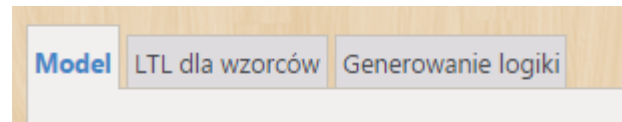
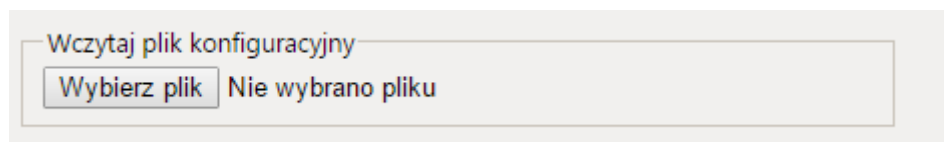


## Obsługa aplikacji.

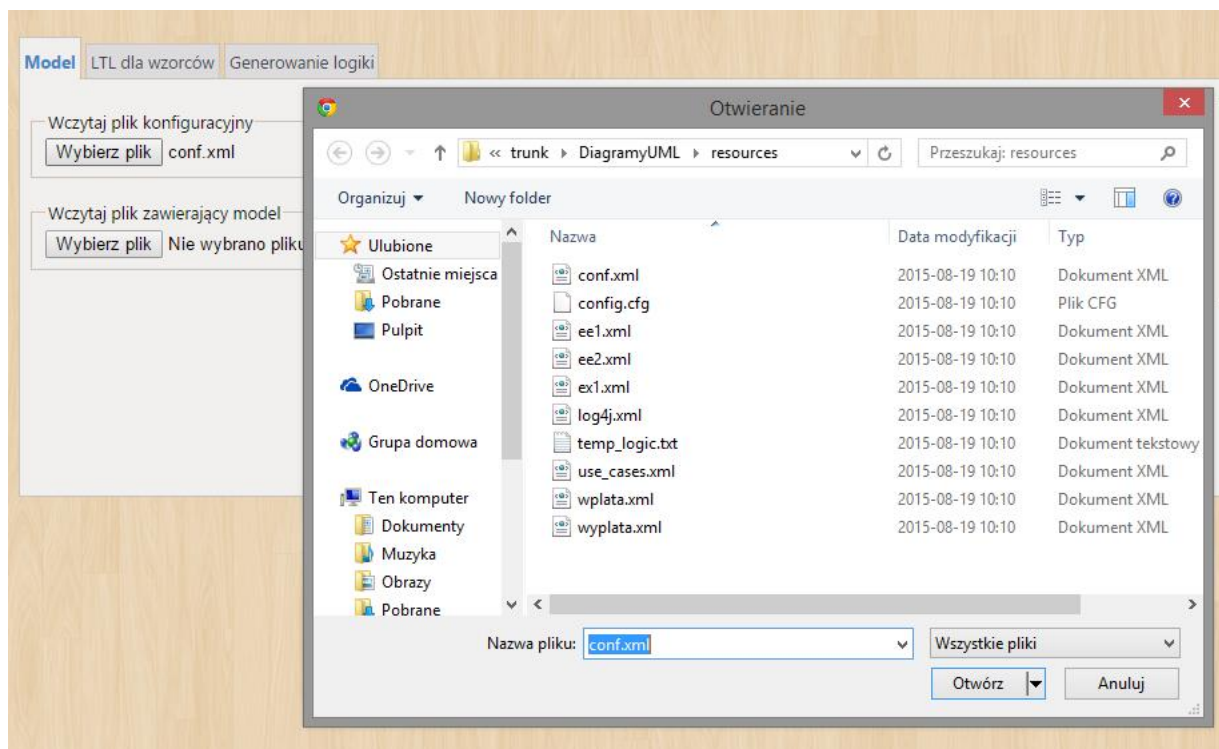
Aplikacja generuje specyfikację logiczną oprogramowania na podstawie analizy diagramów aktywności UML według ustalonych wzorców logiki LTL. Uruchomioną w dowolnej przeglądarce internetowej, aplikację obsługujemy za pomocą trzech zakładek.



