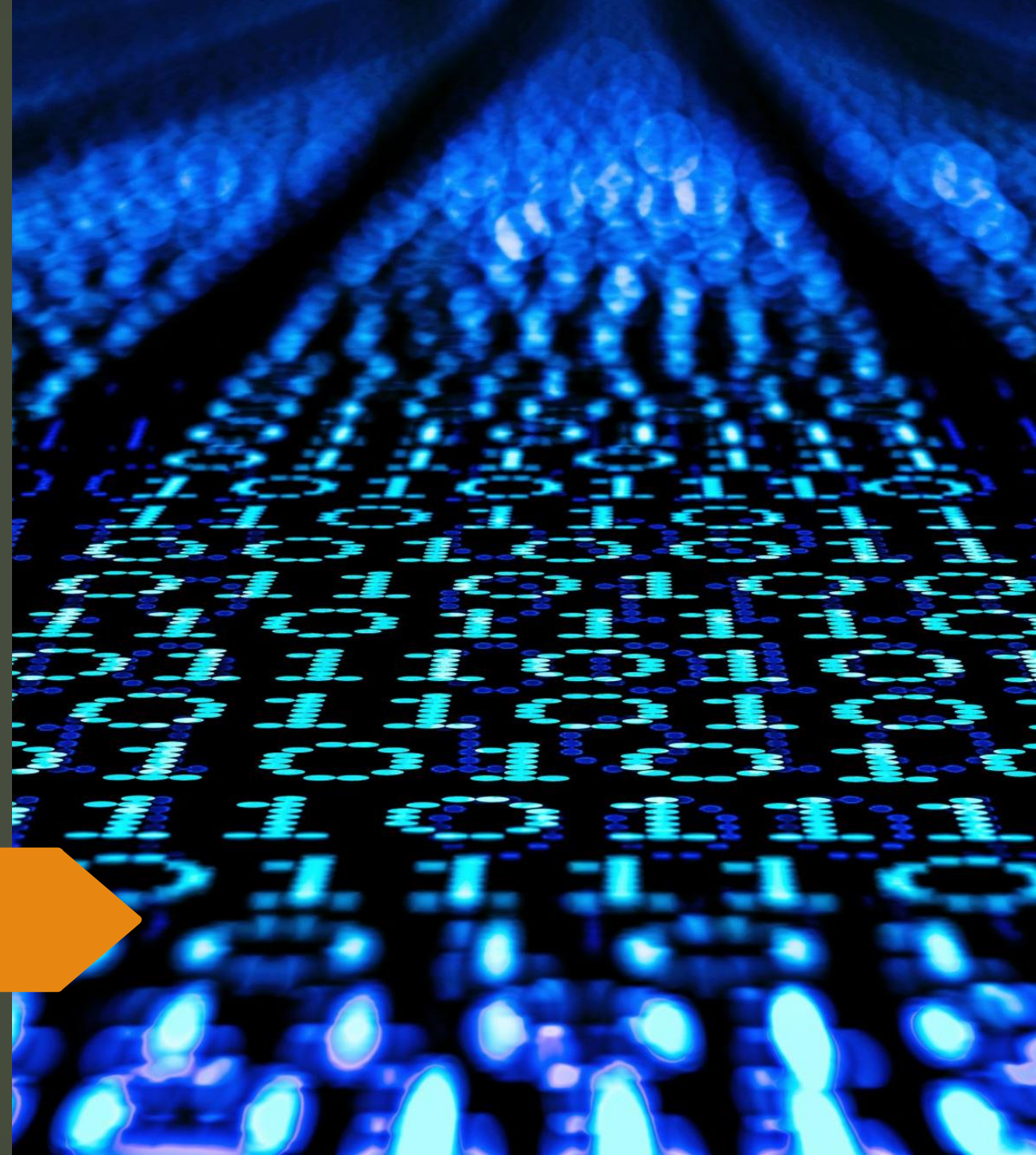


Programování

Základní pojmy a definice v programování



Základní pojmy

Algoritmus

Vlastnosti
algoritmu

Program

Programovací
jazyk

Proměnná

Konstanta

Datový typ

Sekvence

Větvení

Cyklus

Podprogram



Algoritmus



- ▶ = jasný popis postupu pro řešení konkrétního problému / úlohy
- ▶ Lze jej popsat různými způsoby:
 - ▶ Graficky - diagramem
 - ▶ Slovně (recept)
 - ▶ Matematická rovnice
 - ▶ Pseudokód
 - ▶ Implementací
- ▶ Obvykle slouží pro teoretické řešení problému



Vlastnosti algoritmu

- ▶ **Hromadnost**

- ▶ Řešení celé, přesně vymezené třídy problémů, které se liší pouze vstupními hodnotami

