

11. Algoritmizace - řízení programu

Created	@April 26, 2025 7:11 PM
Tags	Done
Kdo vypracoval	Michal

Algoritmus Definice

Přesně definovaná posloupnost kroků nebo pravidel, která vedou k řešení daného problému. Musí být jednoznačný a po konečném počtu kroků dospět k výsledku.

Větvení

Podmínky na základě kterých se program rozdělí do více větví.

Zakladní podmínka if else

```
age = 18
if age >= 18:
    print("Můžeš řídit auto.")
else:
    print("Ještě nemůžeš řídit.")
```

Podmínka s elif

Na podmínu elif dojde jen, pokud je výsledek predchozí podmínky False.

```
age = 20

if age < 12:
    print("Jsi dítě.")
elif age < 18:
    print("Jsi teenager.")
elif age < 60:
```

```
    print("Jsi dospělý.")  
else:  
    print("Jsi senior.")
```

Cykly

Podmínka na začátku

Podmínka while. Opakuje kód, dokud je podmínka True. Podmínka je na začátku, potom kód.

```
x = 0  
while x < 5: # Podmínka na začátku  
    print(x)  
    x += 1
```

Podmínka na konci

Podmínka na konci v pythonu není předepsána, musí se udělat pomocí while cyklu.

```
x = 0  
while True: # Cyklus běží nekonečně, dokud se neukončí  
    print(x)  
    x += 1  
    if x >= 5: # Podmínka na konci  
        break
```

Přesný počet opakování

Cyklus for. Proběhne presný počet opakování.

```
for i in range(5): # Cyklus s přesným počtem opakování  
    print(i)
```

Prohledání 2D pole pomocí cyklu for. Jeden je vnořený do druhého.

```
matrix = [  
    [1, 2, 3],
```

```
[4, 5, 6],  
[7, 8, 9]  
]  
  
# Prohledání 2D pole pomocí indexů  
for i in range(len(matrix)): # Cyklus procházející řádky (indexy)  
    for j in range(len(matrix[i])): # Cyklus procházející sloupce (indexy)  
        print(f"Prvek na pozici ({i}, {j}): {matrix[i][j]}")
```

Break

Ukončí cyklus a už nepokračuje v dalších interacích.

```
for i in range(5):  
    if i == 3:  
        break # Jakmile i == 3, cyklus se ukončí  
    print(i)
```

Continue

Když se provede continue, Python nepokračuje v kódu níže v této iteraci a hned jde na další opakování cyklu. Cyklus se neukončí.

```
for i in range(5):  
    if i == 3:  
        continue # Přeskočí výpis čísla 3 a pokračuje dál  
    print(i)
```

Funkce

Blok kódu, který lze opakovaně použít. Pomáhá organizovat program a zabránit opakování kódu.

```
def pozdrav_jmeno(jmeno):  
    print(f"Ahoj, {jmeno}!")  
  
pozdrav_jmeno("Petr")
```

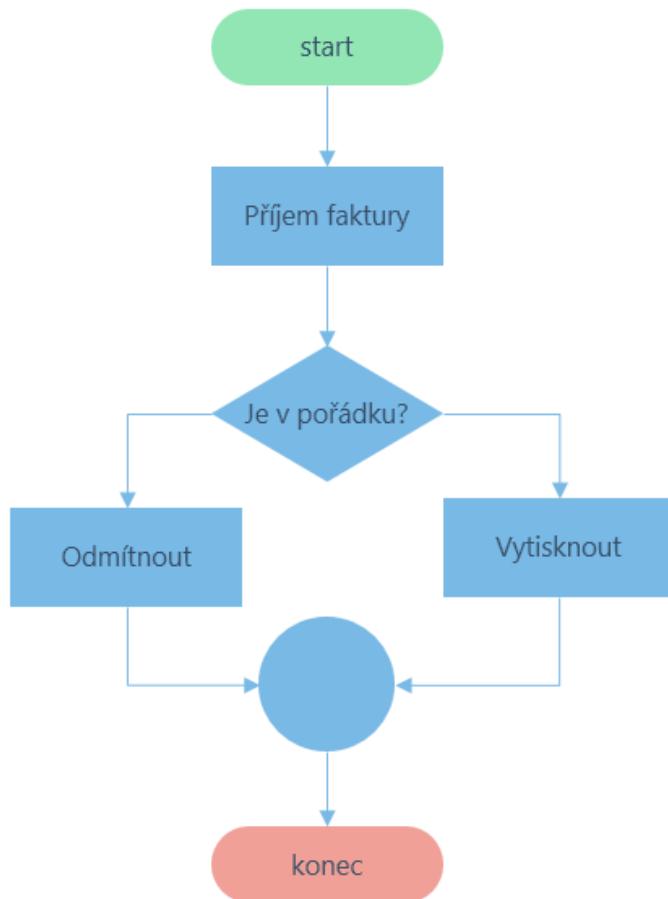
Vývojové diagramy

Diagramy, které znázorňují jednotlivé kroky procesu. Práce s vývojovým diagramem nevyžaduje znalost programovacího jazyka.

Základní symboly

Symbol	Význam
● Ovál	Začátek/Konec programu
◆ Kosočtverec	Podmínka (rozhodování: ano/ne)
◻ Obdélník	Proces (výpočet, přiřazení, operace)
→ Šipky	Tok programu (směr vykonávání)

Příklad vývojového diagramu



Je tady toho mén, Jedlička říkal, že zbytek času bude nějaký praktický příklad.