Čas algoritmu A **na vstupe** x je $\boxed{\text{čas}}$, ktorý algoritmus A potrebuje na vyriešenie vstupu x (označme $T_A(x)$).

Časová zložitosť algoritmu A je funkcia veľkosti vstupu, pričom pre $\boxed{\text{veľkosť vstupu}}$ n je to najhorší čas, ktorý algoritmus potrebuje na riešenie vstupu tejto veľkosti, t.j.

$$T_A(n) = \max\{T_A(x) \mid |x| = n\}$$

Časová zložitosť problému je časová zložitosť $\boxed{\text{najlepšieho algoritmu,}}$ ktorý rieši daný problém.

Označenie	**Definícia**	
$f(n) \in O(g(n))$	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 $f(n) \in \Omega(g(n))$	$=$
$f(n) \in o(g(n))$	Pre ľubovoľné $c > 0$ existuje $n_0 > 0$ také, že $(\forall n > n_0)(0 \leq f(n) < cg(n))$	$<$
$f(n) \in \omega(g(n))$	Pre ľubovoľné $c > 0$ existuje $n_0 > 0$ také, že $(\forall n > n_0)(f(n) > cg(n) \geq 0)$	$>$

Greedy algoritmus pre problém výberu aktivít

```
Sort all activities by their finishing time
(now f[1]<=f[2]<=...<=f[n])

last_activity_end:=-infinity;

for i:=1 to n
  if (s[i]>=last_activity_end) then
     output activity (s[i],f[i]);
     last_activity_end:=f[i];
```

Časová zložitosť: $\Theta(n \log n)$

“Vzor” dôkazu správnosti greedy algoritmu

Lema: Predpokladajme, že greedy algoritmus vráti riešenie G. Potom existuje optimálne riešenie, ktoré sa s riešením G zhoduje na prvých k voľbách.

Dôkaz: Matematickou indukciou podľa k.

Báza indukcie. Pre $k = 0$ – ľubovoľné optimálne riešenie.

Indukčný krok. (Prepokladajme, že sme neurobili chybu pri prvých k voľbách, potom aj $(k + 1)$-vá voľba je OK.)

- Predpokladajme, že existuje optimálne riešenie OPT, ktoré sa zhoduje s G na prvých k voľbách.
- Vyrobíme riešenie OPT':
  - OPT' má rovnakú hodnotu ako OPT
    (a preto je tiež optimálne)
  - OPT' súhlasí s G na jednej ďalšej $(k + 1)$-vej voľbe.