4.7.8 Věta. Je-li jazyk L ve třídě $\mathcal{P}$SPACE ($\mathcal{NP}$SPACE), pak L je rozhodován deterministickým (nedeterministickým) Turingovým strojem M s polynomiální paměťovou složitostí, který se vždy zastaví po nejvýše $c^{q(n)}$ krocích, kde $q(n)$ je vhodný polynom a c konstanta.

4.7.9 Myšlenka důkazy věty 4.7.8. Předpokládejme, že $L \in \mathcal{NP}$SPACE. Pak existuje Turingův stroj M_1, který přijímá jazyk L s paměťovou složitostí $p(n)$ ($p(n)$ je vhodný polynom). Víme (z věty 4.7.6), že existuje konstanta c taková, že Turingův stroj M_1 k tomu, aby aspoň v jednom výpočtu slovo přijal, potřebuje nejvýše $c^{p(n)+1}$ kroků.

Vytvoříme Turingův stroj M_2, který bude mít dvě pásky: první páska bude simulovat M_1, druhá bude počítat kroky na první pásce. Jestliže počet kroků překročí $c^{p(n)+1}$, Turingův stroj M_2 se neúspěsně zastaví. Počítání kroků M_2 provádí v c-adické soustavě (tak, aby zabralo jen $\mathcal{O}(p(n))$ polí pásky).

Hledaný Turingův stroj M je Turingův stroj s jednou páskou, který simuluje Turingův stroj M_2. Turingův stroj M pracuje v s časovou složitostí $\mathcal{O}(c^{2p(n)})$, tedy v maximálně $d\,c^{2p(n)}$ krocích. Nyní stačí položit $q(n) = 2p(n) + \log_c d$ nebo jakýkoli polynom větší. Uvědomte si, že konstrukce jednopáskového Turingova stroje pro vícepáskový asymptoticky nezvětší paměťovou složitost.

4.7.10 Savitchova věta. Platí

$$\mathcal{P}\text{SPACE} = \mathcal{NP}\text{SPACE}.$$

4.7.11 Nástin myšlenky důkazu Savitchovy věty. Zřejmě $\mathcal{P}$SPACE $\subseteq$ $\mathcal{NP}$SPACE. Důkaz opačné inkluze $\mathcal{NP}$SPACE $\subseteq$ $\mathcal{P}$SPACE spočívá v tom, že jsme schopni nedeterministický Turingův stroj M pracující s paměťovou složitostí $p(n)$ simulovat deterministickým Turingovým strojem N, který pracuje s paměťovou složitostí $\mathcal{O}((p(n))^2)$ (o časové složitosti nic dokazovat nebudeme). Konstrukce N je založena na následující rekursivní proceduře REACH$(I, J; m)$, kde I a J jsou situace NTM M a m je kladné přirozené číslo. REACH$(I, J; m)$ vrátí 1, jestliže $I \vdash^\star J$ v nejvýše m krocích.

REACH$(I, J; m)$

Vstup: Situace I a J nedeterministického Turingova stroje M a m je kladné přirozené číslo.

Výstup: TRUE, jestliže J je dostupná z I v nejvýše m krocích, FALSE v opačném případě.

```
begin
   if m = 1 then
         if I = J nebo I ⊢ J then return TRUE
         else return FALSE
   end
   else (induktivní část)
         for každou možnou situaci K do
         if REACH(I, K; ⌊m/2⌋) a REACH(K, J; ⌈m/2⌉ then
               return TRUE
         return FALSE
   end
```

`end`

Uvědomte si, že rekurzivní procedura REACH$(I, J; m)$ má vždy na zásobníku jen jednu trojici $(I_1, J_1; m)$, nejvýše jednu trojici $(I_2, J_2; \frac{m}{2})$, nejvýše jednu $(I_3, J_3; \frac{m}{4})$, atd. Tedy současně nemá na zásobníku víc než $\lg m$ různých trojic.

Nyní využijeme proceduru REACH$(I, J; m)$ k sestrojení deterministického Turingova stroje přijímajícího stejný jazyk a pracujícího s polynomiální paměťovou složitostí. Je dán nedeterministický Turingův stroj M, který přijímá jazyk L s polynomiální paměťovou složitostí $p(n)$. Pro vstup w voláme REACH$(I_0, J; m)$, kde I_0 je počáteční situace M, J je některá přijímající situace M a $m = c^{p(n)+1}$ (c je konstanta z 4.7.6). Dá se dokázat, že pro vykonání procedury REACH$(I, J; m)$ deterministickým Turingovým strojem stačí paměťová složitost $\mathcal{O}([p(n)]^2)$. To vyplývá z toho, že REACH$(I_0, J; m)$ má na zásobníku maximálně $\lg c^{p(n)+1} = d\,p(n)$ trojic $(I, J; r)$ a každá z trojic má nejvýše délku $\mathcal{O}(p(n))$. (Uvědomte si, že nám nezáleží na tom, jak dlouho deterministický Turingův stroj pracuje, zajímáme se pouze o paměťové nároky.)

