

Zadanie na pracownię nr 10

Uwaga: Termin oddania rozwiązania to poniedziałek, 14 maja 2018, godz. 6:00.

Parsowanie wyrażeń arytmetycznych

Na wykładzie nr 10 stworzyliśmy kalkulator wyrażeń arytmetycznych (procedura `calc`), który bierze jako argument ciąg znaków (`string`) będący wyrażeniem w tradycyjnej notacji infiksowej, np. `"2+2*(1+1)"`, i oblicza jego wartość. Kalkulator składa się z następujących faz:

1. Ciąg znaków przetwarzany jest na listę znaków przy użyciu wbudowanej procedury `string->list`.
2. Procedura `parse` przetwarza listę znaków na drzewo rozbioru gramatycznego za pomocą gramatyki `arith-grammar`.
3. Procedura `arith-walk-tree` przetwarza drzewo rozbioru gramatycznego na abstrakcyjne drzewo rozbioru (składnię abstrakcyjną) wyrażeń arytmetycznych z **operatorami binarnymi**, która zdefiniowana była na wykładzie 6.
4. Wynik ostateczny obliczany jest przez procedurę `eval`, również przedstawioną na wykładzie 6.

Nasz kalkulator ma dość ograniczone możliwości: może tylko dodawać i mnożyć. Zadanie polega na rozbudowaniu go o **odejmowanie i dzielenie**. Trudnością jest to, że ani odejmowanie, ani dzielenie nie jest łączne, np.

$$(5 - 2) - 1 \neq 5 - (2 - 1)$$

Dlatego bez dodatkowego założenia nie wiadomo, jaka jest wartość wyrażenia $5 - 2 - 1$. Na potrzeby tego zadania – i tu jest trudność – zakładamy, że zarówno odejmowanie jak i dzielenie **łączy w lewo!** Na przykład, wyrażenie `"5-2-1"` należy rozumieć jako `"(5-2)-1"`.

Wskazówka: Zwróć uwagę, że gramatyka mówi:

```
[add-expr {ADD-MANY mult-expr (token #\+) add-expr} ...
```

Oznacza to, że w drzewie rozbioru gramatycznego dodawanie jest naturalnie łączne w prawo. Nie możemy jednak zmienić gramatyki na następującą:

```
[add-expr {ADD-MANY add-expr (token #\+) mult-expr} ...
```

Powodem jest to, że wówczas parser by się zapętlił – zgodnie z tym, co było omawiane na wykładzie.

Jak więc sobie z tym poradzić? Możliwym sposobem jest odpowiednia modyfikacja procedury `arith-walk-tree` tak, by odpowiednio „wygięła” drzewo w drugą stronę. Zauważ, że liczby też wygodniej parsuje się, gdy cyfry w drzewie rozbioru gramatycznego „łączą” w lewo, a nie sprawiło to nam na wykładzie większego problemu.

Przesyłanie rozwiązań

Rozwiązania należy przetestować i zamieścić testy w przesłanym pliku. Rozwiązania powinny być rozszerzeniem pliku `parser-wykl.rkt` dostępnego na SKOS-ie pod hasłem „Parser wraz z przykładami”. Proszę pamiętać, że kalkulator nie zadziała bez pliku `calc.rkt`, również dostępnego na SKOS-ie pod hasłem „Wyrażenia arytmetyczne (z wyeksportowanymi procedurami)”, którego nie należy dołączać do rozwiązania, ani nie modyfikować jego treści.

Rozwiązanie każdego zadania należy przesłać jako plik o nazwie w formacie `nazwisko-imie.rkt`, jak zwykle bez spacji i polskich znaków.