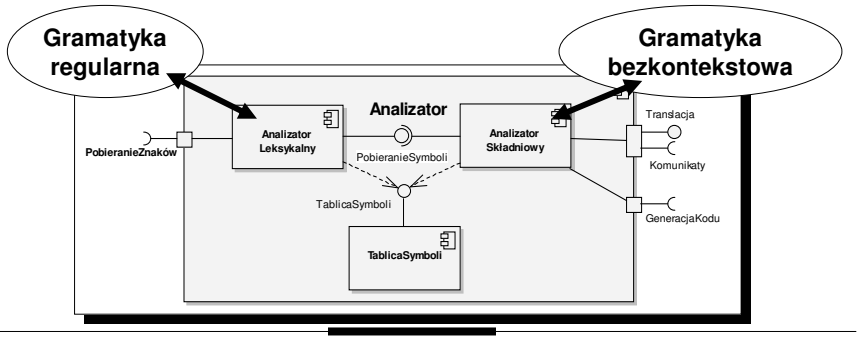
# Gramatyki regularne i automaty skończone

- Alfabet, język, gramatyka - podstawowe pojęcia
- Co to jest gramatyka regularna, co to jest automat skończony?




Języki formalne i kompilatory, © by Michał Śmiałek

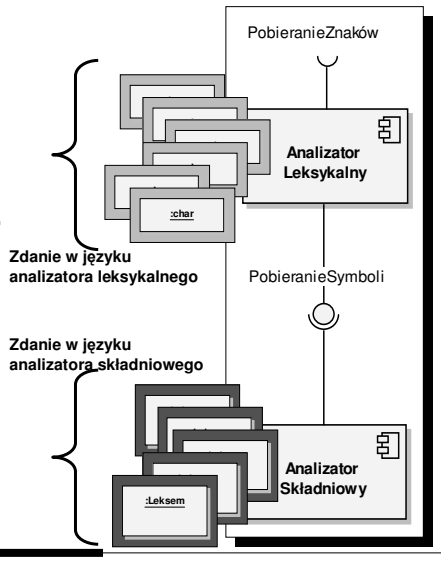
## Gdzie jesteśmy?

**Gramatyka określa poprawne zdania w danym języku.**

**Język dla analizatora leksykalnego określa wszystkie poprawne ciągi znaków tworzących pojedyncze leksemy.**

**Język dla analizatora składniowego określa wszystkie poprawne ciągi leksemów.**

**Gramatyki regularne służą przede wszystkim do definiowania języków dla analizatorów leksykalnych.**




Języki formalne i kompilatory © by Michał Śmiałek

## Alfabety i zdania

**Podstawowe definicje:**

- Alfabet (słownik) to nie uporządkowany zbiór symboli.
- Zdanie (ciąg symboli) to skończona sekwencja złożona z symboli alfabetu (symboli terminalnych).
- Zdanie puste (ozn. λ) to zdanie nie zawierające żadnych symboli.
- Długość zdania (ozn. |ω|) to liczba symboli, z których się ono składa.
- Zbiór wszystkich możliwych zdań alfabetu Σ oznaczamy przez $\Sigma^+$.
- Zbiór $\Sigma^+$ wraz ze zdaniem pustym oznaczamy przez $\Sigma^*$ ($\Sigma^* = \Sigma^+ \cup \lambda$).

**Ważne pytania:**

- Jaki jest alfabet dla analizatora leksykalnego?
- Jaki jest alfabet dla analizatora składni?




## Języki i gramatyki

**Język L określony dla alfabetu Σ stanowi podzbiór zboru $\Sigma^*$:**

- $L \subseteq \Sigma^*$

**Złożeniem (konkatenacją) dwóch języków $L_1$ i $L_2$ jest zbiór napisów αβ:**

- $L_1L_2 = \{\alpha\beta \mid \alpha \in L_1 \text{ i } \beta \in L_2\}$

**Gramatyka G definiuje język L(G) nad alfabetem Σ. Definicja ta polega na określeniu wszystkich możliwych napisów języka (zdań). Nie wypisujemy jednak wszystkich możliwych napisów (może ich być nieskończenie dużo). Zamiast tego opisujemy proces „wyprowadzania" napisów języka.**

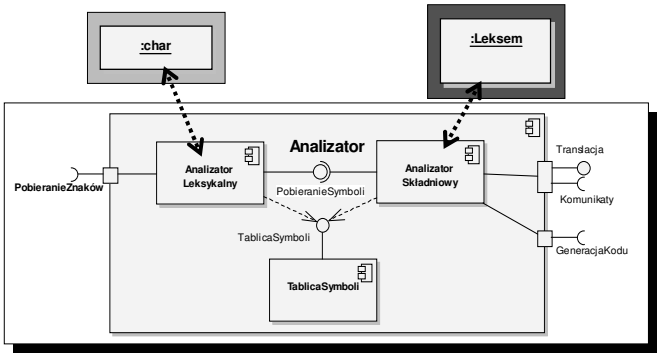
**Ważne pytania:**

- Co opisuje język dla analizatora leksykalnego?
- Co opisuje język dla analizatora składni?