4.7.12 Důsledek. Platí

$$\mathcal{P} \subseteq \mathcal{NP} \subseteq \mathcal{P}\texttt{SPACE}.$$

4.8 Třídy založené na pravděpodobnostních algoritmech

4.8.1 Randomizovaný Turingův stroj. RTM je, zhruba řečeno, Turingův stroj M se dvěma nebo více páskami, kde první páska má stejnou roli jako u deterministického Turingova stroje, ale druhá páska obsahuje náhodnou posloupnost 0 a 1, tj. na každém políčku se 0 objeví s pravděpodobností $\frac{1}{2}$ a 1 také s pravděpodobností $\frac{1}{2}$.

Na začátku práce:

- stroj M se nachází v počátečním stavu q_0;
- první páska obsahuje vstupní slovo w, zbytek pásky pak blanky B;
- druhá páska obsahuje náhodnou posloupnost 0 a 1;
- případné další pásky obsahují B;
- všechny hlavy jsou nastaveny na prvním políčku dané pásky.

Na základě stavu q, ve kterém se stroj M nachází, a na základě obsahu políček, které jednotlivé hlavy čtou, přechodová funkce δ určuje, zda se M zastaví nebo přejde do nového stavu p, přepíše obsah první pásky (**nikoli ale obsah druhé pásky**) a hlavy posune doprava, doleva nebo zůstanou stát (posuny hlav jsou nezávislé).

Definice. Randomizovaný Turingův stroj je sedmice $(Q, \Sigma, \Gamma, \delta, q_0, B.F)$, kde Q, Σ, Γ, q_0, B a F mají stejný význam jako pro deterministický Turingův stroj. Přechodová funkce je parciální zobrazení

$$\delta\colon (Q \setminus F) \times \Gamma \times \{0,1\} \to Q \times \Gamma \times \{L, R, S\}^2$$

Je-li M ve stavu q, hlava na první pásce čte symbol X, na druhé pásce je číslo a a

$$\delta(q, X, a) = (p, Y, D_1, D_2), \quad q, p \in Q, a \in \{0, 1\}, X, Y \in \Gamma, D_1, D_2 \in \{L, R, S\},$$

pak M se přesune do stavu p, na první pásku napíše Y a i-tá hlava se posune doprava pro $D_i = R$, doleva pro $D_i = L$ nebo zůstane na místě pro $D_i = S$.

Jestliže $\delta(q, X, a)$ není definováno, M se zastaví.

M se úspěšně zastaví právě tehdy, když se přesune do koncového (přijímajícího) stavu q_f.

4.8.2 Poznámka. Rozdíl mezi RTM a obyčejným TM je v roli druhé pásky. Turingův stroj s dvěma páskami může přepisovat i obsah druhé pásky a to je v případě RTM zakázáno. Navíc při dvou bězích RTM může být průběh práce RTM různý (záleží na náhodně vygenerovaném obsahu druhé pásky). To se u vícepáskového deterministického TM stát nemůže.

Může se zdát, že tento model je nerealistický — nemůžeme před začátkem práce naplnit nekonečnou pásku. Toto je ale „realizováno“ tak, že v okamžiku, kdy druhá hlava čte dosud nenavštívené políčko druhé pásky, náhodně se vygeneruje 0 nebo 1 každé s pravděpodobností $\frac{1}{2}$ a tento symbol už se nikdy během jednoho průběhu práce TM nezmění.

4.8.3 Příklad. Je dán RTM M, kde $Q = \{q_0, q_1, q_2, q_3, q_f\}$, $\Gamma = \{0, 1, B\}$ a přechodová funkce δ je definována tabulkou:

		$0,0$	$1,0$	$0,1$	$1,1$	$B,0$	$B,1$
$\rightarrow$	q_0	$(q_1,0,R,S)$	$(q_2,1,R,S)$	$(q_3,0,S,R)$	$(q_3,1,S,R)$	–	–
	q_1	$(q_1,0,R,S)$	–	–	–	(q_4,B,S,S)	–
	q_2	–	$(q_2,1,R,S)$	–	–	(q_4,B,S,S)	–
	q_3	$(q_3,0,R,R)$	–	–	$(q_3,1,R,R)$	(q_4,B,S,S)	(q_4,B,S,S)
$\leftarrow$	q_4	–	–	–	–	–	–

Předpokládejme, že na vstupu má RTM M slovo w, pak:

- Jestliže první symbol druhé pásky je 0 (tj. náhodně jsme vygenerovali 0), M zkontroluje, zda $w = 0^n$ nebo $w = 1^n$ pro nějaké $n > 0$.
- Jestliže první symbol druhé pásky je 1 (tj. náhodně jsme vygenerovali 1), hlava na druhé pásce se posune doprava a M zkontroluje, zda se obsah druhé pásky od druhého políčka shoduje se vstupem w.

