1.7.5 Třída $\mathcal{NP}$. Řekneme, že rozhodovací úloha $\mathcal{U}$ leží ve třídě $\mathcal{NP}$, jestliže existuje nedeterministický Turingův stroj, který rozhodne jazyk $L_{\mathcal{U}}$ a pracuje v polynomiálním čase.

1.7.6 Poznámka. V definice 1.7.3 jsme místo existence Turingova stroje mohli požadovat existenci programu P pro RAM, který řeší $\mathcal{U}$ v polynomíálním čase. Abychom přiblížili, které jazyky (rozhodovací úlohy) leží ve třídě $\mathcal{NP}$, zavedeme pojem nedeterministického algoritmu jako analogii RAM.

1.7.7 Nedeterministický algoritmus pracuje ve dvou fázích,

1. Algoritmus náhodně vygeneruje řetězec s.
2. Deterministický algoritmus (program pro RAM) na základě vstupu a řetězce s dá odpověď ANO nebo NEVIM.

Řekneme, že nedeterministický algoritmus řeší úlohu $\mathcal{U}$, jestliže

1. Pro každou ANO instanci úlohy $\mathcal{U}$ existuje řetězec s, na jehož základě algoritmus dá odpověď ANO.

2. Pro žádnou NE instanci úlohy $\mathcal{U}$ neexistuje řetězec s, na jehož základě algoritmus dá odpověď ANO.

Řekneme, že nedeterministický algoritmus *pracuje v čase* $\mathcal{O}(T(n))$, jestliže každý průchod oběma fázemi 1 a 2 pro instanci velikosti n potřebuje $\mathcal{O}(T(n))$ kroků.

1.7.8 Poznámka. Fakt, že nedeterministický algoritmus pracuje v polynomiálním čase, znamená, že každá z fází vyžaduje polynomiální čas a tudíž i řetězec s musí mít polynomíální délku (vzhledem k velikosti instance).

V definice 1.7.5 jsme místo existence nedeterministického Turingova stroje mohli požadovat existenci nedeterministického algoritmu, který řeší úlohu $\mathcal{U}$ v polynomiálním čase.

1.7.9 Příklady $\mathcal{NP}$ úloh.

- Kliky v grafu. Je dán neorientovaný graf G a číslo k. Existuje klika v grafu G o alespoň k vrcholech?
- Nejkratší cesty v obecném grafu. Je dán orientovaný graf s ohodnocením hran a. Jsou dány vrcholy r a v. Je dáno číslo k. Existuje orientovaná cesta z vrcholu r do vrcholu v délky menší nebo rovno k?
- k-barevnost. Je dán neorientovaný graf G. Je graf G k-barevný?
- Problém batohu. Je dáno n předmětů $1, 2, \ldots, n$. Každý předmět i má cenu c_i a váhu w_i. Dále jsou dána čísla A a B. Je možné vybrat předměty tak, aby celková váha nepřevýšila A a celková cena byla alespoň B? Přesněji, existuje podmnožina předmětů $I \subseteq \{1, 2, \ldots, n\}$ taková, že

$$\sum_{i\in I} w_i \leq A \quad \text{a} \quad \sum_{i\in I} c_i \geq B?$$

1.8 Třída $\mathcal{NPC}$

1.8.1 Redukce a polynomiální redukce úloh. Jsou dány dvě rozhodovací úlohy $\mathcal{U}$ a $\mathcal{V}$. Řekneme, že úloha $\mathcal{U}$ se *redukuje* na úlohu $\mathcal{V}$, jestliže existuje algoritmus (program pro RAM, Turingův stroj) M, který pro každou instanci I úlohy $\mathcal{U}$ zkonstruuje instanci I' úlohy $\mathcal{V}$ a to tak, že

$$I \text{ je ANO-instance } \mathcal{U} \text{ iff } I' \text{ je ANO-instance } \mathcal{V}.$$

Fakt, že úloha $\mathcal{U}$ se redukuje na úlohy $\mathcal{V}$ značíme

$$\mathcal{U} \lhd \mathcal{V}.$$

Jestliže navíc, algoritmus M pracuje v polynomiálním čase, říkáme, že $\mathcal{U}$ se *polynomiálně* redukuje na $\mathcal{V}$ a značíme

$$\mathcal{U} \lhd_p \mathcal{V}.$$

Fakt, že se úloha $\mathcal{U}$ redukuje na úlohu $\mathcal{V}$ zhruba řečeno znamená, že $\mathcal{U}$ není obtížnější než $\mathcal{V}$.