## Alfabety i języki a struktura kompilatora

**Ważne: alfabet i język dla analizatora leksykalnego (a.l.) i składniowego (a.s.) są różne:**

- Symbole terminalne dla a.l. to znaki ze strumienia wejściowego.
- Symbole terminalne dla a.s. to leksemy produkowane przez a.l.




Języki formalne i kompilatory © by Michał Śmiałek

## Definicja gramatyki

**Gramatyka G to uporządkowana czwórka G=(Σ, Γ, S, P), gdzie:**

- Σ to zbiór symboli terminalnych (symboli alfabetu, z których składają się napisy języka L(G))
- Γ to zbiór symboli nieterminalnych (zmiennych składniowych), przy czym $\Sigma \cap \Gamma = \varnothing$
- S jest symbolem startowym należącym do zbioru Γ
- P jest zbiorem tzw. produkcji (reguł przepisywania)

**Język L(G) zdefiniowany gramatyką G jest zbiorem napisów składających się z symboli terminalnych. Napisy te można wyprowadzić z symbolu startowego przy pomocy szeregu „przepisań" zdefiniowanych przez zbiór produkcji.**

**Zbiór symboli nieterminalnych stanowi zbiór „zmiennych" umożliwiających zdefiniowanie reguł przepisywania (produkcji), czyli zdefiniowanie składni języka.**

Języki formalne i kompilatory © by Michał Śmiałek

## Produkcje

**Produkcja to uporządkowana para napisów (s, t), należących do zbioru (Σ ∪ Γ)*. Produkcje zapisujemy z postaci s → t. Napis ten oznacza, że napis „s" można przepisać w napis „t".**

**Generowanie (lub sprawdzanie poprawności składniowej) napisu języka z symbolu startowego S polega na stosowaniu kolejnych produkcji zawartych w definicji gramatyki. Taki ciąg produkcji nazywamy wyprowadzeniem (sekwencją wyprowadzenia) dla tego napisu języka.**

**Napis g jest bezpośrednio wyprowadzalny z napisu p (p=>g) jeżeli istnieje produkcja A → b, taka, że istnieją napisy t,u takie, że p=tAu i g=tbu.**

**Napis g jest wyprowadzalny z napisu p (p=>+g), jeżeli istnieje sekwencja $r_0...r_n$, takich, że p=>$r_0$=> ... =>$r_n$=>g. Dodatkowo, jeśli p=>+g lub p=g to piszemy p=>*g.**




## Przykłady gramatyk

**Jakie języki opisują te gramatyki? Podaj przykładowe wyprowadzenia. Gramatyka pierwsza:**

- Σ = {+, *, (, ), i}
- Γ = {E, T, F}
- S = E
- P = {E → T, E → E+T, T → F,
  T → T*F, F → i, F → (E)}

**Gramatyka druga:**

- Σ = {n, ., +, -, E}
- Γ = {C, F, I, X, S, U}
- S = C
- P = {C → n, C → nF, C → .I, F → .I, F → ES, I → n,
  I → nX, X → ES, S → n, S → +U, S → -U, U → n}

## Gramatyki regularne

**Druga z powyższych gramatyk jest gramatyką regularną.**

**Gramatyka regularna jest to taka gramatyka, dla której każda produkcja jest postaci A → v lub postaci A → vB, gdzie A,B należą do zbioru Γ, a v należy do zbioru Σ (lub do zbioru Σ ∪ λ dla postaci A → v).**

**Pierwsza gramatyka z poprzedniego slajdu jest gramatyką bezkontekstową (patrz następne wykłady). Czy zdefiniowany w ten sposób język da się przedstawić za pomocą gramatyki regularnej?**

**Gramatyki regularne są szczególnym przypadkiem gramatyk bezkontekstowych. Wyróżnia je się dlatego, że metody analizy przy pomocy gramatyk regularnych są znacznie mniej złożone i kosztowne. Niestety, nie wszystkie języki da się w ten sposób opisać.**



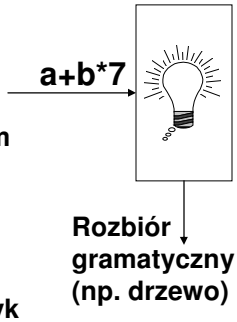

## Gramatyki i automaty

**Analizując zdania jakiegoś języka, kompilator dokonuje rozbioru gramatycznego. Rozbiór ten następuje zgodnie z gramatyką, którą możemy opisać w sposób formalny jak na poprzednich slajdach.**