Nenastane-li ani jeden z předchozích případů, M se neúspěšně zastaví.

V případě RTM M není možné mluvit o tom, že M „přijme“ slovo w; můžeme pouze spočítat pravděpodobnost s jakou se M pro dané vstupní slovo w úspěšně zastaví, tj. zastaví v „přijímacím“ stavu q_f. V našem příkladě platí:

- Jestliže w je prázdné slovo, M se v q_f nikdy nezastaví (tj. pro žádný náhodný obsah druhé pásky).

- Jestliže $w = 0^n$ nebo $w = 1^n$ pro $n > 0$, M se zastaví v q_f s pravděpodobností

$$\frac{1}{2} + \frac{1}{2}\left(\frac{1}{2}\right)^n = \frac{1}{2} + 2^{-(n+1)}.$$

- Jestliže w je jiného tvaru, tj. obsahuje jak 0, tak 1, pak pravděpodobnost, že se M zastaví v q_f je

$$\frac{1}{2}\left(\frac{1}{2}\right)^{|w|} = 2^{-(|w|+1)}.$$

Přestože nemůžeme mluvit o „jazyku příjímaném RMT“, lze zavést několik různých tříd jazyků na základě randomizovaných Turingových strojů. První z nich je následující.

4.8.4 Třída $\mathcal{RP}$. Jazyk L patří do třídy $\mathcal{RP}$ právě tehdy, když existuje RTM M takový, že:

1. Jestliže $w \notin L$, stroj M se ve stavu q_f zastaví s pravděpodobností 0.
2. Jestliže $w \in L$, stroj M se ve stavu q_f zastaví s pravděpodobností, která je alespoň rovna $\frac{1}{2}$.
3. Existuje polynom $p(n)$ takový, že každý běh M (tj. pro jakýkoli obsah druhé pásky) trvá maximálně $p(n)$ kroků, kde n je délka vstupního slova.

4.8.5 Turingův stroj typu Monte-Carlo. RTM splňující podmínky 1 a 2 z předchozí definice 4.8.4, se nazývá RTM typu *Monte-Carlo*.

Uvědomte si, že RTM typu Monte-Carlo obecně nemusí pracovat v polynomiálním čase.

4.8.6 Tvrzení. Je dán jazyk $L \in \mathcal{RP}$, pak pro každou kladnou konstantu $0 < c < \frac{1}{2}$ je možné sestrojit RTM M (pravděpodobnostní algoritmus) s polynomiální složitostí a takový, že:

1. Jestliže $w \notin L$, stroj M se úspěšně zastaví (tj. zastaví se ve stavu q_f) s pravděpodobností 0.
2. Jestliže $w \in L$, stroj M se úspěšně zastaví (tj. zastaví se ve stavu q_f) s pravděpodobností aspoň $1 - c$.

Myšlenka zdůvodnění. Nový RTM N pracuje takto: původní randomizovaný Turingův stroj M (který existuje na základě faktu, že L je ve třídě $\mathcal{RP}$) nechá nový RTM N několikrát nezávisle běžet (počet běhů závisí na čísle c). V případě, že se při některém běhu M úspěšně zastaví, zastaví se i N úspěšně. Jestliže se při každém běhu M zastaví neúspěšně, N se zastaví neúspěšně. Dá se dokázat, že pro vhodný počet opakování, randomizovaný Turingův stroj N má požadované vlastnosti.

4.8.7 Třída $\mathcal{ZPP}$. Jazyk L patří do třídy $\mathcal{ZPP}$ právě tehdy, když existuje RTM M takový, že:

1. Jestliže $w \notin L$, stroj M se úspěšně zastaví (tj. zastaví se ve stavu q_f) s pravděpodobností 0.
2. Jestliže $w \in L$, stroj M se úspěšně zastaví (tj. zastaví se ve stavu q_f) s pravděpodobností 1.
3. Střední hodnota počtu kroků M v jednom běhu je $p(n)$, kde $p(n)$ je polynom a n je délka vstupního slova.

To znamená: M neudělá chybu, ale nezaručujeme vždy polynomiální počet kroků při jednom běhu, pouze střední hodnota počtu kroků je polynomiální.