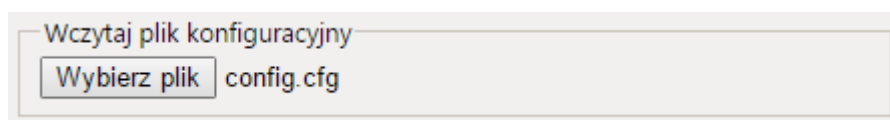
W pierwszym kroku, w zakładce „Model” należy wczytać do programu dwa pliki. Plik z konfiguracją dla aplikacji wybieramy klikając w polu „Wczytaj plik konfiguracyjny” przycisk „Wybierz plik”.



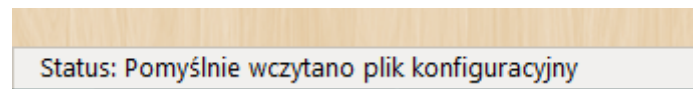
W otwartym okienku wybieramy odpowiedni plik z konfiguracją.



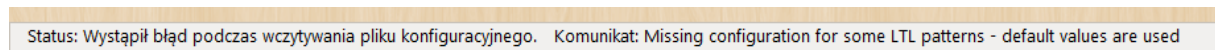
Nazwa otwartego pliku pojawi się tuż obok przycisku. Natomiast klikając ponownie „Wybierz plik” możemy zmienić swój wybór.



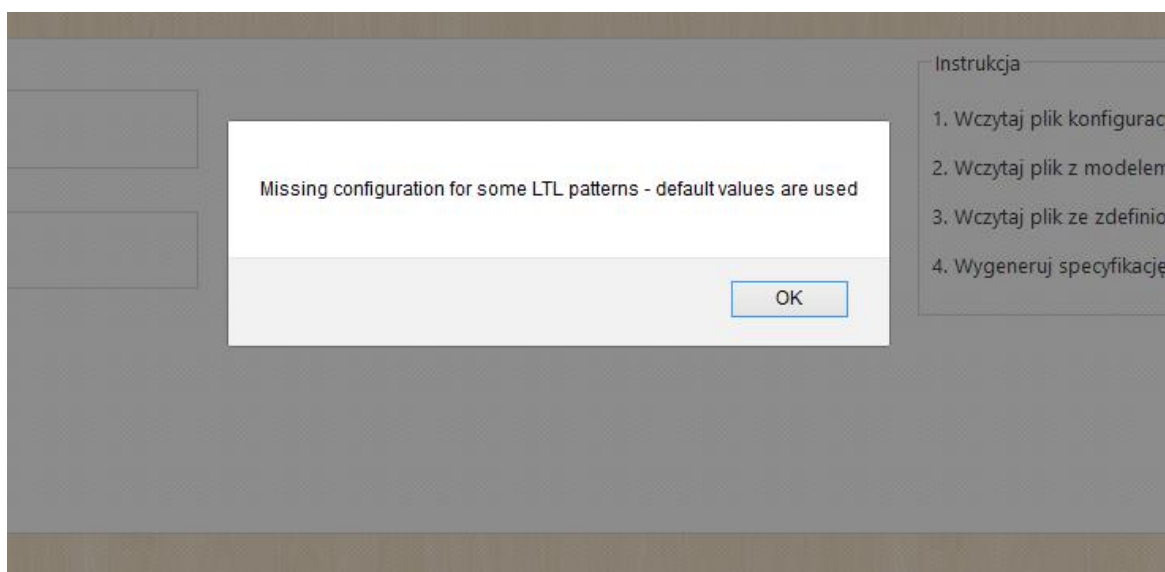
O pomyślnym wykonaniu akcji informuje pasek statusu znajdujący się na dole strony.



