

Informační a komunikační technologie

Ošetřování vstupu

David Weber

Kabinet K13

weber3@spsejecna.cz

Jak to bylo doposud...

- Zatím jsme předpokládali inteligentního uživatele.
- Tzn. uživatel zadával pouze povolené hodnoty.
- **V realitě uživatel však nemusí zadat požadovanou hodnotu!**

Příklad na začátek

```
int value;  
scanf("%d", &value);  
  
if (value == 0) {  
    printf("1");  
    return 0;  
}  
  
int fact = 1;  
for (int i = 1; i <= value; i++) {  
    fact *= i;  
}  
  
printf("%d", fact);  
return 0;
```

Příklad na začátek

- Co bylo na tomto programu špatně?
- \Rightarrow uživatel nemusí zadat **nezáporné celé číslo!**

Jak toto řešit?

- Před každým programem je třeba si promyslet, které vstupy jsou přípustné.
- U předešlé úlohy by se hodilo ošetřit, zda je hodnota na vstupu přípustná.
- **Můžeme např. na začátek programu přidat:**

```
if (n < 0) {  
    printf("CHYBNY VSTUP!");  
    return 0;  
}
```

Problémy nekončí...

- Uživatel nemusí zadat nic nebo např. znak.
- Toto již pomocí podmínek lze jen těžko odchytit. 😬
- \Rightarrow funkce `scanf` má však tzv. **návratovou hodnotu!**

```
int value;  
int paramCount = scanf("%d", &value);  
  
printf("%d", paramCount);
```

- V proměnné `paramCount` je uložen počet úspěšně načtených parametrů.

Úprava programu

- V případě program s faktoriálem uživatel zadává **jeden parametr**.

```
int value;  
int paramCount = scanf("%d", &value);  
  
if (paramCount < 1){  
    printf("CHYBNY VSTUP!");  
    return 0;  
}
```

Otázky?