**Fragment kompilatora dokonujący rozbioru gramatycznego jest pewnego rodzaju automatem rozpoznającym zdania.**

**Postulat: opracować technikę automatycznego generowania takich automatów rozpoznających.**

**Rozwiązanie: algorytmy przekształcania gramatyk regularnych w automaty skończone oraz gramatyk bezkontekstowych w automaty ze stosem.**



## Automaty skończone

**Automat skończony to uporządkowana piątka A=(T, Q, R, $q_0$, F), gdzie:**

- Q jest niepustym zbiorem stanów wewnętrznych;
- T (rozłączny z Q) jest zbiorem symboli terminalnych;
- R jest zbiorem produkcji o postaci qt → q', gdzie q i q' należą do Q, a t należy do T;
- $q_0$ jest stanem początkowym należącym do Q;
- F jest zbiorem stanów końcowych należących do Q (jest podzbiorem Q).

**Automat skończony jest maszyną, która czyta wejściowy napis po jednym znaku i zmienia swój wewnętrzny stan po przetworzeniu każdego symbolu (zgodnie ze zbiorem produkcji).**

**Uwaga: dla każdej gramatyki regularnej G istnieje automat skończony A taki, że L(A) = L(G).**




## Przykładowy automat skończony

**Przedstawiony tu automat odpowiada przykładowej gramatyce regularnej pokazanej kilka slajdów temu.**

- Q = {C, F, I, X, S, U, q}; T = {n, ., +, -, E}
- R = {Cn → q, Cn → F, C. → I, F. → I, FE → S, In → q,
  In → X, XE → S, Sn → q, S+ → U, S- → U,
  Un → q}
- $q_0$ = C; F = {q}

**Ten automat (jak również ten z następnego slajdu) został przekształcony z przykładowej gramatyki w sposób automatyczny (istnieje odpowiedni algorytm, którego opis tu pominiemy). Automat ten jest niezbyt przydatny w praktyce, gdyż jest niedeterministyczny...**

**Ważne pytanie:**

- Czy automat skończony (np. taki jak powyżej) da się narysować graficznie?

## Automat deterministyczny

**Automat jest deterministyczny, jeżeli każde wyprowadzenie może być kontynuowane w co najwyżej jednym ruchu.**

**Automat deterministyczny dla tego samego języka co poprzednio:**

- $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$; $T = \{n, ., +, -, E\}$
- $R = \{q_0 n \rightarrow q_1, q_0 . \rightarrow q_2, q_1 . \rightarrow q_2, q_1 E \rightarrow q_3, q_2 n \rightarrow q_4,$
  $q_3 n \rightarrow q_5, q_3 + \rightarrow q_6, q_3 - \rightarrow q_6, q_4 E \rightarrow q_3, q_6 n \rightarrow q_5\}$
- $F = \{q_1, q_4, q_5\}$

**Automaty skończone mogą być przedstawiane graficznie za pomocą grafu skierowanego. W węzłach grafu umieszczamy stany automatu, a na krawędziach - symbole terminalne powodujące odpowiednie zmiany stanów (określone w zbiorze R). Taki graf nazywamy diagramem stanów.**

**Ważne pytanie:**

- Jak zrealizować automat skończony w strukturze kompilatora?

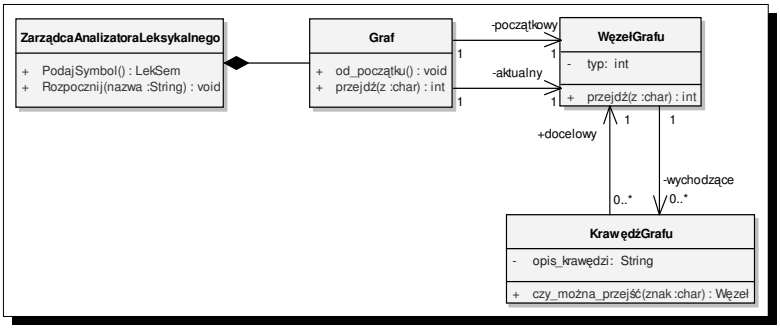



## Struktura automatu skończonego

**Automat skończony ma węzły (stany automatu) i przejścia (produkcje).**

**Ważne:**

- Porównaj strukturę automatu w języku UML z definicją automatu podaną trzy slajdy temu.



## Ćwiczenie

**Proszę narysować jeden lub kilka diagramów sekwencji ilustrujących działanie automatu skończonego. Każdy diagram sekwencji powinien uwzględniać konkretną sekwencję symboli terminalnych na wejściu (np. „0.5E6”).**

**Proszę uwzględnić pobieranie znaków ze strumienia wejściowego i przechodzenie pod wpływem tych znaków przez graf.**

**Proszę krótko opisać diagram.**

**Uwaga: można przyjąć automat pokazany dwa slajdy temu lub inny (własny).**