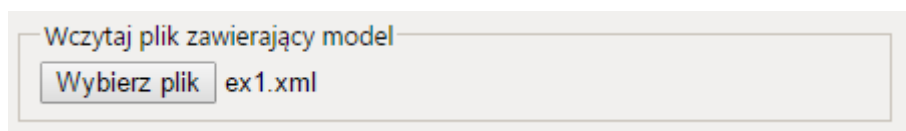
Podczas pracy z aplikacją, mogą wystąpić także różne błędy, pasek statusu wyświetli odpowiedni komunikat. Po wczytaniu nieodpowiedniego pliku konfiguracyjnego pasek statusu wyświetli informację:



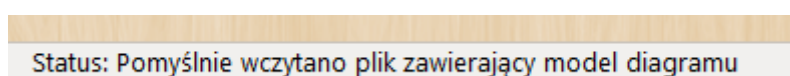
Użytkownik zostanie poinformowany o wystąpieniu błędu także za pomocą okna powiadomienia z odpowiednim komunikatem:



Kolejnym krokiem jest wczytanie do programu pliku zawierającego model diagramu UML. Odpowiedni plik w formacie XML przygotowywany jest za pomocą aplikacji VisualParadigm. Analogicznie jak dla pliku konfiguracyjnego - klikamy przycisk „Wybierz plik” w polu „Wczytaj plik zawierający model”.



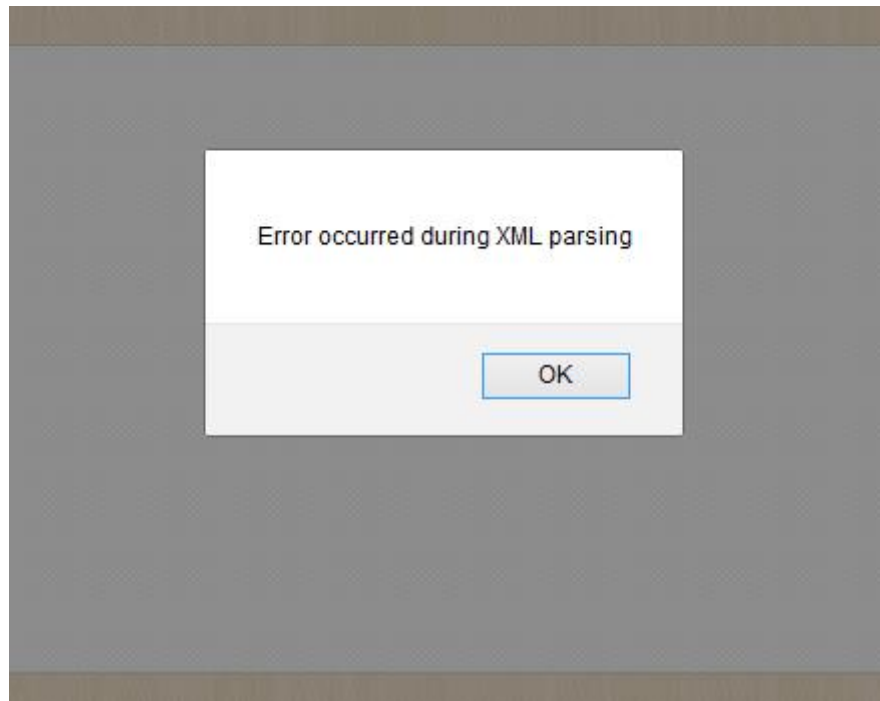
Pomyślne wczytanie pliku zostanie potwierdzone przez pasek statusu:



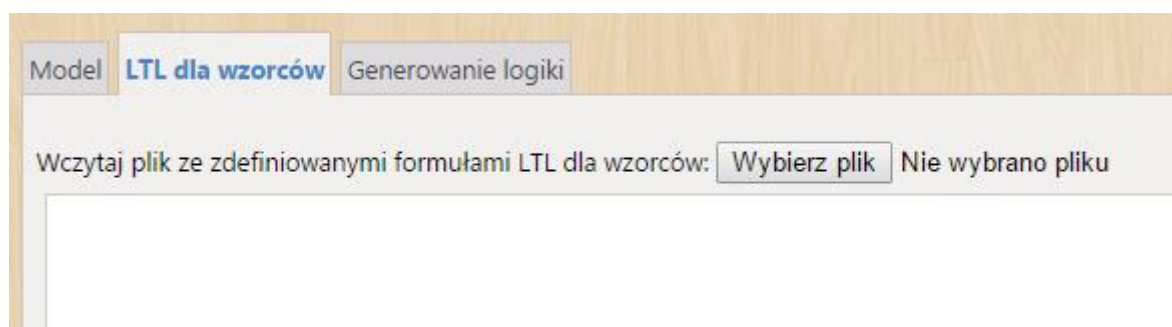
Z kolei błędnie wybrany plik skutkuje poniższym komunikatem:

Status: Wystąpił błąd podczas wczytywania pliku zawierającego model diagramu. Komunikat: Error occurred during XML parsing

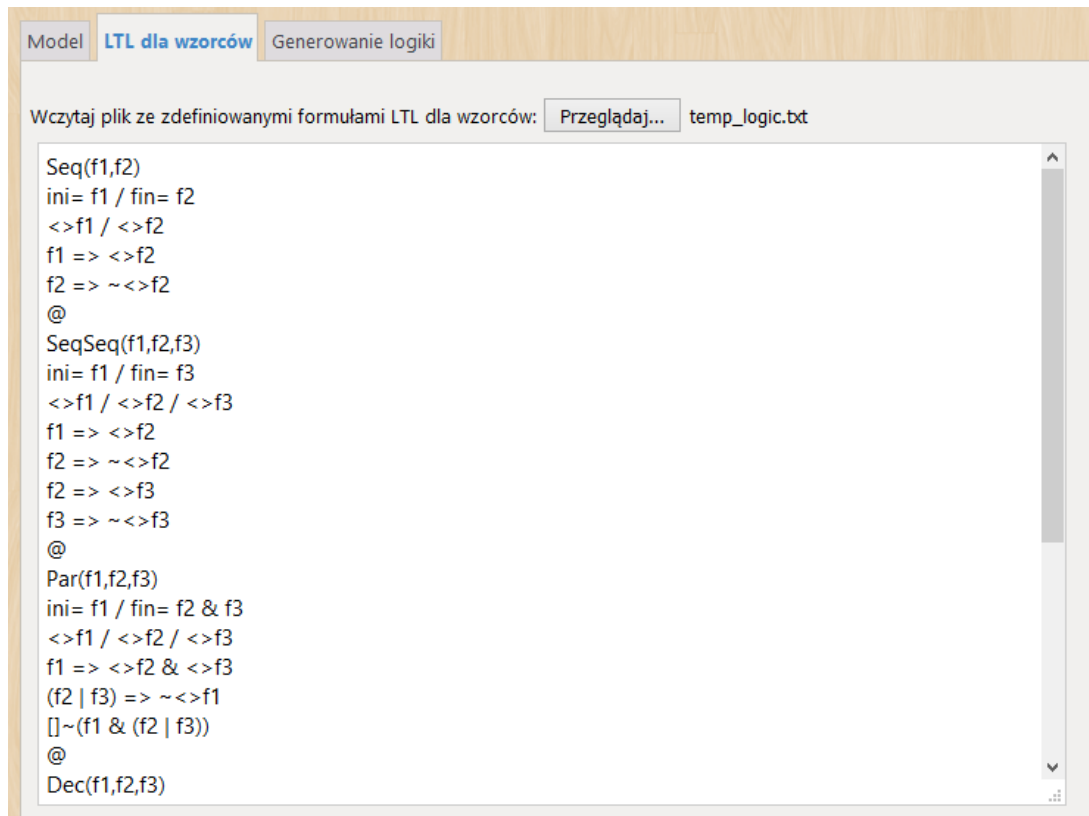
Oraz powiadomieniem:



Następnym etapem jest przejście do zakładki „LTL dla wzorców”. Tutaj również należy analogicznie do poprzednich kroków wybrać odpowiedni plik, tym razem zawierający zdefiniowane formuły logiki LTL. W tym celu klikamy „Wybierz plik”, a następnie z okna systemowego wybieramy odpowiednią pozycję.



