Sto łatwych zadań
z języków formalnych
i złożoności obliczeniowej

przygotował Jerzy Marcinkowski
jako materiał do kursu prowadzonego
w Instytucie Informatyki UWr
wiosną roku 2007

1 Deterministyczne Automaty Skończone

Zadanie 1. Rozważmy język $L = \{w0s : |s| = 9\}$, złożony z tych słów których dziesąty symbol od końca to 0. Udowodnij że DFA rozpoznający ten język ma co najmniej 1024 stany.

Zadanie 2. udowodnij, że język $L = \{a^n b^{2n} : n \in \mathcal{N}\}$ nie jest regularny.

Zadanie 3. Udowodnij że zbiór L tych słów nad alfabetem $\{0, 1\}$ które są zapisem binarnym liczby pierwszej nie jest regularny.

Zadanie 4. (za 2 punkty) Niech L będzie dowolnym podzbiorem 0^*. Udowodnij, że L^* jest językiem regularnym.

Definicja Dla danego słowa w, nad pewnym ustalonym alfabetem, niech w^R oznacza "w czytane od końca" (tzn. $\varepsilon^R = \varepsilon$ i $(aw)^R = w^R a$ jeśli a należy do alfabetu a w jest dowolnym słowem).

Zadanie 5. Czy język $\{ww^Rx : w, x \in \{0, 1\}^* \text{ i } w, x \neq \varepsilon\}$ jest regularny ?
Czy język $\{xwx : w, x \in \{0, 1\}^* \text{ i } x \neq \varepsilon\}$ jest regularny ?

Zadanie 6. Niech $L \subset \mathcal{A}^*$. Relację $\sim_L \subset \mathcal{A}^* \times \mathcal{A}^*$ definiujemy w sposób następujący: $w \sim_L w'$ w.t.w. gdy $\forall v \in \mathcal{A}^* \;\; wv \in L \Leftrightarrow w'v \in L$. Udowodnij następujace *Twierdzenie o indeksie*: L jest regularny wtedy i tylko wtedy gdy liczba klas abstrakcji relacji $\sim_L$ jest skończona. Minimalna ilość stanów DFA rozpoznającego L jest wtedy równa liczbie tych klas abstrakcji.

Zadanie 7. Jaką minimalną liczbę stanów musi mieć deterministyczny automat skończony rozpoznający zbiór tych wszystkich słów nad alfabetem $\{a, b, c\}$, które wśród ostatnich trzech znaków mają po jednym wystąpieniu każdej z liter alfabetu?

Zadanie 8. Jaką minimalną liczbę stanów musi mieć deterministyczny automat skończony, rozpoznający język tych słów nad alfabetem $\{a, b, c\}$, które mają przynajmniej 4 symbole, i których ostatnie 4 symbole są jednakowe?

Zadanie 9. Jaką minimalną liczbę stanów musi mieć deterministyczny automat skończony, rozpoznający język tych słów nad alfabetem zerojedynkowym, które wśród czterech ostatnich symboli mają dokładnie dwie jedynki (słowo 010 ma jedną jedynkę wsród czterech ostatnich symboli)?

Zadanie 10. Rozważmy alfabet A_n składający sie z liter $a, b_1, b_2, \ldots b_n$. Niech język L_n^1 składa się z tych słów nad A_n, które mają parzystą ilość wystąpień wzorca $b_1 b_2$., niech język L_n^2 składa się z tych słów nad A_n, które mają parzystą ilość wystąpień wzorca $b_2 b_3$, i tak dalej. Niech wreszcie język L_n^n składa się z tych słów nad A_n, które mają parzystą ilość wystąpień wzorca $b_n b_1$. Zdefiniujmy język L_n jako przecięcie $L_n^1 \cap L_n^2 \cap \ldots \cap L_n^n$. Jaką minimalną ilość stanów musi mieć deterministyczny automat skończony rozpoznający L_n?

Zadanie 11. Przez *DFA ze stacją benzynową* będziemy w tym zadaniu rozumieć zwykły deterministyczny automat skończony wraz z jednym dodatkowym wyróżnionym stanem q_{BP} i liczbą naturalną b. Powiemy że DFA ze stacją benzynową M akceptuje słowo wejściowe w, jeśli M akceptuje w w zwykłym sensie, a spośród każdych b kolejnych stanów napotkanych w czasie wczytywania w przez M przynajmniej jeden jest stanem q_{BP}. Udowodnij, że nie każdy język regularny może być rozpoznany przez jakiś DFA ze stacją benzynową.

