



Program a programování a utrpení

Neumím programovat a Jáchym taky ne

Eric a Kuba

Gymnázium Evolution Jižní Město

18. května 2025

1. Počítačový program

2. Programovací jazyky

2.1 Proceduranální vs. funkcionální jazyky

2.2 Způsob překlada

Počítačový program (aspoň trochu do hloubky (jako jáchymovy špagety))

- Počítačový program je sada nebo posloupnost instrukcí napsaných ve specifickém programovacím jazyku („řeč počítače v lidském stylu“) určených pro počítač (on čte ty instrukce), který tento program vykoná.
- Počítačový program musí (nebo chceme aby byl) být **přesný a jednoznačný**.
- Tak se liší od lidské řeči, kde si příjematelé (lidi) mohou domýšlet věci, to ale počítač neumí.

Syntax – Sémantika



Syntax – Sémantika

- Syntax popisuje pravidla zápisu programu (gramatika jazyka).
- Sémantika definuje význam kódu. – logistika a tak...

Příklady programovacích jazyků



- Zotročil jsem AI: link

Procedura vs funkce

- Procedura i funkce jsou bloky kódu (soubor příkazů), které se spustí, když se procedura nebo funkce zavolá.
- Ulehčuje to, když se jeden kus kódu používá velmi opakovaně.
- Mohou mít vstupní hodnoty se kterými pracují.
- Rozdíl mezi procedurou a funkcí je, že procedura žádnou hodnotu nevrací. Funkce naopak ano.

Proceduranální vs. funkcionální jazyky

Proceduální – Python, Rust, C, Java, ...

- Program je sekvence příkazů a funkcí.
- Jsou běžnější v praxi.

Funkcionální – Haskell, F#, Scala, OCaml, Idris, Lean, ...

- Zaměřují se na to, co se má vypočítat, nikoli na to, jak se to má provést.
- Často se používá rekurze
- V ideálním funkcionálním programování jsou data neměnná. To znamená, že po vytvoření hodnoty ji nelze změnit.
- Místo změny dat se vytvářejí nové hodnoty.

Způsob překladu

Kompilované – C, Java, Cobol

- Před spuštěním se strčí do zázračného stroje, který to přeloží do strojového kódu.
- Kompilují se do spustitelného souboru (například .exe)
- Kompilují se do nižších vrstev, například z high level do machine code přes low level.
- Po kompilaci nelze jít zpátky nahoru do vyšších vrstev. – Ztratí se hrozně moc informací o kódu, například všechna sémantika (syntax).
- Programy jsou pak optimalizovanější.
- Kód je „schovaný“, super pro closed source.
- Musí se zkompilovat.

Interpretované – Python, BASIC, Shell

- Překládá za letu \Rightarrow je pomalejší.
- Snadná přenositelnost, rychlé testování.

Source code, machine code, assembly, low-level a high-level jazyky

- Source code (zdrojový kód) – kód napsaný programátorem v čitelném jazyce (např. C, Python).
- Machine code (strojový kód) – binární kód, nejnižší úroveň, CPU instrukce
- Assembly – nízkoúrovňový jazyk blízký strojovému kódu, ale stále čitelnější pro člověka. Asembler se pak používá pro překlad do vrstvy strojového kódu.
- Low-level jazyky – blízko hardware, např. C, Rust. Používají ukazatele (pointers), což jsou adresy v paměti, které umožňují přímou manipulaci s daty.
- High-level jazyky – blíže lidské logice, např. Python, Java. Mají automatickou správu paměti (garbage collector), který čistí nevyužitou paměť, aby programátoři nemuseli ručně spravovat alokaci a dealokaci.

Pokračování s důležitými pojmy

- Garbage collector – automaticky uvolňuje paměť, kterou nepoužíváme.

Video co vše vysvětluje



link



Děkujeme za pozornost

Eric a Kuba

Gymnázium Evolution Jižní Město

18. května 2025