

CYKLY

Cyklus nebo také smyčka je řídicí struktura programu, která umožňuje opakovat určitou posloupnost příkazů vícekrát. Podle toho, zda počet opakování je nebo není známý před začátkem cyklu, rozlišujeme cyklus s parametrem a cyklus s podmínkou.

1 CYKLUS FOR

Cyklus for používáme v případě, že předem víme, kolikrát se bude daná posloupnost příkazů provádět. Cyklus for má strukturu:

```
for citac = od : do  
    příkazy  
end
```

nebo

```
for citac = od : krok : do  
    příkazy  
end
```

Cyklus for se skládá z klíčového slova for, čítače, tj. proměnné, která určuje počet opakování, těla cyklu a klíčového slova end. Hodnota čítače se nejprve nastaví na hodnotu od a provede se tělo cyklu. Pokud je definován krok, zvýší se hodnota čítače o krok, jinak se zvýší o jedna. Znovu se provede tělo cyklu. Po každém provedení těla cyklu se hodnota čítače zvýší o krok nebo o jedna. Cyklus se provádí tak dlouho, dokud není hodnota čítače rovna nebo větší než hodnota do. Poté se naposledy provede tělo cyklu.

Jako příklad si uvedeme cyklus

```
for i = 1 : 5  
    i^2  
end
```

Výstupem bude

```
1  
4  
9  
16  
25
```

2 CYKLUS WHILE

Cyklus while je cyklus řízený podmínkou. Tělo cyklu se provádí, dokud je splněna určitá podmínka. Cyklus while má strukturu:

```
while podmínka
    příkazy
end
```

V tomto případě tedy není definován žádný čítač. Podmínka je výraz s logickou hodnotou 1 (true) nebo 0 (false). Nejprve se daná podmínka otestuje. Pokud je splněna, provedou se příkazy těla cyklu. Potom se podmínka opět otestuje a v případě, že je splněna, znovu se provedou příkazy těla cyklu. Takto cyklus pokračuje, dokud se nezjistí, že podmínka není splněna. Potom cyklus skončí.

Jako příklad si uvedeme cyklus

```
i = 1;
while i < 5
    2 * i
    i = i + 1;
end
```

Výstupem bude

```
2
4
6
8
```

V podmínce se často používají relační operátory pro porovnávání proměnných a logické operátory pro tvorbu složené podmínky.

RELAČNÍ OPERÁTORY			
==	rovná se	<	menší než
~=	nerovná se	>=	větší rovno
>	větší než	<=	menší rovno

LOGICKÉ OPERÁTORY	
$A \& B$, $and(A, B)$, $A \& \& B$	A a zároveň B
$A B$, $or(A, B)$, $A B$	A nebo B
$xor(A, B)$	buď A nebo B
$\sim A$, $not(A)$	negace A

V podmínce mohou být také použity funkce, které vrací hodnotu 1 nebo 0. Například funkce *isnumeric(d)* určuje, zda d je číselná proměnná. Vrací hodnotu 1, pokud d je číslo, jinak vrací 0. Funkce *isreal(d)* určuje, zda d je reálná proměnná. Vrací hodnotu 0, pokud d má nenulovou imaginární část, jinak vrací 1.

Výpočet lze přerušit pomocí stisku kláves Ctrl a C.