2. Algoritmus, algoritmická složitost

Algoritmus, algoritmická složitost, rekurze, náhodnost.

## Algoritmus

* Přesný postup řešení problému, úkolu
* Určitý počet definovaných kroků
* Vlastnosti
* Vývojový diagram, slovní popis

### Vlastnosti algoritmů

#### Resultativnost

* + Musí dát výsledek (i chybu)
  + Jinak je zbytečný

#### Konečnost

* + Jednou stejně skončí
  + Chyba, konec
  + Jinak hrozí zacyklení a vyčerpání paměti

#### Elementárnost

* Každý krok algoritmu musí být srozumitelný, jednoznačný – pc musí rozumět
* PC tu chybu stejně udělá

#### Determinovanost

* Za stejných vstupů stejné výstupy
* Každý krok algoritmu jednoznačně definován

#### Obecnost

* Neřeší konkrétní příklad, ale celou řadu

## Algoritmická složitost

* Porovnání kvality algoritmů

### Časová složitost

* Podle doby trvání algoritmu
* Značí se O()
* Počítají se lementární operace v závislosti na velikosti vstupu

### Paměťová složitost

* Podle využívané paměti

## Rekurze

* Funkce volá sama sebe
* Volána dříve než se dokončí její první volání
* Paměťově náročná – hrozí vyčerpání paměti – zacyklení
* Elegantní ale ne vždy výhodné řešení

### Přímá

Funkce volá sama sebe

### Nepřímá

Funkce A volá funkci B a naopak – kruh

### Lineární rekurze

Program volá sám sebe pouze jednou, lineární struktura postupně volaných podprogramů

### Stromová rekurze

* Strom
* Podprogram se volá více jak jednou v rámci úkolu

### Výhody / nevýhody rekurze

* Každé volání zvyšuje hloubku zanoření fce a žere paměť a čas
* Každé volání = nové uložení lokálních proměnných do zásobníku, předání parametrů, návratové adresy a skok do podprogramu

#### Výhody

* Elegantní přehledné matematické řešení

## Náhodnost

* Událost u které nedokážeme předpokládat výsledek
* Obtížné na pc
* Opravdová náhodnost je potřeba u šifrování a vytváření složitých hesel

### Čísla náhodná – TRNG

* True random number generator
* Založeno na mikroskopických jevech
* Generace nízkoúrovňových, staticky náhodných šumových jevech
* Tepelný šum, fotoelektrický jev

### TPM

* Trusted platform module
* Čip – popisuje zabezpečený kryptoprocesor, na který se dají ukládat šifrovací klíče
* Má i RNG (random number generator) – pracuje se šumem, tep. Šumem, prouděním vzduchu…
* Tyto hodnoty jednosměrné zahashuje, čímž zajistí úplnou náhodnost

### Čísla pseudonáhodná

* Stačí nám v programování
* Generují se SW
* RNG vygeneruje random seed – výchozí hodnota
* Z té se další věci vypočítají jako funkční hodnoty nějaké funkce
* Výchozí hodnota – kombinace času, data apod.
* Po dlouhé době opakování
* Snadno odhalitelné