

# Specyfikacja języka

Szkic koncepcyjny 14.05.2011

Jakub Niewczas

## **O czym jest ten dokument**

Dokument jest szkicem specyfikacji języka, którego interpreter stanowić będzie projekt zaliczeniowy przedmiotu AUG. Dokument może ulegać zmianom.

## **Wstęp**

Zainspirowany *badaniami Wojciecha Gwizdały*<sup>1</sup> nad wykorzystywaniem kolorów w budowie intuicyjnych interfaców użytkownika w urządzeniach mobilnych, postanowiłem stworzyć język w którym kolory stanowią pełnoprawną płaszczyznę programowania. Język zakłada, że kod źródłowy programu jest dokumentem tekstowym, w którym kolory pełnią funkcje identyfikowania sygnatur zmiennych oraz funkcji.

## **Założenia**

Realizacja projektu ma na celu poszerzenie wiedzy o budowie kompilatorów poprzez implementację języka opisanego w dokumencie. Język implementuje podstawowe operacje arytmetyczne, definiowanie zmiennych, funkcji, pętli oraz instrukcję wyświetlającą na ekran. Założenia języka nie przewidują optymalizacji składni języka pod względem przejrzystości i intuicyjności kodu.

## **Opis języka**

W języku istnieją tylko dwa typy: string oraz int. Wartości zmiennych stringowych definiowane są czarnym kolorem w cudzysłowie, a nazwy zmiennych stringowych określane są znakiem %, kolor znaku jest sygnaturą zmiennej. Wartości zmiennych liczbowych są określone kolorem znaku &, sygnatury zmiennych liczbowych to znak # odpowiedniego koloru.

przypisanie napisu do **czerwonej** zmiennej stringowej:

```
% = "hello"
```

przypisanie **niebieskiej** liczby ( hex #0000ff; dec 255 ) do **czerwonej** zmiennej liczbowej:

```
# = &
```

przypisanie wyniku dodawania<sup>2</sup> liczby **pomarańczowej** z **zieloną**:

```
# = & + &
```

pod zmienną **#** przechowywana jest teraz **liczba**.

Wywołanie funkcji odbywa się poprzez przekazanie identyfikatora funkcji kolorem znaku „@” a następnie wypisaniu listy argumentów bez mieszania kolorów, tak więc na przykład wyświetlaniem na ekran zajmuje się **żółta** funkcja, która jako jedyny parametr przyjmuje string, mający zostać wypisany na wyjście.

```
@ "Hello World"
```

Funkcje definiuje się za pomocą słowa *def* a kończy słowem *end*, kolor słowa kluczowego *def* oraz *end* będzie nazwą funkcji. Funkcja wyświetlająca napis:

```
def %, #  
  @ "Nazywam sie" % " i lubie cyferke" #  
end
```

Wywołanie funkcji:

```
@ "Kuba" #
```

Pętla while realizowane są poprzez słowo kluczowe *while*, kolor słowa kluczowego jest bez znaczenia.

```
while # < &  
  # = # + &  
end
```

### **Przypisy**

<sup>1</sup> strona Wojciecha Gwizdały [http://gwizdala.neostrada.pl/rgb\\_translator/rgb.html](http://gwizdala.neostrada.pl/rgb_translator/rgb.html)