2 Wyrażenia regularne

Zadanie 12. Skonstruuj automat skończony rozpoznający i wyrażenie regularne definiujące nad alfabetem $\{1, 0\}$ język słów, które zawierają tyle samo jedynek co zer i w których każdym prefiksie ilość zer różni się co najwyżej o dwa od ilości jedynek.

Zadanie 13. Skonstruuj automat skończony rozpoznający i wyrażenie regularne definiujące nad alfabetem $\{a, b\}$ język słów które nie zawierają wzorca $baba$.

Zadanie 14. Dodanie do definicji wyrażeń regularnych pozwolenia na użycie symbolu $\cap$, oznaczającego przekrój języków nie umożliwia reprezentowania nowych zbiorów, wyrażenia jednak stają się krótsze.

Udowodnij że użycie $\cap$ może wykładniczo skrócić wyrażenie.

Wskazówka: rozważyć język składający się z jednego słowa $(\ldots(a_0 a_1)^2 a_2)^2 \ldots)^2$

Zadanie 15. Skonstruuj automat skończony rozpoznający i wyrażenie regularne definiujące nad alfabetem $\{a, b, c, d\}$ język słów które zawierają tyle samo symboli a co b, tyle samo symboli c co d, i w których każdym prefiksie ilość symboli a różni się co najwyżej o jeden od ilości symboli b zaś ilość symboli c różni się co najwyżej o jeden od ilości symboli d.

Niech ϕ bąŞdzie wyrażeniem regularnym nad alfabetem $\mathcal{A}$, a w słowem nad tym alfabetem. Niech f będzie funkcją, której argumentami są wystąpienia liter alfabetu w słowie w (czyli "kolejne litery słowa w"), a wartościami są wystąpienia liter w wyrażeniu ϕ. Powiemy, że f jest **poprawnym mapowaniem** w na ϕ, jeśli zachodzi któryś z warunków:

1. ϕ jest słowem nad $\mathcal{A}$, $\phi = w$ i f jest identycznością, lub $\phi = \varepsilon$ i w jest puste;
2. $\phi = \phi_1 + \phi_2$ i f jest poprawnym mapowaniem w na ϕ_1 lub f jest poprawnym mapowaniem w na ϕ_2;
3. $\phi = \phi_1 \phi_2$, $w = w_1 w_2$ i f ograniczona do w_1 jest poprawnym mapowaniem tego słowa na ϕ_1, zaś f ograniczona do w_2 jest poprawnym mapowaniem tego słowa na ϕ_2;
4. $\phi = \psi^*$, $w = w_1 w_2 \ldots w_k$, dla jakiegoś $k \geq 0$ i dla każdego $1 \leq i \leq k$ funkcja f ograniczona do w_i jest poprawnym mapowaniem w_i na ψ.

Intuicja jest taka, że poprawne mapowanie słowa przyporządkowuje każdej jego literze, literę wyrażenia z której ta litera słowa "się wzięła". Wyrażenie ϕ jest **deterministycznym wyrażeniem regularnym**, jeśli dla każdego $w \in L_\phi$ istnieje dokładnie jedno poprawne mapowanie w na ϕ. Deterministyczne wyrażenie regularne pozwala odczytać, które litery w słowie biorą sie z których liter w wyrażeniu, ale to odczytanie nastąpuje dopiero, gdy znamy całe słowo. Inaczej jest dla deterministycznych on-line wyrażeń regularnych. Wyrażenie regularne ϕ jest **deterministyczne on-line**, jeśli dla każdych słów $ww_1, ww_2 \in L_\phi$ i każdych funkcji f_1, f_2, będących poprawnymi mapowaniami słów (odpowiednio) ww_1 i ww_2 na ϕ, funkcje f_1 i f_2 zgadzają się na prefiksie w.

UWAGA: deterministic regular expressions, znane również jako unambiguous regular expression pojawiają sie w definicji standardu XML.

Zadanie 16. **A.** Które z poniższych wyrażeń są deterministyczne a które są deterministyczne on-line?
i. $0^*10^* + 0^*$ **ii.** $(0+1)^*1(0+1)$ **iii.** $(0+1)(0+2)^* + (1+2)(0+1)^* + (0+2)(1+2)^*$
B. Znajdź deterministyczne wyrażenie regularne oznaczające język tych wszystkich słów nad alfabetem zerojedynkowym, które zawierają wzorzec 101.