- ▶ **Determinovanost/determinismus**

- ▶ Každý krok je přesně definován, víme co má následovat a kdy tak má nastat
- ▶ Při stejných podmínkách algoritmus vrátí vždy stejné výsledky pro stejné vstupy

- ▶ **Konečnost**

- ▶ Algoritmus je ukončen v konečném počtu elementárních kroků

- ▶ **Rezultativnost**

- ▶ Algoritmus má minimálně jeden výstup, který je očekávaným výsledkem



Program

- ▶ = zápis algoritmu v konkrétním programovacím jazyce
- ▶ = posloupnost instrukcí, zpracovávaná počítačem, která řeší konkrétní úlohu/problém



Programovací jazyk

- ▶ = umělý jazyk sloužící, pro definování sekvence příkazů, které lze zpracovat počítačem
- ▶ Slouží pro implementaci programu, řešící algoritmus
- ▶ Dělní programovacích jazyků:
 - ▶ **Míra abstrakce**
 - ▶ Vyšší, nižší
 - ▶ **Způsob překladu**
 - ▶ Kompilované, interpretované

Proměnná

- ▶ = pojmenované místo v paměti, jehož obsahem je konkrétní hodnota
- ▶ Pro potřebu práci s proměnnou se na ni odkazujeme jejím pojmenováním
- ▶ Při pojmenování dodržujeme konvence daného programovacího jazyku
- ▶ **Lokální proměnná**
 - ▶ Proměnná přístupná pouze pro určitou část programu (blok kódu, funkce, třída)
- ▶ **Globální proměnná**
 - ▶ Proměnná přístupná z kteréhokoliv místa v programu
- ▶ Pro vložení hodnoty do proměnné využíváme přiřazovací operátor =

PROMĚNNÁ = HODNOTA



Datový typ

- ▶ = specifikace určující, jaký typ hodnoty je uložen v proměnné
- ▶ Datový typ je kontrolován při vkládání hodnoty do proměnné
- ▶ Nelze vkládat do proměnné datového typu **A** hodnoty typu **B**
- ▶ **Číselné datové typy**
 - ▶ short, int, double, float, long
- ▶ **Znakové**
 - ▶ char
- ▶ **Pravdivostní**
 - ▶ bool

Konstanta

- ▶ = typ proměnné, jehož hodnota je po celou dobu programu neměnná
- ▶ Do tohoto typu proměnné nelze vkládat hodnotu
- ▶ Její hodnota je definována již při její deklaraci
- ▶ Příkladem konstanty může být: Ludolfovo číslo, Euklidovo číslo, hodnota gravitačního zrychlení, ...
- ▶ Deklarace konstanty v jazyce C#

```
double const KONSTANTNI_HODNOTA = 1.23456789;
```

Sekvence

- = lineární posloupnost příkazů, tvořící program
- Pořadí těchto příkazů je i pořadím vykonávání instrukcí
- Nejjednodušší typ algoritmů – neobsahuje žádné cykly ani větvení

K zamyšlení: Prohodte hodnoty v proměnných **A** a **B** bez použití pomocné proměnné





Větvení

- ▶ = rozdělení chování algoritmu z důvodu více možností, jak dále pokračovat
- ▶ Lze jej použít pro ošetření nevalidních stavů v algoritmu
- ▶ Pro větvení využíváme podmínku, která rozhodne o dalším průběhu algoritmu
- ▶ Podmínka musí být rozhodnutelná – výsledek je ano / ne (bool)
- ▶ V programování lze větvení reprezentovat pomocí konstrukcí:
 - ▶ **If-else**
 - ▶ **Switch-case**



Cyklus

- = část algoritmu (blok kódu), která se opakuje
- Dělení cyklů:
 - Cykly s pevným počtem opakování (**for**, **foreach**)
 - Cykly s neznámým počtem opakování
 - Cykly řízené podmínkou na konci (**do-while**)
 - Cykly řízená podmínkou na začátku (**while**)
- Podmínky v cyklech mají stejné náležitosti, jako podmínky u větvení



Podprogram

- ▶ = malá celistvá část programu, kterou lze opakovaně volat z různých míst
- ▶ Využíváme pro redukci duplicit částí kódu
- ▶ = funkce
- ▶ Funkce mohou mít vstupní parametry (data se kterými má pracovat)
- ▶ Tyto podprogramy mohou vracet na základě vstupů vracet konkrétní hodnoty
- ▶ Hlavička funkce v jazyce C# musí obsahovat
 - ▶ Viditelnost, návratový datový typ, název, seznam vstupních parametrů (může být prázdný)