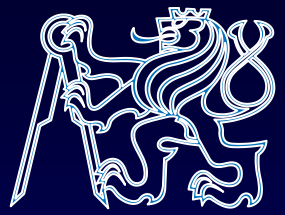
- ©Jan Schmidt 2011
  Katedra číslicového návrhu
  Fakulta informačních technologií
  České vysoké učení technické v Praze

- Zimní semestr 2013/14



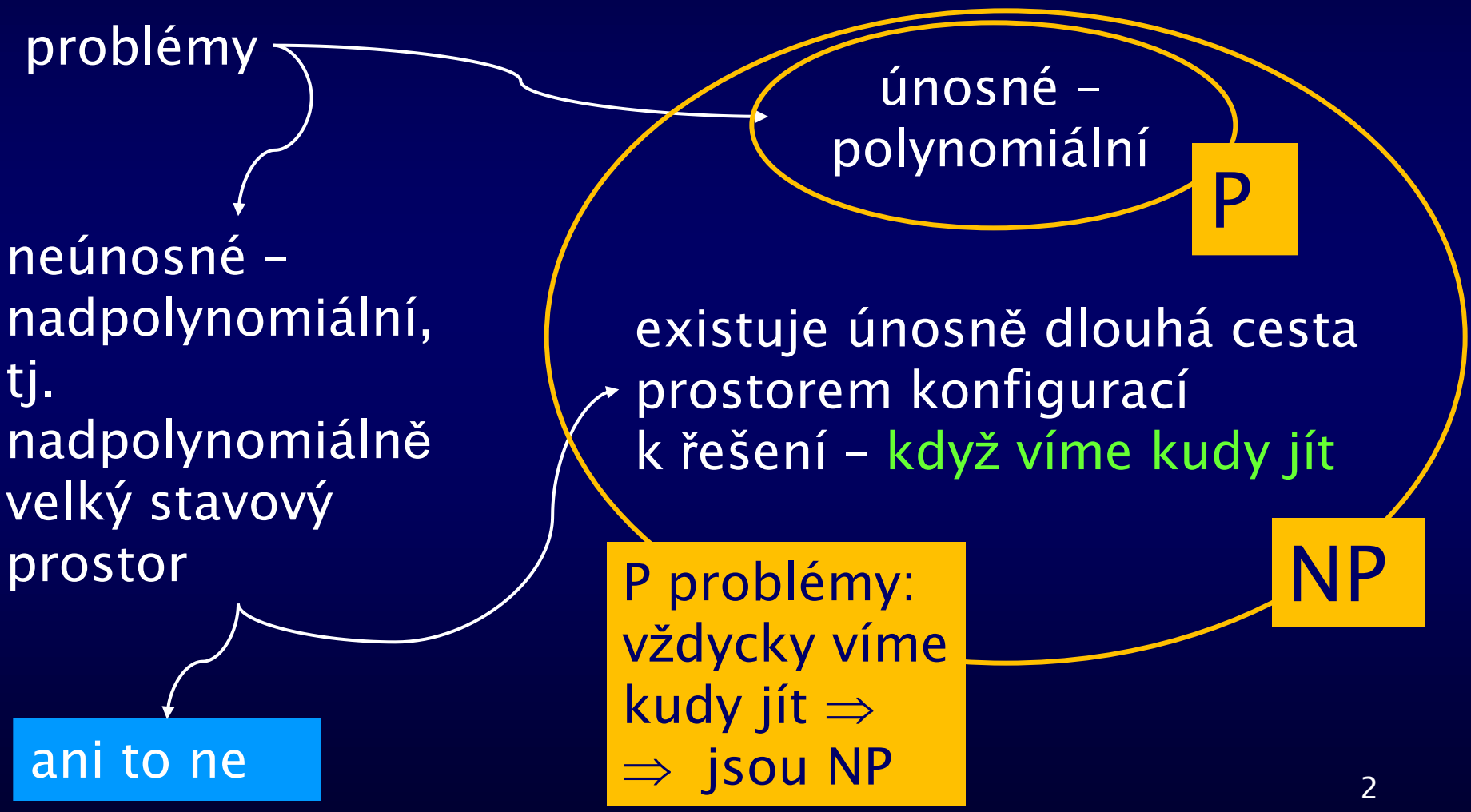
EVROPSKÝ SOCIÁLNÍ FOND
PRAHA & EU: INVESTUJENE
DO VAŠÍ BUDOUCNOSTI

MI-PAA

3. Třídy P a NP

- Model výpočtu: Turingův stroj
- Rozhodovací problémy: třídy P a NP

Složitost problémů

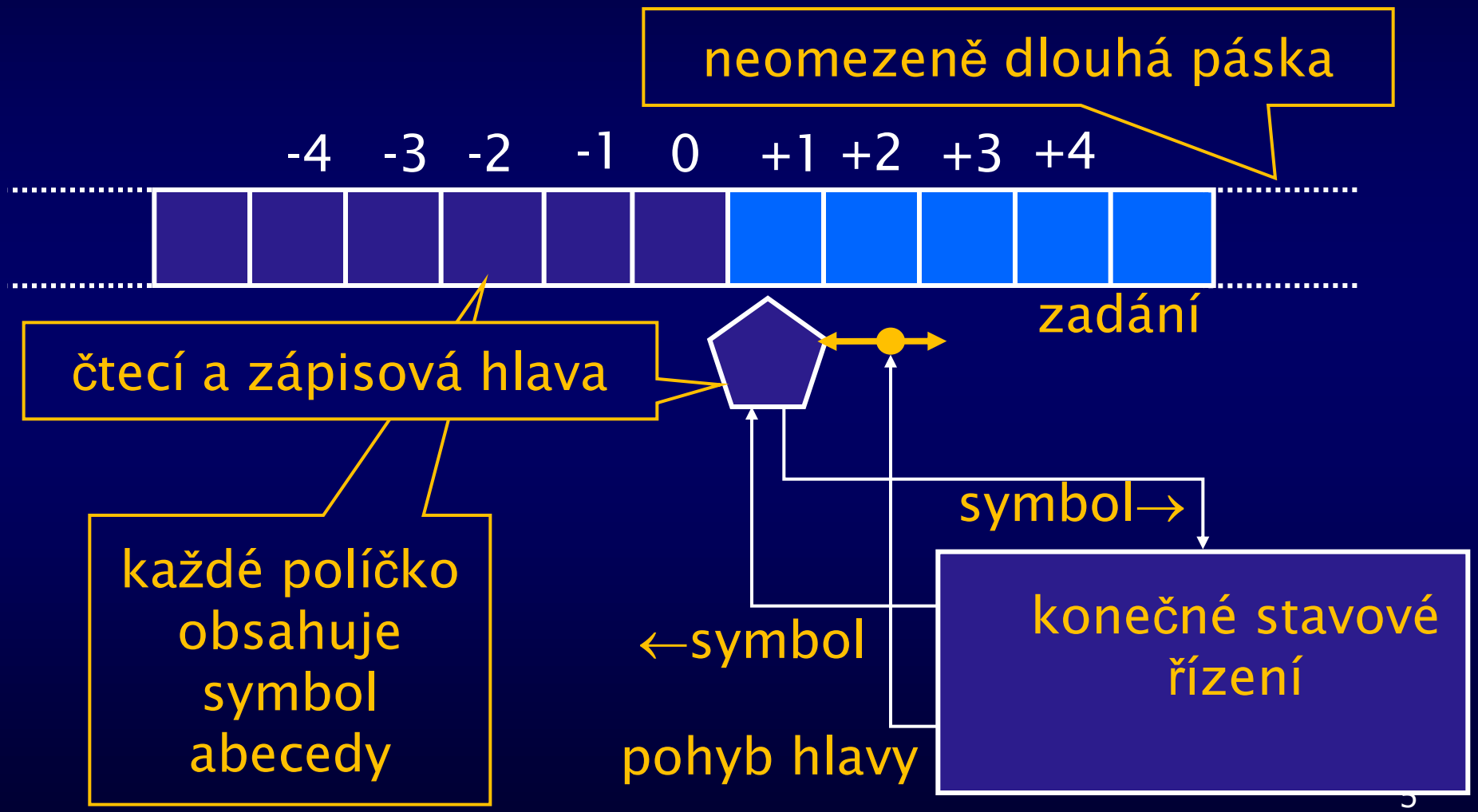


… měření složitosti

- Jak měřit velikost instance?
  - hrubá míra: počet prvků instance (uzlů, čísel, prvků množiny)
  - jemná míra: počet bitů, nutných k zakódování instance
- Jak měřit čas výpočtu?
  - počet „typických operací“
  - počet kroků jednotného výpočetního modelu

VÝPOČETNÍ MODELY A TŘÍDA P

Turingův stroj



Turingův stroj

- Program:
  - množina Γ symbolů pásky, symbol $b \in \Gamma$
  - $\Sigma \subset \Gamma$ množina vstupních symbolů, $b \notin \Sigma$
  - množina stavů Q, počáteční stav $q_0 \in Q$, koncové stavy $q_{ANO}, q_{NE} \in Q$
  - přechodová funkce $\delta: (Q - \{q_{ANO}, q_{NE}\}) \times \Gamma \rightarrow Q \times \Gamma \times \{-1, +1\}$
- Inicializace: stav q_0, políčko 1 pásky
- Konec: q_{ANO}, q_{NE}
- Výpočet: $(q \in Q, s \in \Gamma) \rightarrow (q', s', \Delta)$

stav

čtený symbol

nový stav

zapsaný symbol

pohyb pásky

Řešení problému deterministickým Turingovým strojem

- Definice: řešení problému Turingovým strojem
- Program *M* pro deterministický Turingův stroj řeší rozhodovací problém *Π*, jestliže se výpočet zastaví po konečném počtu kroků pro každou instanci problému *Π*.
- Program *M* pro deterministický Turingův stroj řeší rozhodovací problém *Π* v čase *t*, jestliže se výpočet zastaví po *t* krocích pro každou instanci problému *Π*.
- Program *M* pro deterministický Turingův stroj řeší rozhodovací problém *Π* s pamětí *m*, jestliže počet použitých políček je nejvýše *m* pro každou instanci problému *Π*.

Kódování instance

- Vstup Turingova stroje:
  řetěz q^* symbolů $q \in \Gamma$
