
GenuineLanguage [GLang]

Instrukcja do języka

Autorzy

Bartłomiej Kopyś

Filip Olejniczak

Contents

1	Opis języka	2
2	Zmienne	2
2.1	Typy	2
2.2	Deklarowanie	3
2.3	Przypisywanie wartości	3
3	Operacje wejścia - wyjścia	3
3.1	Odczyt danych	3
3.2	Wypisywanie danych	3
4	Operacje matematyczne	3
5	Bloki warunkowe if/else	4
6	Funkcje	4
6.1	Definiowanie funkcji	4

1 Opis języka

Język GLang jest silnie typowanym językiem przygotowanym jako prototyp. Oferuje wiele możliwości między innymi:

- Deklarowanie zmiennych
- Podstawowe operacje matematyczne
- Operacje wejścia - wyjścia
- Definiowanie funkcji
- Bloki warunkowe if/else
- Pętle

2 Zmienne

2.1 Typy

Dostępne są dwa rodzaje typów:

- int - typ liczby całkowitej
- real - typ liczby rzeczywistej

2.2 Deklarowanie

Liczby można definiować w następujący sposób: `<typ> <nazwa zmiennej> ;`

Przykład

```
int a ;
```

2.3 Przypisywanie wartości

Wartość do zadeklarowanej zmiennej może zostać przypisana od razu przy deklaracji lub po.

Przykład

- `<typ> <nazwa zmiennej> = <wartość> ;`
- `<nazwa zmiennej> = <wartość> ;`

3 Operacje wejścia - wyjścia

Jezyk obsługuje możliwość wczytywania i wypisowania danych w linii komend.

3.1 Odczyt danych

Odczyt danych następuje przez użycie operacji `read`. Wymagane jest by zmienna została już wcześniej zadeklarowana.

Przykład:

```
<typ> <nazwa zmiennej> ;  
read <nazwa zmiennej> ;
```

3.2 Wypisywanie danych

W celu wypisania danych na wyjście należy wykorzystać operację `print`. Wymagane jest by zmienna została już wcześniej zadeklarowana.

Przykład:

```
<typ> <nazwa zmiennej> ;  
write <nazwa zmiennej> ;
```

4 Operacje matematyczne

Jezyk definiuje podstawowe metody matematyczne. Jest możliwość dokonywania operacji na zmiennych z użycie innych zmiennych lub wartości liczbowych.

Są dostępne następujące operacje:

- Dodawanie: `+`

- Odejmowanie: -
- Dzielenie: /
- Mnożenie: *

Przykład:

`<zmienna> = <wartość> <operator> <wartość>`

5 Bloki warunkowe if/else

Jezyk oferuje możliwość deklarowania bloków warunkowych. W tym celu jest używane wyrażenie warunkowe, które definiuje który blok programu zostanie wykonany. Blok musi być zakończony słowem kluczowym `end`. Możliwe są dwa kombinacje deklarowania bloków:

- `if <wartość> <operator warunkowy> <wartość> do`
`<blok>`
`end`
- `if <wartość> <operator warunkowy> <wartość> do`
`<blok>`
`else`
`<blok>`
`end`

Dostępne są następujące operatory warunkowe:

- Mniejsze: <
- Mniejsze bądź równe: <=
- Większe: >
- Większe bądź równe: >=
- Równe: ==
- Nierówne: !=

6 Funkcje

Jezyk pozwala na definiowanie i używanie funkcji.

6.1 Definiowanie funkcji

Funkcje są deklarowane przez słowo kluczowe `function`. Musi zawsze zostać podany typ zwracany przez funkcję. Dodatkowo przy deklaracji muszą zostać podane typy argumentów i same argumenty. Cały blok funkcji kończony jest słowem kluczowym. Zmienne w globalnym kontekście nie są dostępne wewnątrz funkcji. `end`.

Przykład:

```
function <typ> <nazwa funkcji> (<typ argumentu> <argument>, ...) do  
<blok funkcji>  
end
```