## Sprawozdanie 5 Filip Olszewski I2-L3 136036

**Temat zajęć: Generatory Flex i Bison - ćwiczenia**

Skład repozytorium:

1) Zadania teoretyczne od dr. inż. Jankowskiej

2) Folder zad6c- zadanie 6.1c od dr. inż. Jankowskiej

3) Folder zad6d- zadanie 6.1d od dr. inż. Jankowskiej

* **Zadanie 6.1a**

Zbiór danych jest niepustym zbiorem wierszy, z których każdy składa się z liczb nieujemnych bez znaku:

a) całkowitych zapisanych w układzie dziesiętnym, np. 3256,

b) całkowitych zapisanych w układzie szesnastkowym, np. 43C3,

c) rzeczywistych zapisanych w układzie dziesiętnym w formacie stałopozycyjnym, np. 12.93,

d) rzeczywistych zapisanych w układzie dziesiętnym w formacie zmiennoprzecinkowym,

z cechą koniecznie poprzedzoną znakiem, np. 23.64E+12.

Poszczególne wiersze są niepuste i mają jednorodną postać, tzn. znajdujące się w nich

liczby należą do tej samej kategorii (a, b, c lub d). Każda liczba w wierszu jest

poprzedzona dowolną (niezerową) liczbą spacji lub znaków tabulacji.

Skonstruuj wyrażenie regularne/wzorzec definiujące opisany zbiór danych.

* **Rozwiązanie zadania 6.1a:**

((([ \t]+[1-9]+[0-9]\*)+\n)|

(([ \t]+0x[0-9A-F]+)+\n)|

(([ \t]+(([1-9]+[0-9]\*)|0)\.[0-9]+)+\n)|

(([ \t]+(([1-9]+[0-9]\*)|0)\.[0-9]+E(\+|\-)(([1-9]+[0-9]\*)|0))+\n))+

* Jak to rozumieć:

Liczby dziesiętne:

* [1-9]+ – oznacza jedno lub więcej wystąpień znaków od 1 do 9 ( pierwsza cyfra nie może być 0)
* [0-9]\* – oznacza dowolną ilość znaków z przedziału od 0 do 9 lub całkowity ich brak
* \n – oznacza znak nowej linii
* [ \t]+ – oznacza jedno lub więcej wystąpień tabulacji lub spacji poprzedzających liczbę

Liczby szesnastkowe:

* Liczby w notacji szesnastkowej rozpoczynają się znakiem 0x
* [0-9A-F]+ –oznacza wystąpienie dowolnej liczby znaków z przedziału od 0 do 9 oraz od A do F. Dozwolone są liczby zaczynające się od 0 np. 0x00D5

Liczby stałopozycyjne:

* (([1-9]+[0-9]\*)|0) – oznacza możliwość wystąpienia liczby całkowitej lub zera przezd przecinkiem
* (.[0-9]+)+ – oznacza wystąpienie kropki (jako przecinka) oraz rozwiniecie dziesiętne liczby z przedziału od 0 do 9

Notacja naukowa:

* (([1-9]+[0-9]\*)|0) – oznacza liczbę całkowitą lub 0 przed przecinkiem
* E(\+|\-) – oznacza wystąpienie litery E a za nią znaku + lub –
* (([1-9]+[0-9]\*)|0)) – cyfra lub cyfry występujące po literze E ze znakiem
* **Zadanie 6.1b**

Zdefiniuj gramatykę bezkontekstową generującą zbiór zbiorów danych scharakteryzowanych w zadaniu 6.1a.

* **Rozwiązanie zadania 6.1b:**

Symbol startowy: PLIK

Symbole nieterminalne:

* NOWA\_LINIA >> ‘\n’ | ‘\r’ ‘\n’
* SPACJA >> ‘ ‘
* TAB >> ‘\t’
* KROPKA >> ‘.’
* PLUS >> ‘+’
* MINUS >> ‘-‘
* ZERO >> ‘0’
* CYFRA >> ‘1’ | ‘2’ | ‘3’ | ‘4’ | ‘5’ | ‘6’ | ‘7’ | ‘8’ | ‘9’
* ZNAK >> ‘A’ | ‘B’ | ‘C’ | ‘D’ | ‘E’ | ‘F’
* E >> ‘E’
* X >> ‘x’
* ODSTEP >> TAB | SPACJA | ODSTEP TAB | ODSTEP SPACJA
* PLUS\_MINUS >> PLUS | MINUS
* CYFRA\_ZERO >> CYFRA | ZERO
* CALKOWITA >> CYFRA | CALKOWITA CYFRA | CALKOWITA ZERO
* SZESNASTKOWA\_CYFRA >> CYFRA | ZNAK | SZESNASTKOWA\_CYFRA ZNAK | SZESNASTKOWA\_CYFRA CYFRA\_ZERO |
* SZESNASTKOWA >> ZERO X SZESNASTKOWA\_CYFRA
* CZ\_DZIESIETNA >> ZERO | CALKOWITA
* ROZWINIECIE >> ZERO | CYFRA | ROZWINIECIE ZERO | ROZWINIECIE CYFRA
* STALO >> CZ\_DZIESIETNA KROPKA ROZWINIECIE
* ZMIENNO >> CZ\_DZIESIETNA KROPKA ROZWINIECIE E PLUS\_MINUS CZ\_DZIESIETNA
* PLIK >> WIERSZ | PLIK NOWA\_LINA WIERSZ
* WIERSZ >> WIERSZ\_CALKOWITA | WIERSZ\_SZESNASTKOWA | WIERSZ\_STALO | WIERSZ\_ZMIENNOPRZECINKOWA
* WIERSZ\_CALKOWITA >> ODSTEP CALKOWITA | WIERSZ\_CALKOWITA ODSTEP CALKOWITA
* WIERSZ\_SZESNASTKOWA >> ODSTEP SZESNASTKOWA | WIERSZ\_SZESNASTKOWA ODSTEP SZESNASTKOWA
* WIERSZ\_STALOPRZECINKOWA >> ODSTEP STALOPRZECINKOWA | WIERSZ\_STALOPRZECINKOWA ODSTEP STALOPRZECINKOWA
* WIERSZ\_ZMIENNOPRZECINKOWA >> ODSTEP ZMIENNOPRZECINKOWA | WIERSZ\_ZMIENNOPRZECINKOWA ODSTEP ZMIENNOPRZECINKOWA

Symbole terminalne:

‘0’; ‘1’ ‘2’; ‘3’; ‘4’; ‘5’; ‘6’; ‘7’; ‘8’; ‘9’; ‘A’; ‘B’; ‘C’; ‘D’; ‘E’; ‘F’; ‘X’; ‘+’; ‘-’; ‘ ’; ‘\t’; ‘.’; ‘\n’; ‘\r’;

* **Zadanie 6.1c**

Skonstruuj w języku Bison przetwornik weryfikujący poprawność budowy przedmiotowego zbioru danych i generujący na wyjściu ciąg liczb, w odpowiednich formatach, będących sumami liczb przechowywanych w kolejnych wierszach zbioru.

* **Zadanie 6.1d**

Załóż zmodyfikowaną postać zbioru z liczbami. Modyfikacja polega na jawnym umieszczeniu we wszystkich wierszach – numerów tych wierszy, od numeru 1 poczynając, np.

1 34 891 2

2 23 7A1 C45 90

3 34.9

4 23.4E+22 0.1E-3

Skonstruuj w języku Bison przetwornik weryfikujący poprawność budowy przedmiotowego zbioru danych i generujący na wyjściu ciąg par liczb, z których pierwsza oznacza numer wiersza, a druga - sumę liczb przechowywanych w odpowiednim wierszu zbioru.

* **Opis działania zadania 6.1b/ 6.1d**

Głównym zadaniem tych ćwiczeń było połączenie funkcji programów flex i Bison. Dzięki nim można było podzielić zadanie na analizę leksykalną oraz analizę gramatyczną. Spełniają one dwie różne lecz korelujące ze sobą funkcje. Dzięki analizie leksykalnej mogliśmy przeanalizować tekst z zadania 6.1a i co za tym idzie, wyszukać tokeny i przekazać je do narzędzia Bison.

Analiza gramatyczna bazuje na zadaniu 6.1b. Jej zadaniem jest scharakteryzowanie sposobu w jaki tokeny powinny występować w składni. Po uruchomieniu programu jego zadaniem jest zsumowanie wartości wszystkich elementów jednego typu. Następnie wynik wyświetlany jest na wyjściu w konsoli.

Dodatkowo w zadaniu 6.1d uwzględniany jest numer linii w której znajduje się dany typ danych wejściowych. Jest on reprezentowany jako token INDEX, w momencie kiedy następuje nieprawidłowe rozpoznanie linijki zwracany jest token UND.