- Výstup: „ano“, jestliže q^* kóduje instanci problému, která má řešení
- … lze mluvit o P-jazycích, NP-jazycích, jazyce TSP, jazyce SAT …
- abeceda Γ nezávisí na instanci
- způsob kódování instance neovlivní čas výpočtu více než polynomiálně
- je možné použít nějakého binárního kódování
- $\Rightarrow$ problém je podmnožina $\{0,1\}^*$
  (charakterizován podmnožinou) $\{0,1\}^*$

MI-PAA 1

Příklad: kódování grafu $G=(V, E)$

$|X|$ … počet prvků množiny X

- matice sousednosti:
  $|V|^2$ bitů
- incidenční matice:
  $|V||E| = O(|V|^3)$ bitů
- seznam hran jako dvojic indexů uzlů:
  $O(|E|\log |V|) = O(|V|^2 \log|V|)$ bitů
- schválnost – patologický případ
  matice sousednosti a za ní 2^n nul

Třída P

- Definice: třída P
  Rozhodovací problém patří do třídy P, jestliže pro něj existuje program pro deterministický Turingův stroj, který jej řeší v čase $O(n^k)$, kde n je velikost instance a k konečné číslo.
- PSPACE:
  v paměti $O(n^k)$, kde n je velikost instance a k konečné číslo.
- EXPTIME:
  v čase $O(2^{P(n)})$, kde $P(n)$ je polynom ve velikosti instance n.

NP

NON-DETERMINISTICALLY POLYNOMIAL

NEDETERMINISTICKY POLYNOMIÁLNÍ

Nedeterministický Turingův stroj

- Program: …přechodová relace
  $\delta \subset (Q - \{q_{ANO}, q_{NE}\}) \times \Gamma \times Q \times \Gamma \times \{-1, +1\}$
- Výpočet: $(q \in Q, s \in \Gamma) \rightarrow \{(q', s', \Delta)\}$
- Představa: v každém kroku se stroj naklonuje a každá kopie vykoná jeden konkrétní možný krok
- Představa: jedna z kopií vykoná onu únosně dlouhou cestu stavovým prostorem,
  pokud tato cesta existuje, tj. pokud instance má řešení
- Jiná možnost: v každém kroku si vybereme jednu možnost, náhodou je to vždycky ta „správná“

Řešení problému **ne**deterministickým Turingovým strojem

- Definice: řešení problému nedeterministickým Turingovým strojem

  Nechť Π_{ANO} je množina instancí problému Π, které mají výstup ANO. Program M pro nedeterministický Turingův stroj řeší rozhodovací problém Π v čase t, jestliže se výpočet zastaví po t krocích pro každou instanci $I \in \Pi_{ANO}$ problému Π.

