

# Jazyk S1mpl3

---

SEMESTRÁLNÍ PRÁCE Z PŘEDMĚTU KIV/FJP

DAVID MARKOV, JAROSLAV HRUBÝ

# Datové typy

---

`int` – celé číslo

`bool` – pravdivostní hodnota

`int[<velikost>]`, `bool[<velikost>]` – pole hodnot

# Práce s datovými typy

---

## Inicializace

- `<datový_typ> <identifikátor>;`
  - `int a;`

## Přřazení

- `<datový typ> <identifikátor> = <hodnota>;`
  - `int a = 10;`
- `<identifikátor> = <hodnota>;`
  - `a = 10;`
  - `a = b;`

## Konstanta

- `const <datový typ> <identifikátor> = <hodnota>;`

## Přístup do pole

- `<identifikátor>[<index>]`

# Práce s datovými typy

---

## Aritmetické operace

- $(+ a, b)$
- $(- a, b)$
- $(* a, b)$
- $(/ a, b)$  + modulo

## Logické operace

- $(\text{AND } a, b)$
- $(\text{OR } a, b)$
- $(!( < a, b))$

# Jazykové konstrukce

---

Názvy identifikátorů:

- Malá a velká písmena

Cykly:

- `while <podmínka> { <tělo> }`
- `do { <tělo> } while <podmínka>;`
- `for <identifikátor> in <pole_hodnot> { <tělo> }`
- `for <identifikátor> in <horní_hranice> { <tělo> }`

Podmínky:

- `if <podmínka> { <tělo> } else { <tělo> }`
- `switch (<identifikátor>) { case: <hodnota> { <tělo> }  
default: { <tělo> } }`

Ternární operátor:

- `(<a, b> ? a : b;`

# Jazykové konstrukce

---

Definice podprogramů:

```
func <návratová_hodnota> <název_podprogramu>(<parametry>) { <tělo> }
```

Volání podprogramů:

```
(<název_podprogramu> <parametry>)  
(instanceof <identifikátor>, <typ>)
```

Funkce pro výpis/čtení: - **nebude**

```
(read <identifikátor>)  
(write <identifikátor>/<hodnota>)
```

# Další vlastnosti

---

- Cílovou platformou PL/0
  - Kontrola přístupu na indexy pole tam, kde je to možné
  - Implicitní konverze int na bool a zpět
  - Žádná explicitní konverze
  - Case bloky nebudou propadávat
- 
- Pro implemetaci lexeru a parseru využití ANTLR

Děkujeme za pozornost

---