1.8.2 Tvrzení. Jsou dány tři rozhodovací úlohy $\mathcal{U}$, $\mathcal{V}$ a $\mathcal{W}$. Jestliže platí

$$\mathcal{U} \lhd_p \mathcal{V} \text{ a } \mathcal{V} \lhd_p \mathcal{W}, \quad \text{pak} \quad \mathcal{U} \lhd_p \mathcal{W}.$$

1.8.3 $\mathcal{NP}$ úplné úlohy. Řekneme, že rozhodovací úloha $\mathcal{U}$ je $\mathcal{NP}$ *úplná*, jestliže

1. $\mathcal{U}$ je ve třídě $\mathcal{NP}$;
2. každá $\mathcal{NP}$ úloha se polynomiálně redukuje na $\mathcal{U}$.

Třída všech $\mathcal{NP}$ úplných úloh se značí $\mathcal{NPC}$.

Zhruba řečeno, $\mathcal{NP}$ úplné úlohy jsou ty „nejtěžší“ mezi všemi $\mathcal{NP}$ úlohami.

1.8.4 Tvrzení. Jsou dány dvě $\mathcal{NP}$ úlohy $\mathcal{U}$ a $\mathcal{V}$, pro které platí $\mathcal{U} \lhd_p \mathcal{V}$. Pak

1. jestliže $\mathcal{V}$ je ve třídě $\mathcal{P}$, pak také $\mathcal{U}$ je ve třídě $\mathcal{P}$;
2. jestliže $\mathcal{U}$ je $\mathcal{NP}$ úplná úloha, pak také $\mathcal{V}$ je $\mathcal{NP}$ úplná úloha.

1.8.5 Tvrzení. Kdyby by některá $\mathcal{NP}$ úplná úloha patřila do třídy $\mathcal{P}$ (tj. byla by polynomiálně řešitelná), pak $\mathcal{P} = \mathcal{NP}$. Jinými slovy, každá $\mathcal{NP}$ úloha by byla polynomiálně řešitelná.

1.8.6 $\mathcal{NP}$ obtížné úlohy. Jestliže o některé úloze $\mathcal{U}$ pouze víme, že se na ní polynomiálně redukuje některá $\mathcal{NP}$ úplná úloha, pak říkáme, že $\mathcal{U}$ je $\mathcal{NP}$ těžká, nebo též $\mathcal{NP}$ obtížná. Poznamenejme, že to vlastně znamená, že $\mathcal{U}$ je alespoň tak těžká jako všechny $\mathcal{NP}$ úlohy.

1.8.7 Cookova věta. Úloha CNF SAT, splňování formulí v konjunktivním normálním tvaru, je $\mathcal{NP}$ úplná úloha.

1.8.8 Myšlenka důkazu. Není těžké se přesvědčít, že úloha CNF SAT je ve třídě $\mathcal{NP}$. První fáze nedeterministického algoritmu vygeneruje ohodnocení logických proměnných a na základě tohoto ohodnocení jsme schopni v polynomiálním čase ověřit, zda je v tomto ohodnocení formule pravdivá nebo ne.

Druhá část důkazu spočívá v popisu práce Turingova stroje formulí výrokové logiky. Načrtneme základní myšlenku tohoto popisu.

Je dán nedeterministický Turingův stroj M s množinou stavů Q, vstupní abecedou Σ, páskovou abecedou Γ, přechodovou funkcí δ, počátečním stavem q_0 a koncovým stavem q_f. Předpokládejme, že M příjímá slovo w a potřebuje přitom $p(n)$ kroků.

Zavedeme logické proměnné:

$h_{i,j}$, $i = 0, 1, \ldots, p(n)$, $j = 1, 2, \ldots, p(n)$;

fakt, že hodnota proměnné $h_{i,j}$ je rovna 1 znamená, že hlava Turingova stroje v čase i čte j-té pole pásky.

s_i^q, $i = 0, 1, \ldots, p(n)$, $q \in Q$;

fakt, že hodnota proměnné s_i^q je rovna 1 znamená, že Turingův stroj v čase i je ve stavu q.

$t_{i,j}^A$, $i = 0, 1, \ldots, p(n)$, $j = 1, 2, \ldots, p(n)$, $A \in \Gamma$;

fakt, že hodnota proměnné $t_{i,j}^A$ rovna 1 znamená, že v čase i v j-tém poli pásky je páskový symbol A.

Nyní je třeba formulemi popsat následující fakta:

1. V každém okamžiku je Turingův stroj v právě jednom stavu.
2. V každém okamžiku čte hlava Turingova stroje právě jedno pole vstupní pásky.
3. V každém okamžiku je na každém poli pásky Turingova stroje právě jeden páskový symbol.
4. Na začátku práce (tj. v čase 0) je Turingův stroj ve stavu q_0, hlava čte první pole pásky a na pásce je na prvních n polích vstupní slovo, ostatní pole pásky obsahují B.
5. Krok Turingova stroje je určen přechodovou funkcí, tj. stav stroje, obsah čteného pole a poloha hlavy v čase $i+1$ je dána přechodovou funkcí.
6. V polích pásky, které v čase i hlava nečte, je obsah v čase $i+1$ stejný jako v i.
7. Na konci práce Turingova stroje, tj. v čase $p(n)$, je stroj ve stavu q_f.

Ukážeme jak utvořit formule pro body 1, 4, 5, 6 a 7.

Bod 1. V okamžiku i je Turingův stroj v aspoň jednom stavu:

$$\bigvee_{q \in Q} s_i^q.$$

V okamžiku i Turingův stroj není ve dvou různých stavech:

$$\bigwedge_{q \neq q'} (\neg s_i^q \vee \neg s_i^{q'}).$$

Nyní fakt, že Turingův stroj je v okamžíku i právě jednom stavu je konjunkce obou výše uvedených formulí:

$$(\bigvee_{q \in Q} s_i^q) \wedge \bigwedge_{q \neq q'} (\neg s_i^q \vee \neg s_i^{q'}).$$

Bod 4. Na začátku práce (tj. v čase 0) je Turingův stroj ve stavu q_0, hlava čte první pole pásky a na pásce je na prvních n polích vstupní slovo $a_1 a_2 \dots a_n$, ostatní pole obsahují B.

$$s_0^{q_0} \wedge h_{0,1} \wedge t_{0,1}^{a_1} \wedge \dots \wedge t_{0,n}^{a_n} \wedge t_{0,n+1}^B \wedge \dots \wedge t_{0,p(n)}^B.$$

Bod 5. Jestliže Turingův stroj je v čase i ve stavu q, hlava je na j-tém poli pásky, hlava čte páskový symbol A a $\delta(q, A)$ se skládá z trojic (p, C, D) (zde $D = 1$ znamená posun hlavy doprava, $D = -1$ znamená posun hlavy doleva), pak formule má tvar:

$$\bigwedge_j \bigwedge_{A \in \Gamma} ((s_i^q \wedge h_{i,j} \wedge t_{i,j}^A) \Rightarrow \bigvee (s_{i+1}^p \wedge t_{i+1,j}^C \wedge h_{i+1,j+D})).$$

Bod 6. Obsah polí kromě j-tého zůstává v čase $i+1$ stejný:

$$\bigwedge_j \bigwedge_{A \in \Gamma} ((\neg h_{i,j} \wedge t_{i,j}^A) \Rightarrow t_{i+1,j}^A).$$

Bod 7. Na konci práce Turingova stroje, tj. v čase $p(n)$ je stroj ve stavu q_f.

$$s_{p(n)}^{q_f}.$$

Výslednou formuli dostaneme jako konjunkci všech dílčích formulí pro všechny časové okamžiky $i = 0, 1, \dots, p(n)$.