Vlastnosti

- Nic se neříká o instancích Π_{NE}. Pokud hledáme únosně dlouhou cestu k řešení, na Π_{NE} nemá význam.
- Věta (výpočetní mohutnost nedeterminismu):
  Jestliže nedeterministický Turingův stroj řeší problém Π v čase $T(n)$,
  pak deterministický Turingův stroj řeší Π v čase $2^{O(T(n))}$.

Třída NP

- Definice: třída NP
  Rozhodovací problém Π patří do třídy NP, jestliže pro něj existuje program pro nedeterministický Turingův stroj, který každou instanci $I \in \Pi_{ANO}$ problému Π řeší v čase $O(n^k)$, kde n je délka vstupních dat a k konečné číslo.
- Definice: třída NP
  Rozhodovací problém Π patří do třídy NP, jestliže pro každou instanci $I \in \Pi_{ANO}$ problému existuje konfigurace Y taková, že kontrola, zda Y je řešením, patří do P. V této souvislosti nazýváme Y certifikátem.

omezující podmínky lze vyhodnotit v polynomiálním čase

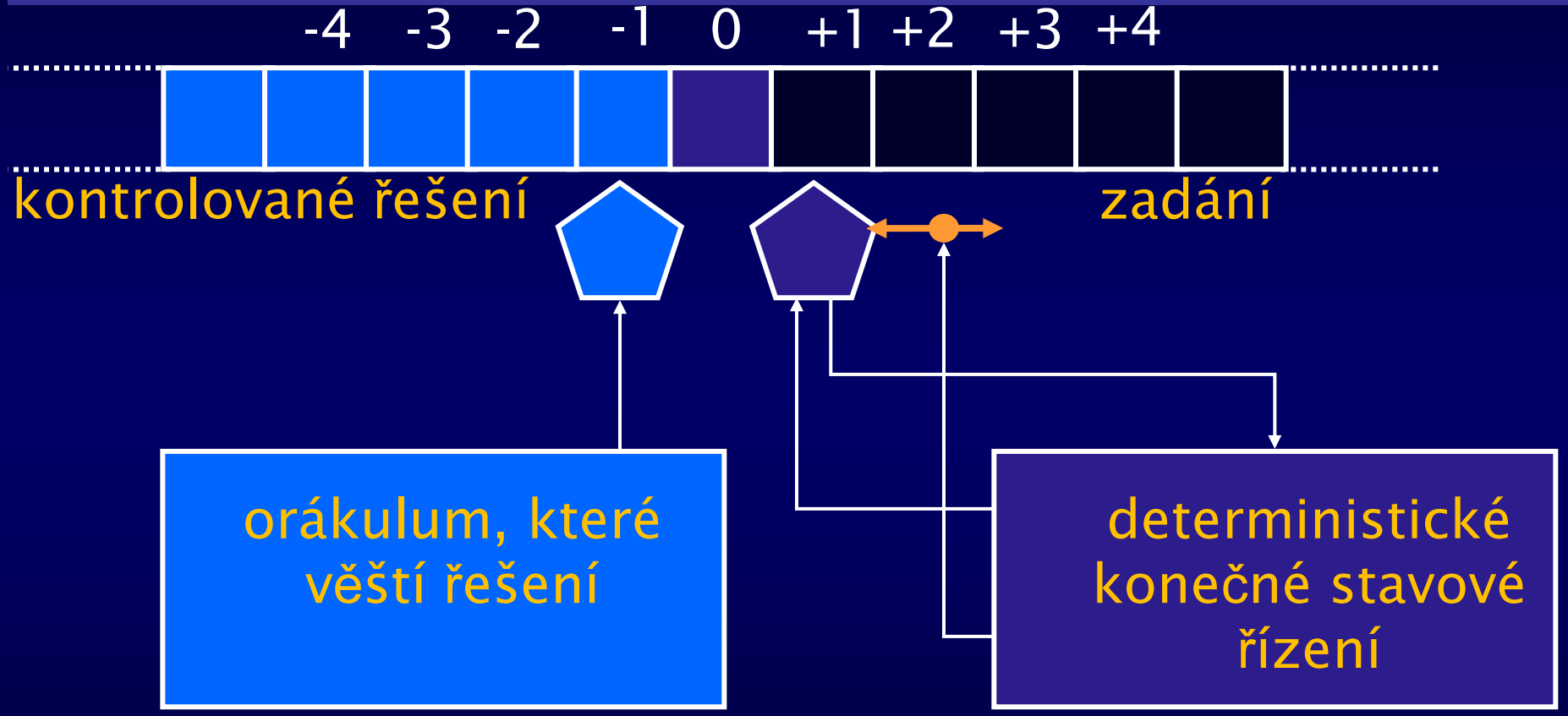
Příklad: Hamiltonova kružnice v grafu (HC), nedeterministický algoritmus

1. Nechť podgraf $G' = (V', E')$ je tvořen libovolným uzlem v původního grafu $G=(V,E)$.
2. V každém kroku, nechť každá „kopie“ algoritmu přidá
   - jednu (různou) hranu $e \in E-E'$, $e = (u,v)$ takovou, že $u \in V'$, $v \notin V'$, stupeň $u=1$
   - uzel v.
3. Není-li to možné, příslušná „kopie“ končí.
4. Jestliže přidaná hrana utvoří z G'
   1. kružnici kratší než $|V|$, příslušná „kopie“ končí.
   2. kružnici délky $|V|$, algoritmus vydá výstup „ano“.

Nedeterministický algoritmus, poznámky

- Jestliže graf obsahuje Hamiltonovu kružnici, po $|V|$ krocích je nalezena.
- Existuje nedeterministický algoritmus řešící problém HC $\Rightarrow$ HC patří do NP
- V tomto případě máme štěstí – pokud Hamiltonovu kružnici nenajdeme po $|V|$ krocích, pak neexistuje. Obecně tomu tak není

Nedeterministický Turingův stroj – představa, která vychází z kontroly řešení



- nejjednodušší případ „stroje s orákulem“
- obecně: když mám pomocný prostředek nebo podprogram určité výkonnosti, jaká složitost zbývá?