Zadanie 17. Czy dla każdego języka regularnego istnieje deterministyczne on-line wyrażenie regularne, które go definiuje?

Zadanie 18. Znajdź deterministyczne on-line wyrażenie regularne oznaczające język tych wszystkich słów nad alfabetem zerojedynkowym, które zawierają jedną lub dwie jedynki.

3 Niedeterministyczne Automaty Skończone

Zadanie 19. Skonstruuj niedeterministyczny automat skończony rozpoznający język tych słów nad $\{0,1\}^*$ które jako liczba w systemie dwójkowym dzielą się przez 5, przy czym liczba jest wczytywana

a) począwszy od najbardziej znaczącego bitu,
b) począwszy od najmniej znaczącego bitu.

Zadanie 20. Udowodnij że jeśli dla pewnego języka L istnieje rozpoznający go NDFA, to istnieje również NDFA rozpoznajacy język $L^R = \{w : w^R \in L\}$.

Zadanie 21. Wiadomo że L językiem regularnym. Pokaż, że język

$\{w : \exists n \in \mathcal{N} \exists v \in L \ \ w^n = v\}$

jest językiem regularnym. Przez w^n rozumiemy tu słowo w skonkatenowane ze sobą n razy.

Zadanie 22. Udowodnij, że jeśli L_1 i L_2 są językami regularnymi nad pewnym alfabetem $\mathcal{A}$ to również języki $L_1 \cup L_2$, $L_1 \cap L_2$ i $\mathcal{A}^* - L_1$ są językami regularnymi.

Zadanie 23. (za 2 punkty)
Załóżmy że L jest pewnym językiem regularnym.

Czy język $L/2 = \{w : \exists v \ \ vw \in L \wedge |v| = |w|\}$ jest regularny ?

Zadanie 24. Podaj algorytm rozstrzygający, dla danych dwóch niedeterministycznych automatów skończonych, czy języki rozpoznawane przez te automaty są równe.

Zadanie 25. Minimalny DFA rozpoznający jezyk L ma zawsze tyle samo stanów co minimalny DFA rozpoznający dopełnienie L. Stwierdzenie to przestaje być prawdziwe, jeśli rozważamy automaty niedeterministyczne. Udowodnij, że istnieje język L który daje się rozpoznać NDFA o mniej niz 20 stanach, ale którego dopełnienie nie daje się rozpoznać żadnym NDFA o mniej niż 200 stanach. *Wskazówka: wystarczy rozważać alfabet jednoelementowy.*

Zadanie 26. Czy istnieje wyrażenie regularne ϕ, takie że $L_{a\phi} = L_{\phi b}$? Czy istnieje wyrażenie regularne ϕ, takie że $L_{a^*\phi} = L_{\phi b^*}$?

Zadanie 27. Przez *NDFA ze stacją benzynową* będziemy w tym zadaniu rozumieć zwykły niedeterministyczny automat skończony, wraz z jednym dodatkowym wyróżnionym stanem q_{BP} i liczbą naturalną b. Powiemy że NDFA ze stacją benzynową M akceptuje słowo wejściowe w, jeśli istnieje taka scieżka akceptująca słowo w w M w zwykłym sensie, że spośród każdych b kolejnych stanów na tej ścieżce, przynajmniej jeden jest stanem q_{BP}. Udowodnij, że nie każdy język regularny może być rozpoznany przez jakiś NDFA ze stacją benzynową.

4 Gramatyki bezkontekstowe i automaty ze stosem

Zadanie 28. Zbuduj automat ze stosem rozpoznający *język dobrze rozstawionych nawiasów dwóch rodzajów* generowany przez gramatykę:
$S \longrightarrow \ SS|(S)|[S]|\varepsilon$
Która ma jeden symbol nieterminalny S i cztery symbole terminalne (,), [,].

Zadanie 29. Zbuduj gramatykę bezkontekstową generujacą język: $\{w \in \{0,1\}^* : |w|_0 = 2|w|_1 \wedge 2||w|_1\}$.

Zadanie 30. Czy język $\{w \in \{0,1\}^* : \exists n \in \mathcal{N} \ \ 2n|w|_0 \leq |w|_1 \leq (2n+1)|w|_0\}$ jest bezkontekstowy ?

Zadanie 31. Podaj algorytm rozstrzygajacy, dla danej gramatyki bezkontekstowej G, czy $L(G)$ jest niepusty.