W oknie tekstowym wyświetlone zostaną formuły logiki LTL zawarte w pliku. Rozmiar okna można dostosować za pomocą przycisku w prawym dolnym rogu.

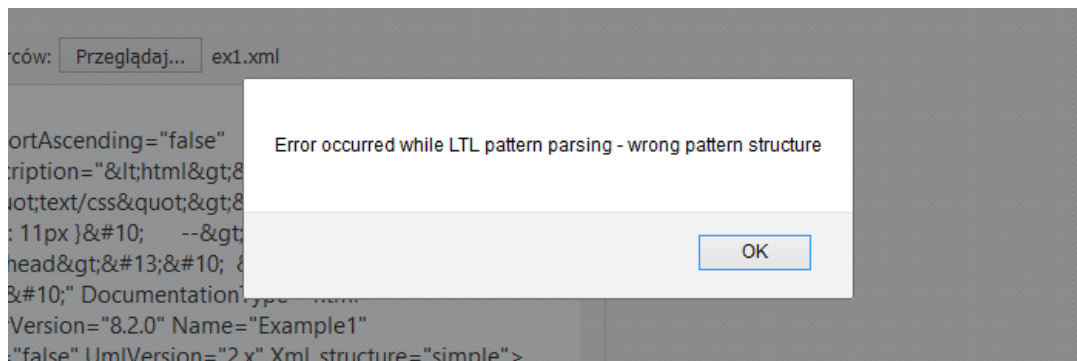


Pomyślne zakończenie akcji potwierdzi pasek statusu:

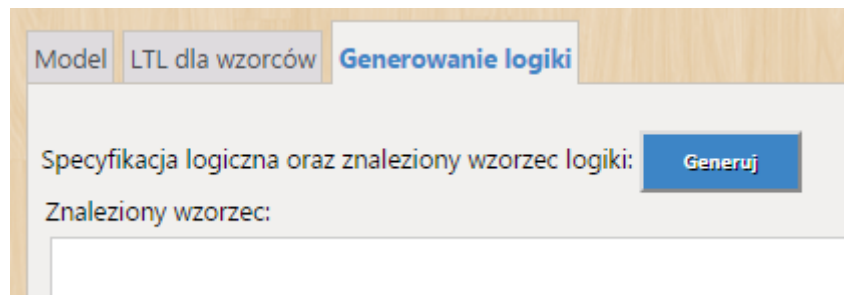
Status: Pomyślnie wczytano plik ze zdefiniowanymi formułami LTL

Komunikat po wczytaniu nieodpowiedniego pliku:

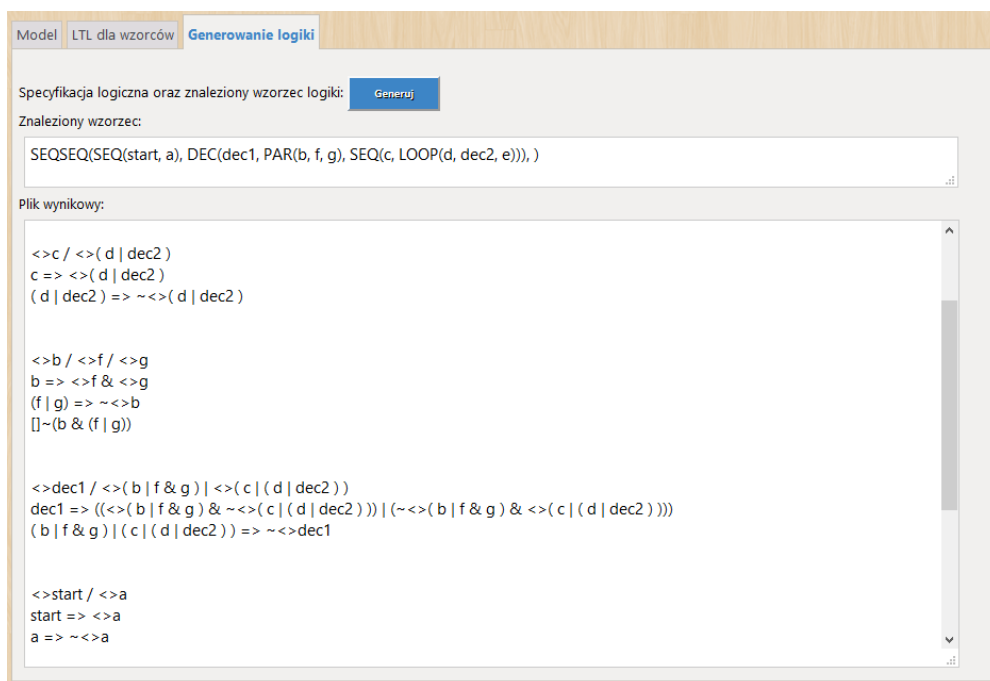
Status: Wystąpił błąd podczas wczytywania pliku z formułami LTL. Komunikat: Error occurred while LTL pattern parsing - wrong pattern structure



Ostatnim krokiem jest przejście do zakładki „Generowanie logiki” i kliknięcie przycisku „Generuj”. Na podstawie wczytanych wcześniej plików w aplikacji zostanie wygenerowana specyfikacja logiczna, a następnie wyświetlona w polu tekstowym.



Wynik działania programu:

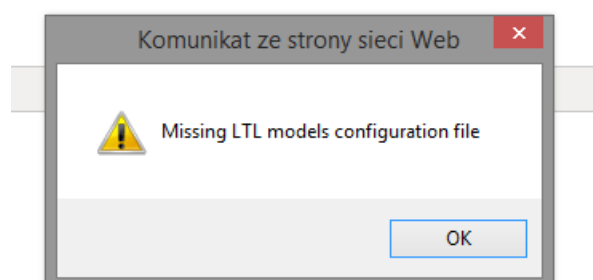


Potwierdzenie na pasku statusu:

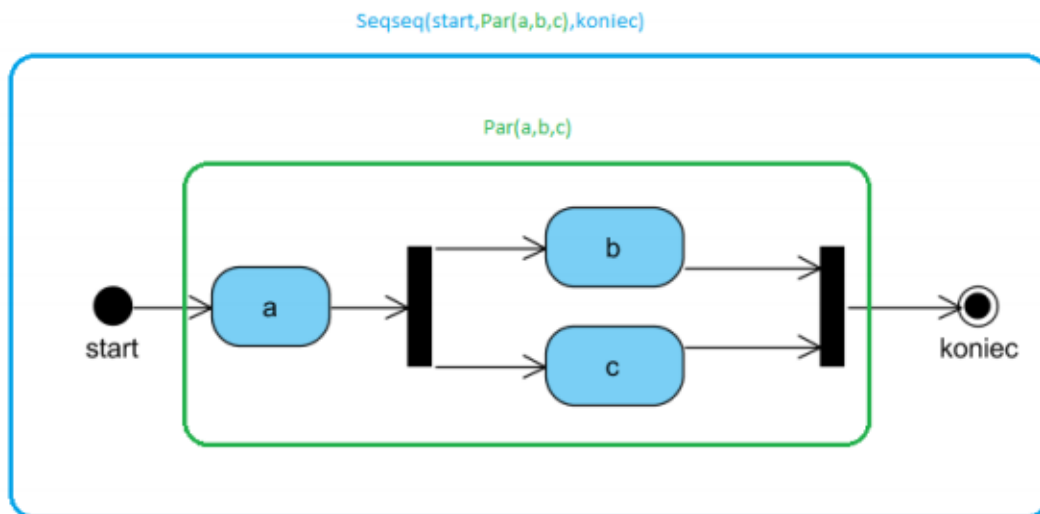
Status: Pomyślnie wygenerowano specyfikację!