Příklad: Hamiltonova kružnice v grafu – polynomiální kontrola

- Konfigurace: podgraf $G' = (V', E')$ původního grafu $G=(V,E)$.
- Certifikát: podgraf $G' = (V', E')$, o kterém se tvrdí, že je to Hamiltonova kružnice
- Kontrola:
  - $|V|$ uzlů … $O(|V|)$
  - $E' \subset E$ … $O(|V|)$
  - žádný uzel dvakrát ($|V|$ čítačů) … $O(|V|)$
- Existuje deterministický algoritmus kontrolující certifikát HC $\Rightarrow$ HC patří do NP

Vztah tříd P a NP

funkce je zvláštním případem relace

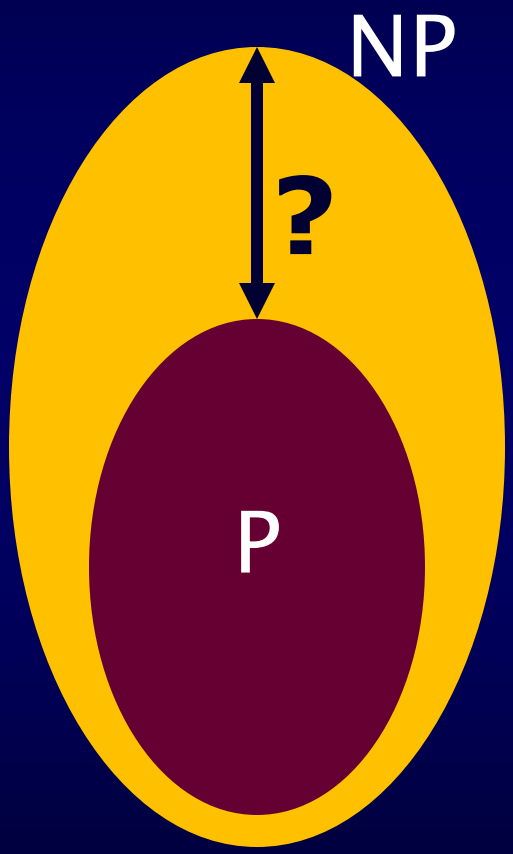
deterministický automat je zvláštním případem nedeterministického automatu

deterministický Turingův stroj je zvláštním případem nedeterministického Turingova stroje

$P \subseteq NP$

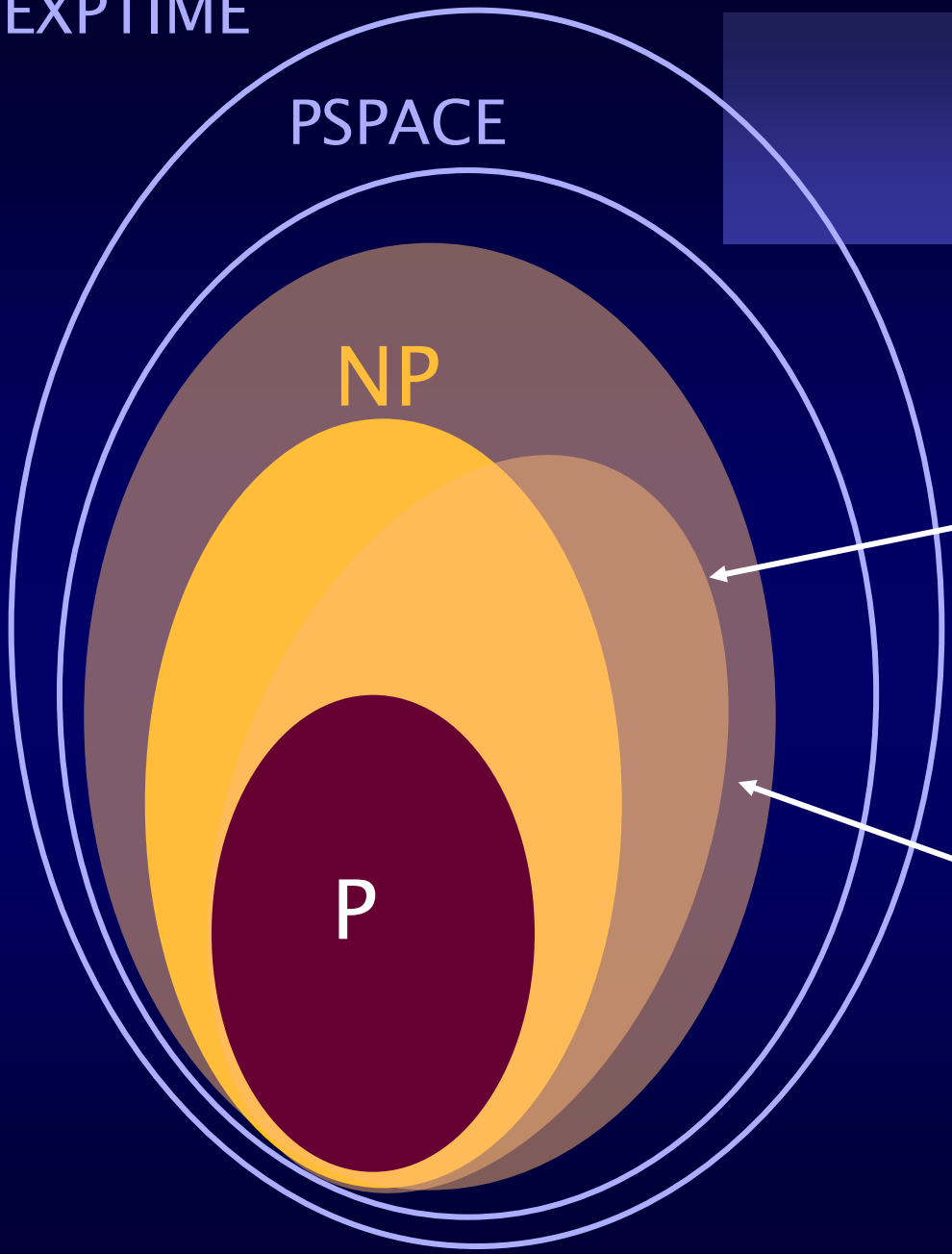
instance $I \in \Pi_{ANO}$ jsou podmnožinou všech instancí

Vztah tříd P a NP



- možná, že P = NP:
  na každý NP-problém existuje polynomiální algoritmus,
  ale my o něm nevíme
- ale jsou příznaky, že P⊂NP
- jeden z hlavních příznaků:
  viz příště

… a dál



co-NP: komplementární, „srovnatelné“ s NP

polynomiální hierarchie: „mezi“ P a PSPSACE

Když trochu „otočíme“ NP problém…

Je dán graf $G=(V,E)$. Je tento graf prost Hamitonových kružnic?

Je dána Booleovská formule $F(X)$ n proměnných $X = (x_1, x_2, \ldots x_n)$. Je tato formule nesplnitelná?