Zadanie 32. Niech G będzie gramatyką generujacą poprawnie zbudowane formuły rachunku zdań ze zmiennymi zdaniowymi p i q. Symbolami terminalnymi G są $p, q, (,), \neg$ i $\Rightarrow$ a produkcjami
$S \rightarrow \neg S|(S \Rightarrow S)|p|q$
Znajdź gramatykę w postaci normalnej Chomsky'ego generującą ten sam język.

Zadanie 33. Czy istnieje gramatyka bezkontekstowa generująca zbiór tych slów nad alfabetem zerojedynkowym, które nie są postaci vww dla żadnych słów w, v takich, że $|v| = |w|$?

Zadanie 34. Czy język L złożony z tych wszystkich słów nad alfabetem $\{0,1\}$ które są postaci wvw, dla pewnych słów w, v, takich, że $|w| = |v|$, jest bezkontekstowy?

Zadanie 35. Czy język $\bar{L}$ bedący dopełnieniem języka L z poprzedniego zadania do $\{0,1\}^*$ jest bezkontekstowy?

Zadanie 36. Czy język

$$L_3 = \{w \in \{0,1,2\}^* : \neg\exists x \in \{0,1,2\}^* \ \ w = xx\}$$

jest bezkontekstowy ?

Zadanie 37. Czy dopełnienie języka L_3, język

$$L_4 = \{w \in \{0,1,2\}^* : \exists x \in \{0,1,2\}^* \;\; w = xx\}$$

jest bezkontekstowy ? *Wskazówka. (1) Rozważ język* $L_4 \cap L$ *gdzie* $L = L_{0^*10^*10^*10^*1}$. *(2) Skorzystaj z lematu o pompowaniu, pamiętaj że podział* $w = xyztr$ *którego istnienie postuluje lemat jest taki, że* $|yzt| \leq c$*, gdzie* c *jest stałą z lematu.*

Zadanie 38. Zbuduj NDPDA i gramatykę bezkontekstową dla języka $\{0,1\}^* - \{www : w \in \{0,1\}^*\}$.

Zadanie 39. Czy język: $\{vww : v, w \in \{a,b,c\}^*, w \neq \varepsilon\}$ jest bezkontekstowy ?

Zadanie 40. Czy język $\{w \in \{0,1\}^* : |w|_0 \leq |w|_1 \leq 2|w|_0\}$ jest bezkontekstowy ?

Zadanie 41. Niech $L_=$ będzie językiem tych słów nad alfabetem zerojedynkowym, które mają tyle samo zer co jedynek, a L_R niech będzie językiem wszystkich palindromów. Czy przekrój $L_=$ z dopełnieniem L_R jest językiem bezkontekstowym? (mamy tu na myśli dopełnienie do $\{0,1\}^*$)

Zadanie 42. Niech $L_=$ będzie językiem tych słów nad alfabetem zerojedynkowym, które mają tyle samo zer co jedynek, a L_R niech będzie językiem wszystkich palindromów. Czy przekrój $L_=$ z L_R jest jezykiem bezkontekstowym?

5 Zbiory i funkcje rekurencyjne

Zadanie 43. Rozszerz definicję zbioru rekurencyjnego tak aby można było rozważać rekurencyjne zbiory par liczb naturalnych i udowodnij, że jeśli zbiór $A \subset \mathcal{N}^2$ jest rekurencyjny to zbiór $\{n : \exists m \;\; [n,m] \in A\}$, czyli rzut A na pierwszą oś, jest zbiorem rekurencyjnie przeliczalnym.

Zadanie 44. Pokaż, że każdy zbiór rekurencyjnie przeliczalny jest rzutem pewnego zbioru rekurencyjnego, to znaczy jeśli B jest r.e. to istnieje taki rekurencyjny $A \subset \mathcal{N}^2$ rekurencyjny, że $B = \{n : \exists m \;\; [n,m] \in A\}$.

Zadanie 45. Powtórz podany na wykładzie dowód nierozstrzygalności problemu stopu, to znaczy faktu, że zbiór $K = \{n : \phi_n(n) \in \mathcal{N}\}$ nie jest rekurencyjny.

Zadanie 46. Pokaż, że $\{n : |Dom(\phi_n)| \geq 7\}$ jest rekurencyjnie przeliczalny.

Zadanie 47. Udowodnij, że jeśli ϕ jest niemalejącą całkowitą funkcją rekurencyjną to zbiór $\phi(\mathcal{N})$ jej wartości jest rekurencyjny. Czy pozostaje to prawdą bez założenia o całkowitości ϕ ?

