

```
for i:=1 to n
| min:=i;
| for j:=i+1 to n
| | if a[min]>a[j] min:=j;
| tmp:=a[i]; a[i]:=a[min]; a[min]:=tmp;
```

```
function merge(from,mid,to)
| copy a[from..mid] to a new array b
| copy a[mid+1..to] to a new array c
| add infinity to both b and c as the last element
| k:=1; m:=1;
| for j:=from to to
| | if b[k]<c[m] then
| | | a[j]:=b[k]; k++;
| | else
| | | a[j]:=c[m]; m++;
```

```
function sort(from,to)
| if (from>to)
| | mid:=floor((from+to)/2);
| | sort(from,mid); sort(mid+1,to);
| | merge(from,mid,to);
```

```
clear array count[1..1000];
for i:=1 to n
| count[a[i]]++;
k:=0;
for i:=1 to 1000
| for j:=1 to count[i];
| | a[k]:=i; k++;
```

$n =$	Counting sort $O(n)$	Merge sort $O(n \log n)$	Selection sort $O(n^2)$	Exponential algorithm $O(2^n)$
10	ε	ε	ε	ε
50	ε	ε	ε	2 weeks
100	ε	ε	ε	2800 univ.
1000	0.01s	ε	ε	—
10000	0.01s	0.03s	0.27s	—
100000	0.06s	0.15s	26.5s	—
1 mil.	0.74s	1.6s	44.2m	—
10 mil.	7.4s	16.5s	3.1d	—

Označenie	Definícia	Analógia
$f(n) \in O(g(n))$	Existuje $c > 0$ a $n_0 > 0$ také, že $(\forall n > n_0)(0 \leq f(n) \leq cg(n))$	\leq
$f(n) \in \Omega(g(n))$	Existuje $c > 0$ a $n_0 > 0$ také, že $(\forall n > n_0)(f(n) \geq cg(n) \geq 0)$	\geq
$f(n) \in \Theta(g(n))$	$f(n) \in O(g(n))$ a $f(n) \in \Omega(g(n))$	$=$

Efektívne algoritmy a dátové štruktúry, 1-AIN-105

Vyučujú:

Tomáš Vinař, M-163, tomas.vinar@fmph.uniba.sk

Andrej Baláž, M-25, andrej.balaz@fmph.uniba.sk

Sabína Samporová, T-2, sabina.samporova@trojsten.sk

Marcel Palaj, T-2, marcel.palaj@trojsten.dk

Web: <http://compbio.fmph.uniba.sk/vyuka/eads>

Google Classroom: <https://classroom.google.com> kód **5jhwelu**

Literatúra:

Brassard, Bratley: Fundamentals of Algorithmics, Prentice Hall 1995

Cormen, Leiserson, Rivest, Stein: Introduction to Algorithms, MIT

Press 2009

Oznamy a diskusia na Google Classroom

<https://classroom.google.com> kód **5jhwelu**

- **Vaša povinnosť sledovať oznamy!!!**
- Cieľ: odpoveď na vaše otázky v najkratšom možnom čase
- Odpovedajú učitelia, študenti
- Všetky otázky sú verejné \Rightarrow žiadne detaily vlastných riešení (resp. až po termíne odovzdania danej úlohy)
- Link aj na web stránke predmetu

O čom je tento predmet?

- Pre daný problém, úlohou je nájsť **efektívny algoritmus**, ktorý tento problém rieši.
- Rozpoznať, kedy efektívny algoritmus **neexistuje**.
- Urobiť z vás **lepších programátorov**.

Osnova predmetu:

- Úvod, výpočtová zložitosť
- Základné dátové štruktúry
- Techniky tvorby efektívnych algoritmov
(greedy algoritmy, dynamické programovanie, rozdeľuj a panuj)
- NP-ťažké problémy

Hodnotenie predmetu

- 30%: Domáce úlohy (cca každé dva týždne)
(vr. dvoch programátorských príkladov)
- 20%: Midterm (9. novembra, 18:10-20:10)
- 50%: Písomná skúška
- zo skúšky je potrebné získať aspoň 50% bodov
- 90+ = A, 80+ = B, 70+ = C, 60+ = D, 50+ = E

Opisovanie

- Budeme kruto trestať:
 - –100% príslušného bodového hodnotenia
 - disciplinárna komisia
- **Podporujeme** diskusiu o domácich úlohách, **ale**:
 - Nerobte si poznámky
 - Počkajte niekoľko hodín, kým začnete spisovať vlastné riešenie