Příklad:
$F(X) = 1$… vstup X způsobuje chybnou funkci programu
$\forall X$, $F(X) = 0$ … mám pokoj

NP nefunguje!
Kde mám certifikát u instancí Π_{ANO}*?*

Komplement:
co je to „otočit“ problém, třídu

- Připomeneme: instance je charakterizována řetězem q^* symbolů $q \in \Gamma$, kde Γ je například {0,1}
- Problém Π je charakterizován množinou řetězů, které kódují instance s výstupem ANO, tedy podmnožinou množiny $\{q^*\}$.
- Problém komplementární k problému Π je charakterizován doplňkem této podmnožiny do $\{q^*\}$.
- Doplněk vytvoříme De Mogranovým pravidlem
- Ke každé třídě X problémů lze zkonstruovat třídu co-X jako množinu problémů komplementárních ke všem problémům z X.

…dostaneme…

- NP problém: $\exists Y, R(I, Y)$
- co-NP problém: $\forall Y, R'(I, Y)$

komplement třídy NP:
třída co-NP

- I … instance
- Y … konfigurace
- $R(I, Y)$ … omezující podmínky
  polynomiální složitost
- $R'(I, Y)$ … jejich komplement
  polynomiální složitost

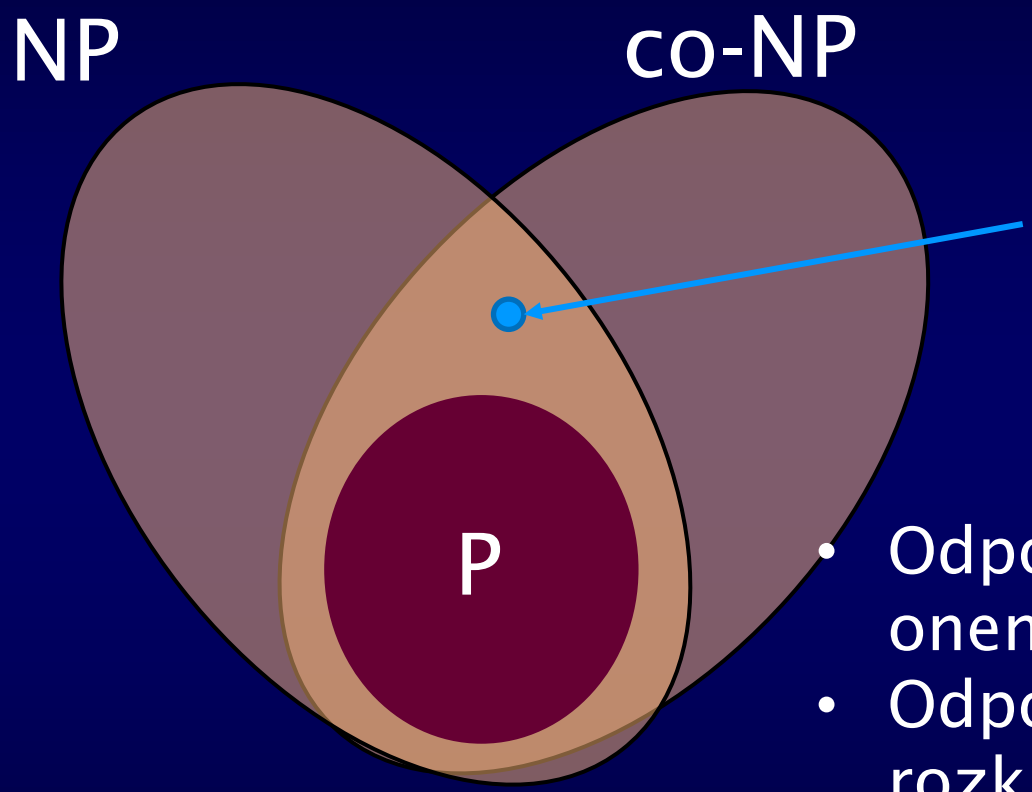
Je dán graf $G=(V,E)$. Platí pro každou kružnici Y, že prochází méně než $|V|$ uzly?

Je dána Booleovská formule $F(X)$ n proměnných $X = (x_1, x_2, \ldots x_n)$. Platí pro každé ohodnocení Y proměnných X, $F(Y) = 0$?

Problémy mimo NP: co-NP

- NP problémy:
  - *Y* je krátký svědek odpovědi ANO (certifikát)
  - Krátkého svědka odpovědi NE nemáme
- co-NP problémy:
  - *Y*, pro které neplatí *R*(*I*, *Y*), je krátkým svědkem odpovědi NE (protipříklad)
  - Krátkého svědka odpovědi ANO nemáme
- Svědkové odpovědi ANO
  - NP: ∃-certifikát (krátký)
  - co-NP: ∀-certifikát
    (dlouhý, ale krátký pro každou konfiguraci)
  - Nemůžeme-li žádat krátké vyhodnocení celého certifikátu, můžeme žádat aspoň krátké vyhodnocení pro každou konfiguraci

Třída P, NP, co-NP



Faktorizace čísla jako rozhodovací problém:
Dáno celé číslo *N*, existuje jeho prvočinitel, jehož poslední číslice je 7?

- Odpověď ANO má ∃-certifikát: onen prvočinitel
- Odpověď NE má ∀-certifkát: rozklad na prvočinitele

- pokud je v P … radši na to nemyslet
- pokud patří k nejtěžším v NP … pak P=NP

Problémy potenciálně horší než NP

- SAT: Booleovská formule $F(X)$
  n proměnných $X = (x_1, x_2, \ldots x_n)$.
- $\exists\ Y,\ F(Y) = 1$?

QBF_2 nebo $QSAT_2$
Quantified Boolean Formula

- Booleovská formule $F(X_1, X_2)$
  $2n$ proměnných $X_1 = (x_1, x_2, \ldots x_n)$,
  $X_2 = (x_{n+1}, x_{n+2}, \ldots x_{2n})$
- $\exists Y_1, \forall Y_2, F(Y_1, Y_2) = 1$?

Dáno řešení Y_1 , kontrola:
platí, že $\forall Y_2, F(Y_1, Y_2) = 1$?

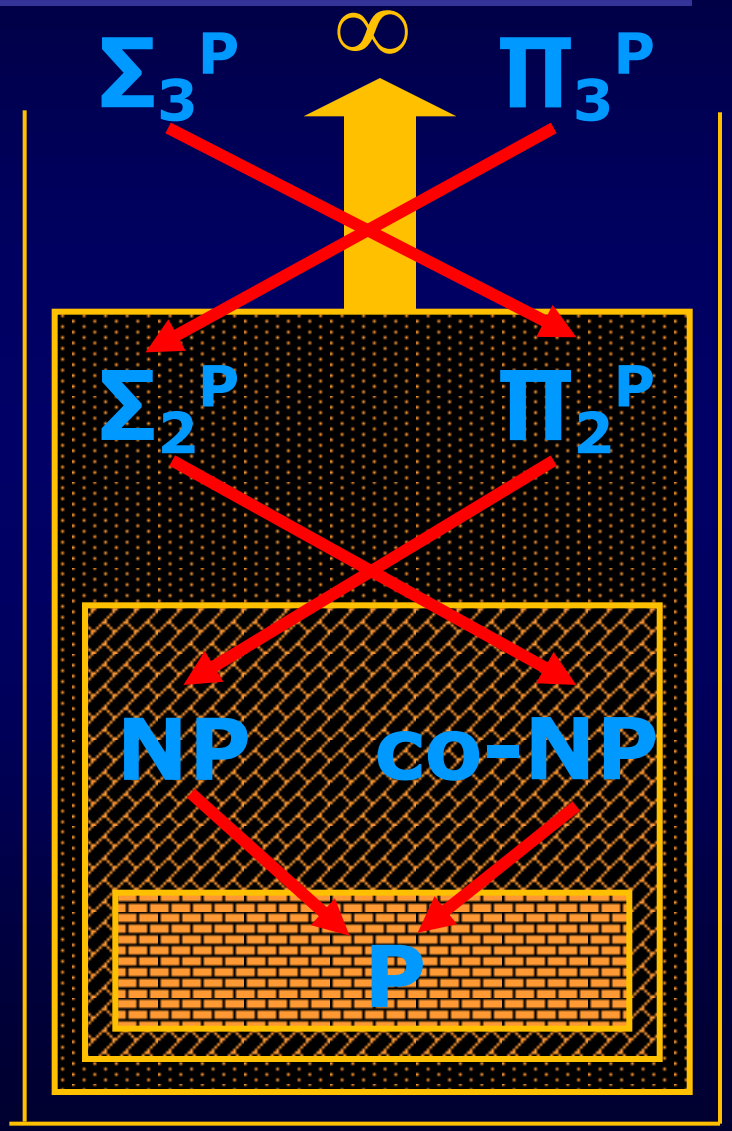
! problém kontroly je v **co-NP**

MI-CPX 5

Polynomiální hierarchie

pokud P ≠ NP a pokud polynomiální hierarchie někde nekončí

obvykle se polynomiální hierarchie zavádí pomocí Turingových strojů s orákulem



třída problémů, jejichž kontrola leží ve třídě komplementů předchozí třídy

třída problémů, jejichž kontrola leží v P a jejich komplementů

MI-CPX 6

třída P

Problémy v Σ_k^P a v Π_k^P

- $\underbrace{\exists Y_1\ \forall Y_2\ \exists Y_3 \ldots}_{k}$, $F(Y_1, Y_2, Y_3, \ldots) = 1$ je v Σ_k^P
- $\underbrace{\forall Y_1\ \exists Y_2\ \forall\ Y_3 \ldots}_{k}$, $F(Y_1, Y_2, Y_3, \ldots) = 1$ je v Π_k^P

- … a nejsou níže, protože jsou v té třídě nejtěžší
- … a jak se to zjistí, příště

Čemu teď rozumíme

Modelování výpočtu deterministickým a nedeterministickým Turingovým strojem.

Význam kódování vstupní instance.

Vztah výpočetní mohutnosti determinismu a nedeterminismu.

Význam a definice třídy P

Význam třídy NP; její definice pomocí nedeterminismu a pomocí kontroly certifikátu.

Komplementace NP problému, význam certifikátu, význam a definice třídy co-NP.

Základní podoba polynomiální hierarchie.

Jaké pojmy k tomu potřebujeme

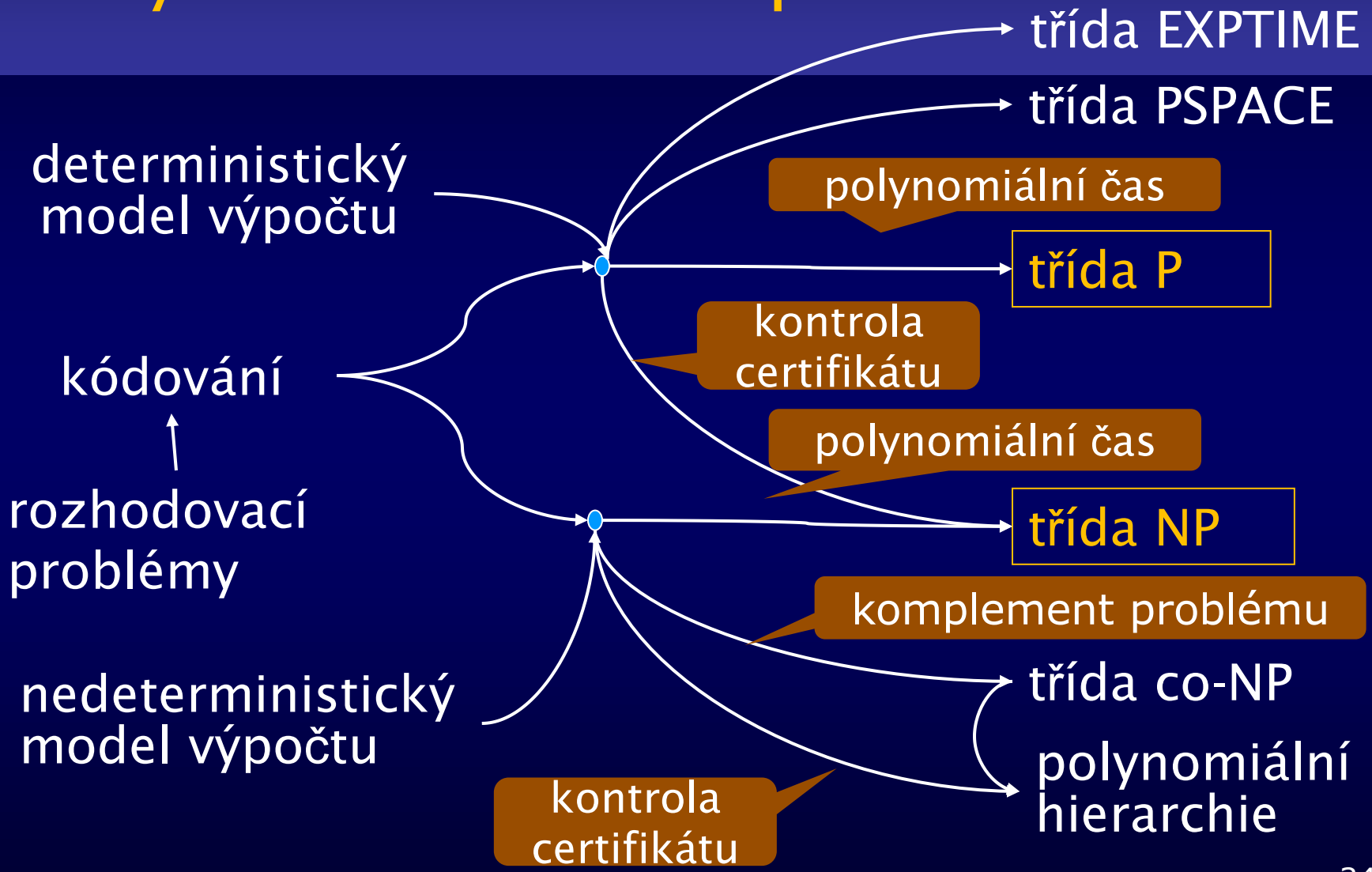
Turingův stroj deterministický, nedeterministický

komplementární problém, třída

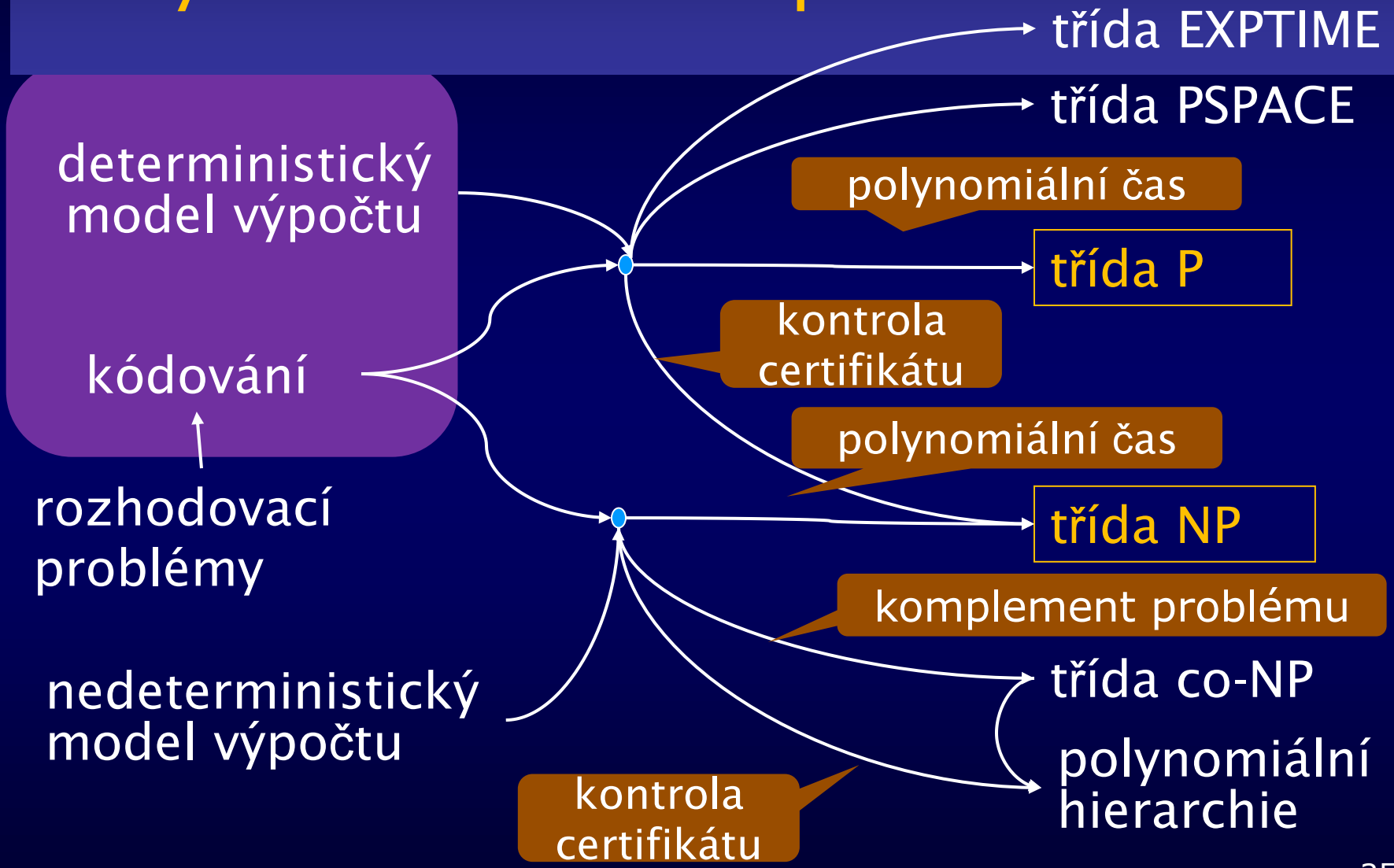
certifikát, $\exists$-certifikát, $\forall$-certifikát

třídy: P, NP, co-NP, PSPACE, EXPTIME,
třídy polynomiální hierarchie

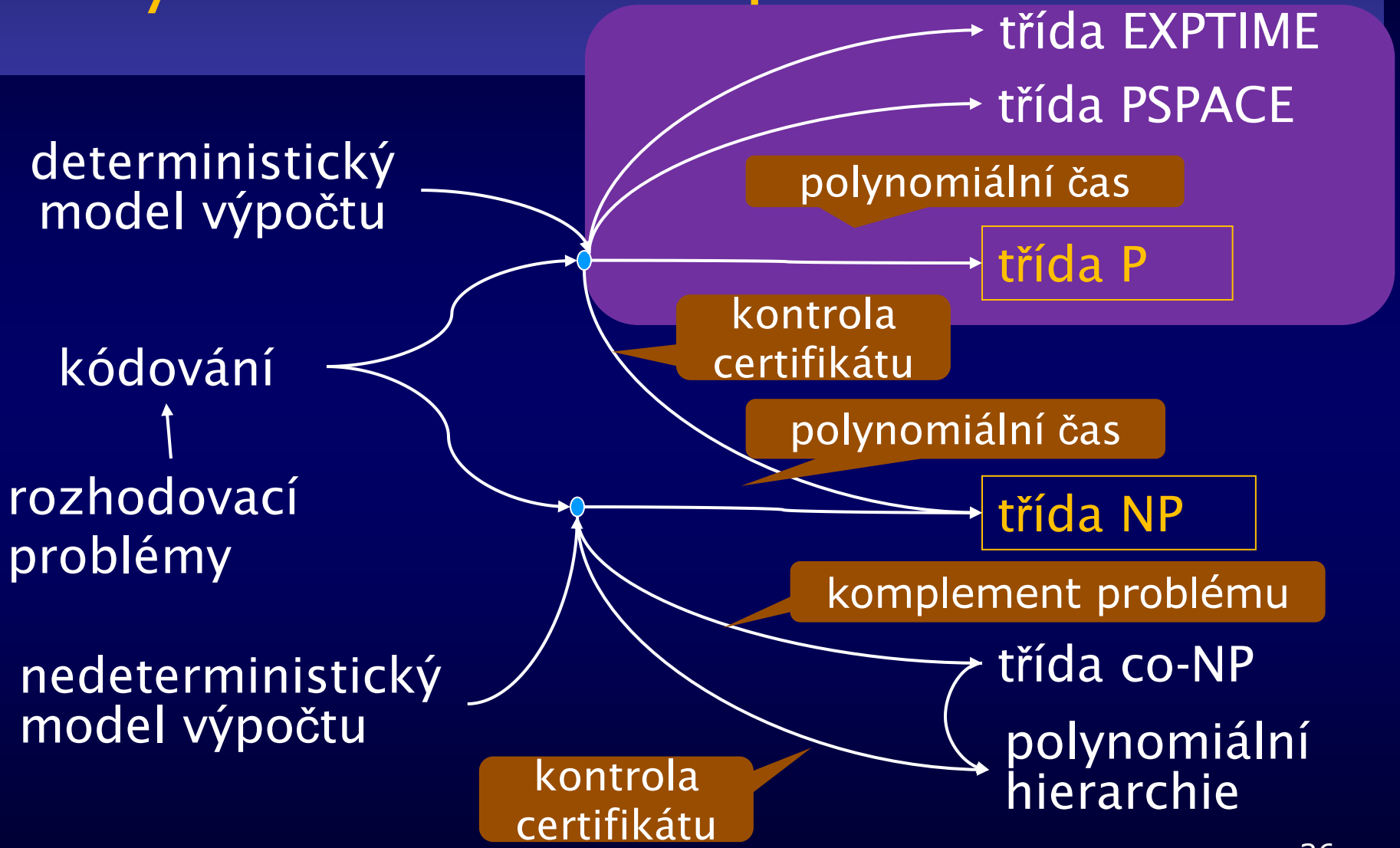
Třídy rozhodovacích problémů



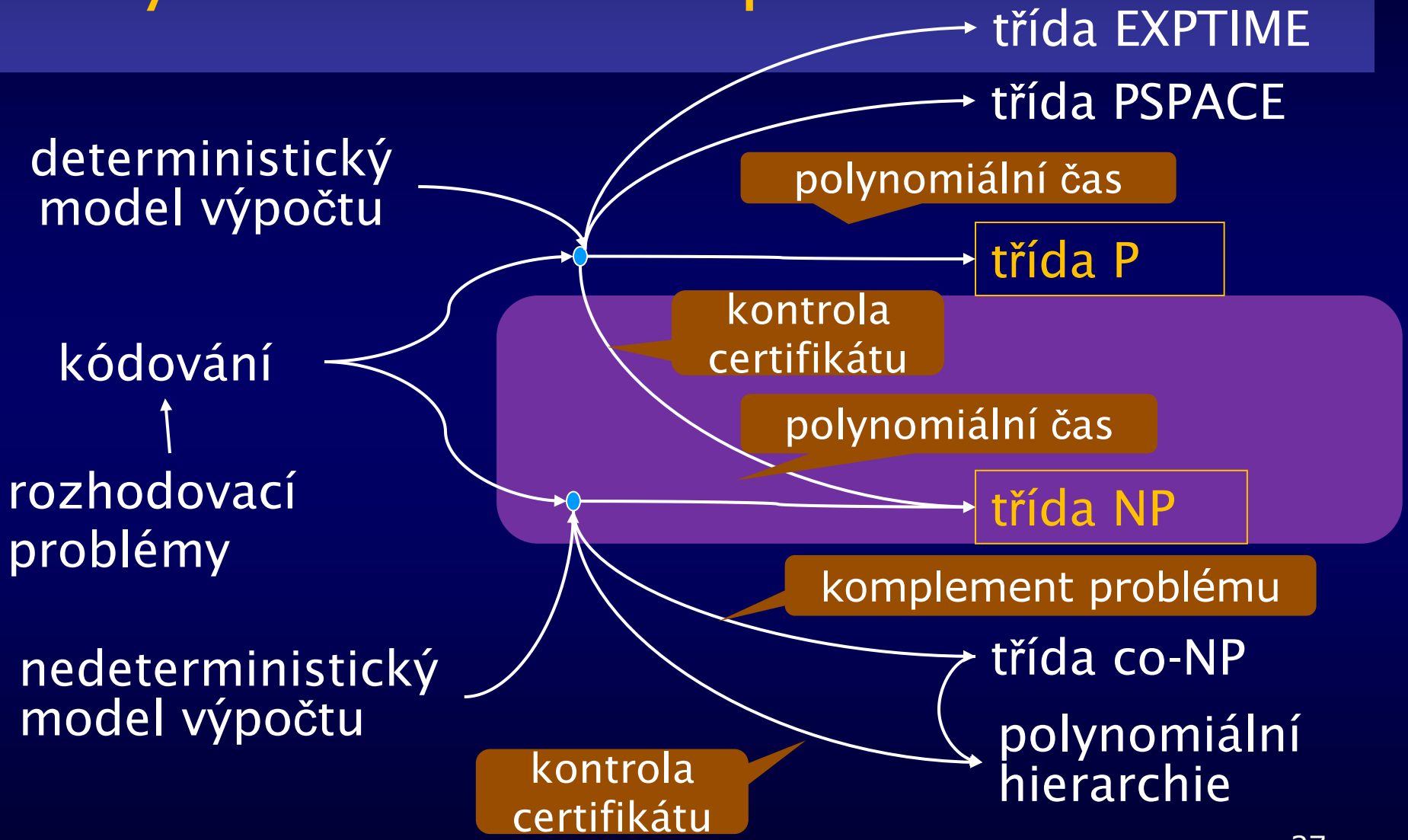
Třídy rozhodovacích problémů



Třídy rozhodovacích problémů



Třídy rozhodovacích problémů



Třídy rozhodovacích problémů