4.8.8 Turingův stroj typu Las-Vegas. RTM splňující podmínky z předchozí definice 4.8.7, se nazývá typu *Las-Vegas*.

4.8.9 Tvrzení. Jestliže jazyk L patří do třídy $\mathcal{ZPP}$, pak i jeho doplněk $\overline{L}$ patří do třídy $\mathcal{ZPP}$.

Zdůvodnění. Stejný RTM M typu Las-Vegas slouží „k přijetí“ jak jazyka L, tak i jeho doplňku $\overline{L}$; stačí koncové (přijímající) stavy RTM M prohlásit za nekoncové a ze všech nekoncových stavů M udělat koncové.

4.8.10 Poznámka. Pro jazyky ze třídy $\mathcal{RP}$ se tvrzení obdobné 4.8.9 neumí dokázat. To motivuje následující třídu jazyků.

4.8.11 Třída co-$\mathcal{RP}$. Jazyk L patří do třídy co-$\mathcal{RP}$ právě tehdy, když jeho doplněk $\overline{L}$ patří do třídy $\mathcal{RP}$.

4.8.12 Věta.

$$\mathcal{ZPP} = \mathcal{RP} \cap \text{ co-}\mathcal{RP}.$$

Nástin důkazu. Ukážeme nejprve $\mathcal{RP} \cap$ co-$\mathcal{RP} \subseteq \mathcal{ZPP}$.

Předpokládejme, že jazyk L leží v obou třídách $\mathcal{RP}$ i co-$\mathcal{RP}$. Existují proto dva RTM M_1 a M_2 typu Monte Carlo pracující v polynomiálním čase a to M_1 – pro jazyk L a M_2 – pro jazyk $\overline{L}$.

Označme $p(n)$ ten větší z polynomů, které určují počet kroků M_1 a M_2. Vytvoříme novýRTM M typu Las-Vegas pro jazyk L takto: Pro dané vstupní slovo w

1. M nechá pracovat M_1 po dobu $p(n)$ kroků. Jestliže M_1 úspěšně skončí, M také skončí **úspěšně**.
2. M nechá pracovat M_2 po dobu $p(n)$ kroků. Jestliže M_2 úspěšně skončí, M skončí **neúspěšně**.
3. Jestliže M neskončí ani v kroku 1 ani v kroku 2, M pokračuje krokem 1.

Dá se dokázat, že RTM M je typu Las-Vegas.

Nyní ukážeme, že $\mathcal{ZPP} \subseteq \mathcal{RP} \cap$ co-$\mathcal{RP}$.

Předpokládejme, že jazyk L leží ve třídě $\mathcal{ZPP}$, existuje tedy pro něj RTM M_1 typu Las-Vegas. Označme $p(n)$ polynom, který udává střední hodnotu počtu kroků RTM M_1 pro vstupní slovo délky n. Vytvoříme RTM M typu Monte Carlo pracující polynomiálním čase pro jazyk L.

M nechá na vstupu w pracovat RTM M_1 po dobu $2p(n)$. Jestliže M_1 úspěšně skončí, M úspěšně skončí; ve všech ostatních případech RTM M skončí neúspěšně.

Dá se dokázat, že M splňuje všechny podmínky pro RTM typu Monte Carlo. Protože pracuje v čase $2p(n)$, jedná se o polynomiální RTM typu Monte Carlo. Proto je jazyk L ve třídě $\mathcal{RP}$.

Protože třída $\mathcal{ZPP}$ je uzavřena na doplňky, je každý jazyk ze třídy $\mathcal{ZPP}$ také ve třídě co-$\mathcal{RP}$.

4.8.13 Věta. Platí

$$\mathcal{P} \subseteq \mathcal{ZPP}, \;\; \mathcal{RP} \subseteq \mathcal{NP}, \;\; \text{co-}\mathcal{RP} \subseteq \text{co-}\mathcal{NP}.$$

Myšlenka zdůvodnění. První inkluze je zřejmá, každý deterministický Turingův stroj pracující v polynomiálním čase můžeme považovat za randomizovaný Turingův stroj typu Las-Vegas.

Druhá inkluze je trochu složitější. Její důkaz spočívá v tom, že pro daný polynomiální RTM M typu Monte Carlo pracující v polynomiálním čase zkonstruujeme nedeterministický Turingův stroj, který přijímá jazyk $L(M)$ a to tak, že kdykoli by se přechodová funce M lišila pro různé obsahy náhodné pásky, povolíme nedeterministickému Turingovu stroji oba přechody.

Třetí inkluze jednoduše vyplývá z definic tříd co-$\mathcal{RP}$, co-$\mathcal{NP}$ a z druhé inkluze.