Komunikat zwrócony przez aplikację, po próbie wygenerowania specyfikacji z nieodpowiednich plików:

Status: Wystąpił błąd podczas generowania specyfikacji. Komunikat: Missing LTL models configuration file



## Analiza prostego przykładu.



Wykryty wzorzec: `seqseq(start,par(a,b,c),koniec)`

**A.** Rozpoczynamy od najprostszego wzorca, a więc `par`.

Nie posiada on w sobie zagnieżdżonych żadnych innych wzorców, więc do zbioru formuł dopisywane są formuły z podanego pliku i odpowiednie argumenty podstawiane są w miejsca parametrów `f1`, `f2` i `f3`:

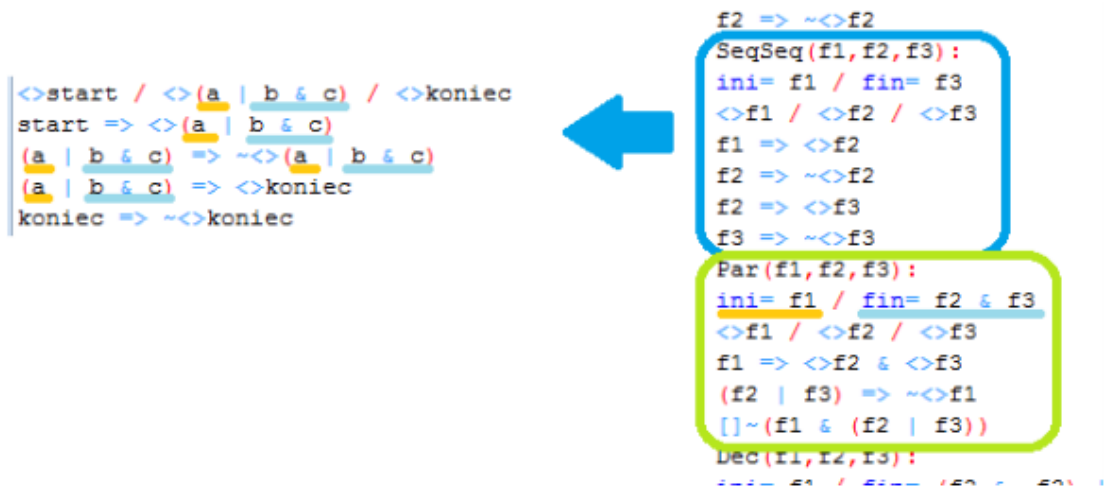
<pre> &lt;&gt;a / &lt;&gt;b / &lt;&gt;c a =&gt; &lt;&gt;b &amp; &lt;&gt;c (b   c) =&gt; ~&lt;&gt;a []~(a &amp; (b   c))         </pre>		<pre> I3 =&gt; ~&lt;&gt;I3 Par(f1,f2,f3): ini= f1 / fin= f2 &amp; f3 &lt;&gt;f1 / &lt;&gt;f2 / &lt;&gt;f3 f1 =&gt; &lt;&gt;f2 &amp; &lt;&gt;f3 (f2   f3) =&gt; ~&lt;&gt;f1 []~(f1 &amp; (f2   f3)) Dec(f1,f2,f3):         </pre>
--	--	--

B. Kolejnym wzorcem jest **Seqseq**.

Jednym z argumentów wzorca **Seqseq** jest w tym wypadku inny wzorzec (**par**).

Tak więc najpierw do zbioru formuł dopisujemy formuły dla **Seqseq** w miejscu argumentu **f2** podstawiając alternatywę wyrażeń **ini** oraz **fin** wzorca **par**, które są w tym wypadku odpowiednio: **a** oraz **b & c**. Argumenty **f1** i **f3** wynoszą odpowiednio **start** i **koniec**.

Formuły dla tego wzorca zostaną stworzone w następujący sposób:



Całość przedstawia się następująco:

