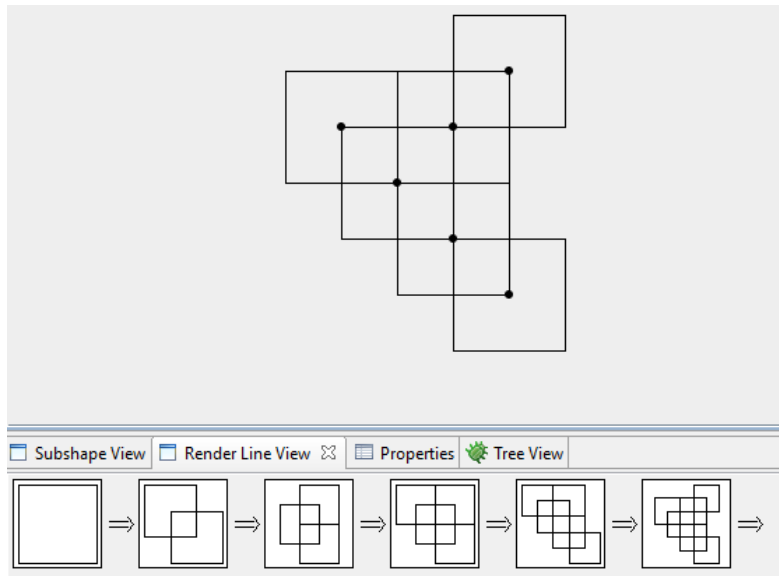
W tym celu wykorzystać można mechanizm markerów. Otwórzmy raz jeszcze kształt Square i wybierzmy narzędzie Marker



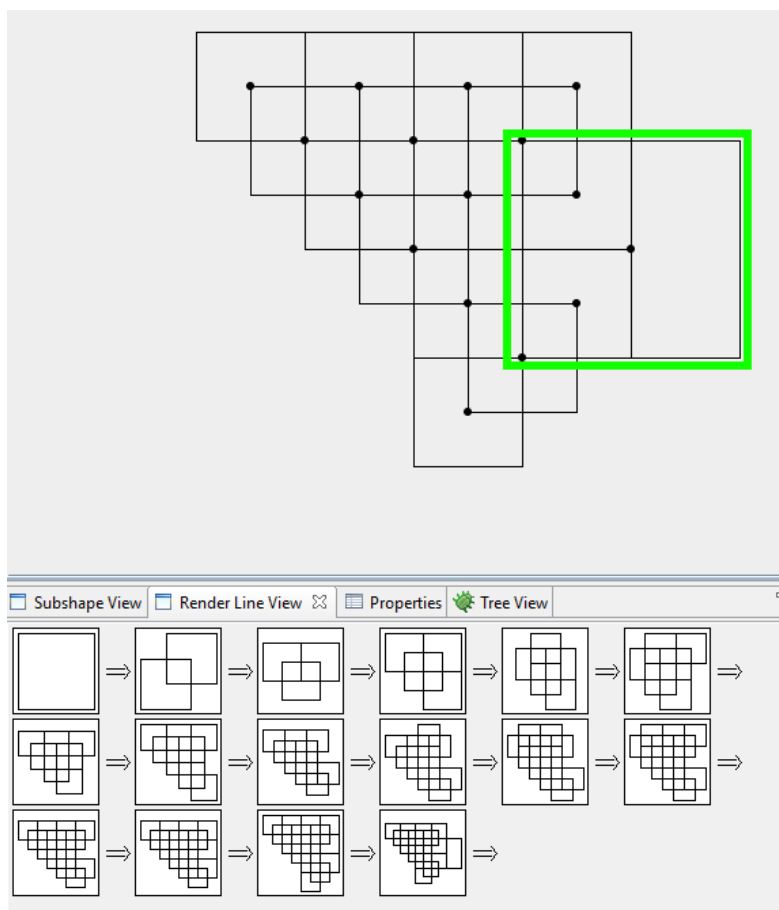
Dodajmy marker na środku kwadratu:



Teraz uruchamianie produkcji w trybie subshape w istocie nie wykorzystują małych kwadratów:



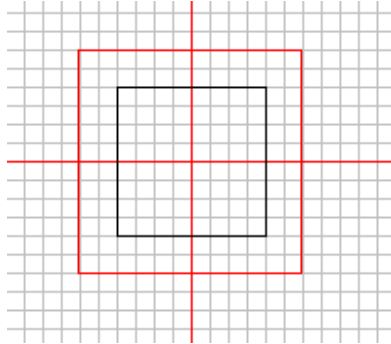
natomiast wykorzystują większe:



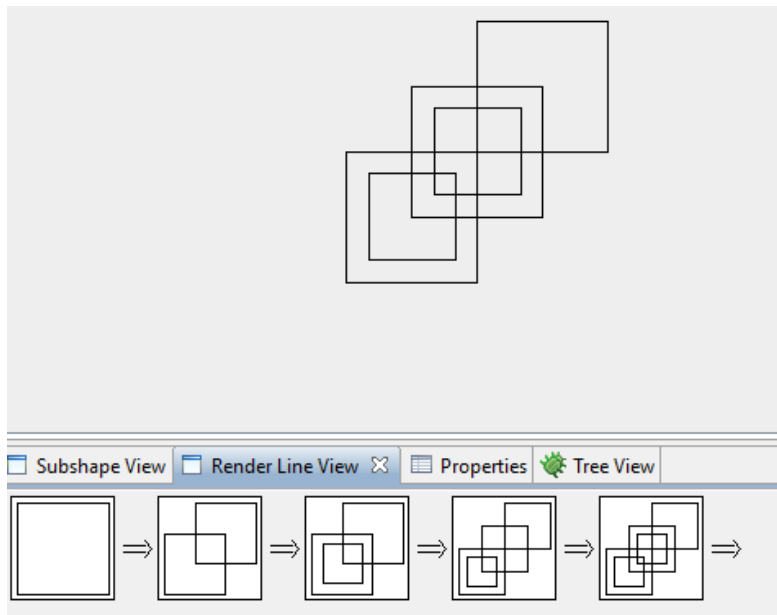
Usuńmy marker przed przejściem do kolejnych części (**Modify layout**)

Produkcje modyfikujące

Stwórzmy nową produkcję typu modyfikującego i nazwijmy ją **Shrink**. Tym razem wystarczy dodać kształt do lewej strony – po prawej zawsze znajduje się ten sam kształt, podany modyfikacji. Przeskalujemy kształt po prawej stronie (czarny) tak, by był mniejszy od oryginalnego (czerwonego), lub odwrotnie – przeskalujemy czerwony tak, by był większy od czarnego (działa w obie strony):

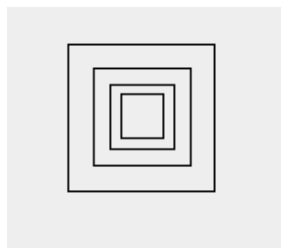


Jak można zaobserwować, nasze dwie produkcje wykonywane są na przemian:



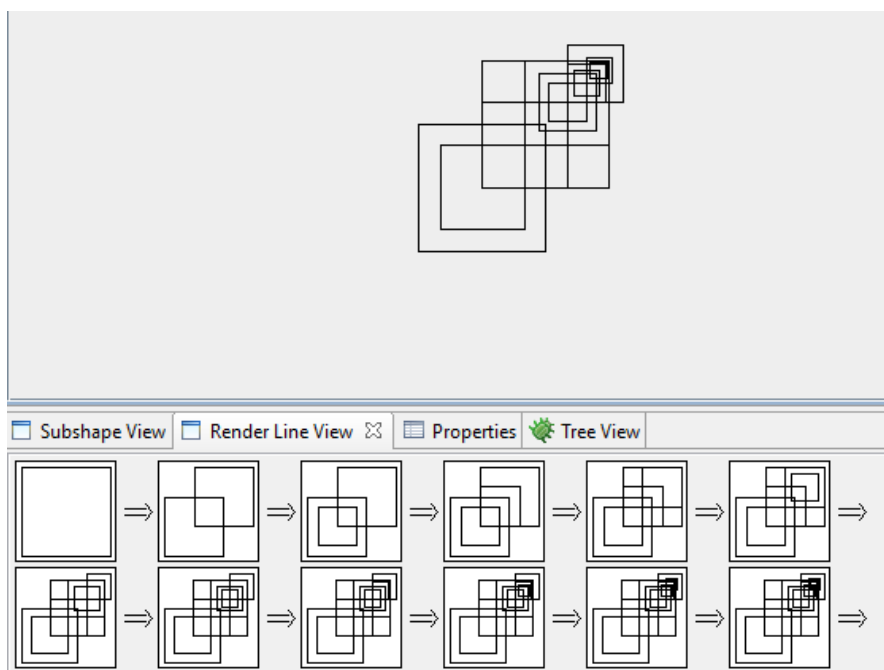
Zastosowanie produkcji gramatyk 2

Zmieńmy tryb renderingu na **Depth-first** – tym razem wykonywana cały czas jest produkcja zmniejszająca:

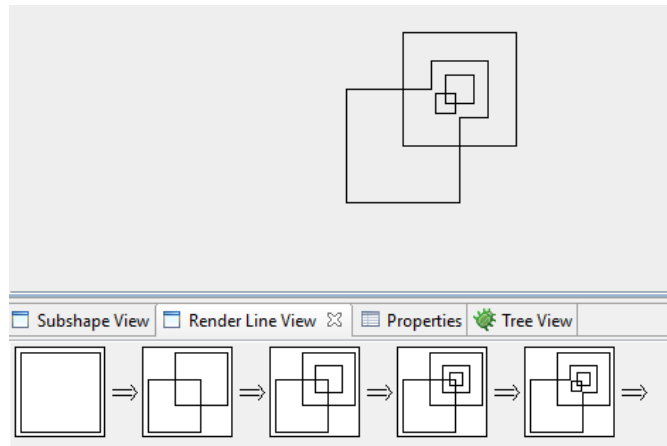


Wybór pomiędzy DFS a BFS w tym przypadku oznacza wybranie w jaki sposób rozwijane są drzewa możliwych zastosowań produkcji gramatyk. DFS wybiera cały czas gałąź dotyczącą tej samej produkcji (jeśli może), natomiast BFS stara się przechodzić kolejno przez wszystkie produkcje. To, co robi Subshape zdaje się nie mieć tak prostej interpretacji (nie jest jasne, jak wybierane są produkcje).

Nieco bardziej ciekawym rezultatem strategii DFS można uczynić wprowadzając losowość – zmiana z **None** na **Low** daje następujący rezultat:

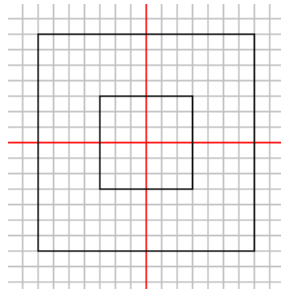


Dodatkowo, strategia **Subshape** potrafi teraz zmniejszać małe kwadraty powstałe w wyniku przecięcia dużych:

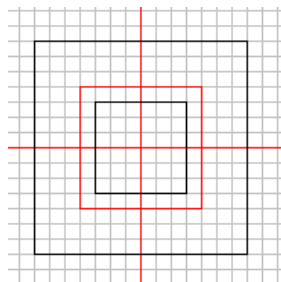


Produkcje podstawiające

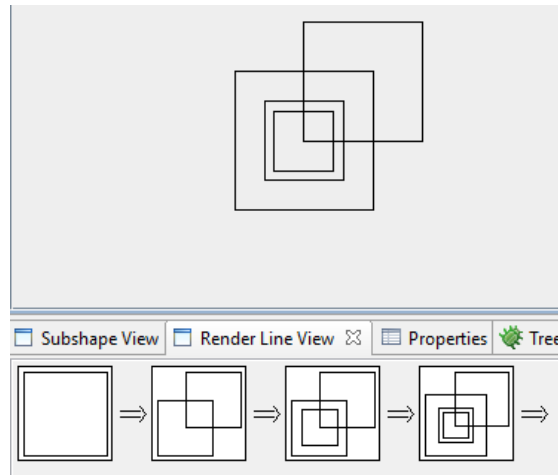
Ostatni rodzaj produkcji to produkcje podstawiające, które zamieniają kształt po lewej stronie na kształt po prawej. Stworzmy najpierw kształt składający się z dwóch kwadratów:



Do lewej strony produkcji podstawiającej dodajmy **Square**, do prawej – nowo stworzony kształt:



Produkcja działa bez większych niespodzianek:

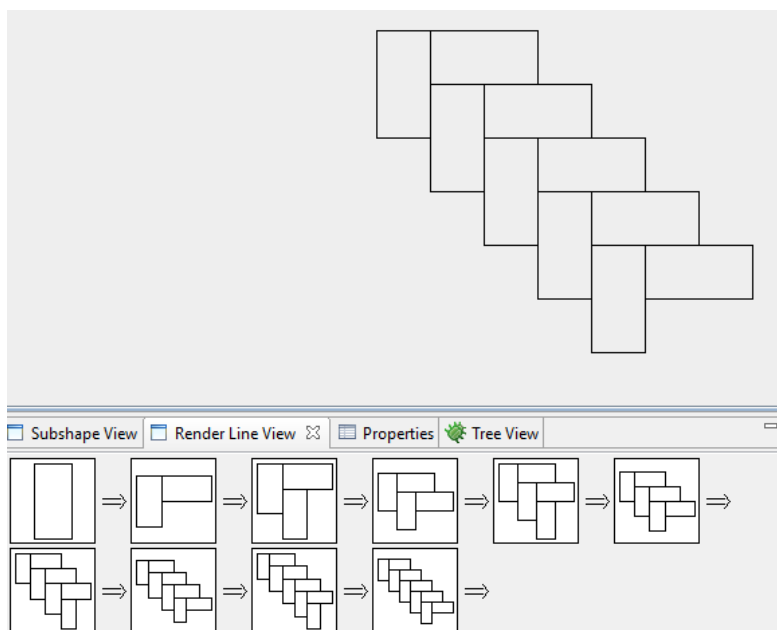


4 Zadania

4.1 „Warkocz”

Przy użyciu dowolnych konstrukcji stwórz kształt zaprezentowany na poniższym diagramie. Wyprowadzenie z gramatyki powinno zgadzać się z tym widocznym pod samym kształtem (żadnego „hardkodowania” :). Gramatyka powinna mieć możliwość tworzenia dowolnie długich warkoczy.

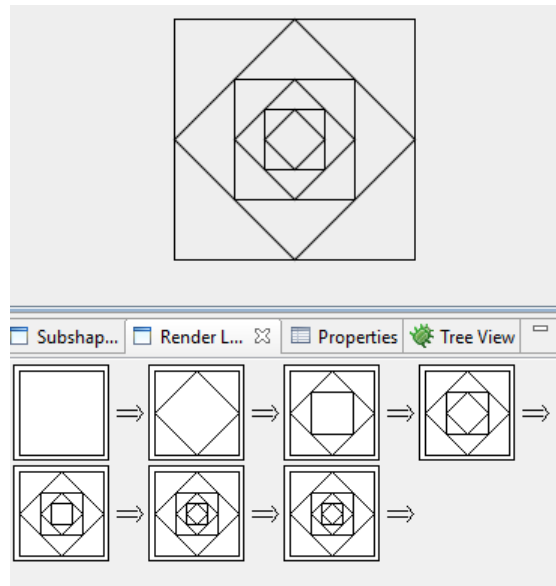
Hint: można to zrobić używając dwóch kształtów i dwóch produkcji



4.2 Zagnieżdżone kwadraty

Ograniczenia i warunki jak w zadaniu powyżej.

Hint: przydatny może okazać się subshape rendering



4.3 * „Warkocz” raz jeszcze

Skonstruuj figurę z zadania pierwszego używając dwóch kształtów i jednej produkcji

Hint: wykorzystaj Subshape rendering i markery