Zadanie 48. Udowodnij, że każdy niepusty zbiór rekurencyjnie przeliczalny jest postaci $\phi(\mathcal{N})$ dla pewnej całkowitej funkcji rekurencyjnej ϕ.

Zadanie 49. Udowodnij, że każdy nieskończony zbiór rekurencyjnie przeliczalny jest postaci $\phi(\mathcal{N})$ dla pewnej całkowitej różnowartościowej funkcji rekurencyjnej ϕ.

Zadanie 50. Udowodnij, że zbiór $\{n : Dom(\phi_n) = \mathcal{N}\}$ nie jest rekurencyjnie przeliczalny.

Zadanie 51. (długie, więc za 2 punkty)
Załóżmy, że f jest funkcją rekurencyjną całkowitą. Które z poniższych implikacji sa prawdziwe?
(i) jeśli A jest rekurencyjny to $f(A)$ też;
(ii) jeśli A jest rekurencyjny to $f^{-1}(A)$ też;
(iii) jeśli A jest r.e. to $f(A)$ też;
(iii) jeśli A jest r.e. to $f^{-1}(A)$ też;
Co zmieni się, jeśli założymy, że f jest funkcją częściową ?

Zadanie 52. Nie korzystając z tw. Rice'a udowodnij, że zbiór $B = \{n : Dom(\phi_n)$ i $\mathcal{N} - Dom(\phi_n)$ są nieskończone$\}$ nie jest rekurencyjny.

Zadanie 53. Udowodnij, że zbiór $B = \{n : Dom(\phi_n)$ i $\mathcal{N} - Dom(\phi_n)$ są nieskończone$\}$ z poprzedniego zadania nie jest nawet rekurencyjnie przeliczalny.

Zadanie 54. Niech A, B, C, D będą zbiorami rekurencyjnie przeliczalnymi, takimi że każda liczba naturalna należy do dokładnie dwóch z nich. Udowodnij że w takim razie wszystkie te cztery zbiory są rekurencyjne.

Zadanie 55. Udowodnij, że zbiór numerów tych programów które zatrzymują się dla wszystkich argumentów oprócz co najwyżej skończonej ilości, nie jest rekurencyjnie przeliczalny.

Zadanie 56. Udowodnij podane na wykładzie twierdzenie Rice'a.

Zadanie 57. Poniższe zbiory nie są rozstrzygalne:

1. zbiór numerów tych maszyn Turinga które obliczają funkcje o dziedzinie różnej od $\mathcal{N}$;
2. zbiór numerów tych maszyn Turinga które obliczają funkcję całkowitą i których czas działania jest rosnący względem rozmiaru danych;
3. zbiór numerów tych maszyn Turinga których czas działania dla żadnych danych jest nie liczbą pierwszą;
4. zbiór numerów tych maszyn Turinga które obliczają funkcje częściowe, których wartościami są wyłącznie liczby pierwsze;

Nierozstrzygalność których z nich daje się udowodnić wprost z twierdzenia Rice'a?

Zadanie 58. Udowodnij nierozstrzygalność zbioru z punktu 2. poprzedniego zadania.

Zadanie 59. Niech $A, B \subset \mathcal{N}$. Mówimy, że $A \leq_{rek} B$ jeśli istnieje całkowita funkcja rekurencyjna f (zwana redukcją) taka że $f(x) \in B$ wtedy i tylko wtedy gdy $x \in A$. Pokaż, że dla każdych dwóch zbiorów $A, B \subset \mathcal{N}$ istnieje ich najmniejsze ograniczenie górne w sensie $<_{rek}$, to znaczy taki zbiór C, że:
(i) $A \leq_{rek} C$ i $B \leq_{rek} C$,
(ii) jeśli D jest taki, że $A \leq_{rek} D$ i $B \leq_{rek} D$ to $C \leq_{rek} D$.

Zadanie 60. (za 2 punkty)
Czy $K \leq_{rek} \bar{K}$? Czy $\bar{K} \leq_{rek} K$?

Zadanie 61. Pokaż, że zbiór:
$\{n : \phi_n$ zatrzymuje się dla wszystkich danych po czasie mniejszym niż dwukrotna długość danych$\}$
nie jest rekurencyjny. Czy można do tego użyć tw. Rice'a?

Zadanie 62. Niech T będzie zbiorem tych par liczb $\langle n, m\rangle$ dla ktorych ϕ_m i ϕ_n to ta sama funkcja częściowa.

a. Pokaż, że T nie jest rekurencyjnie przeliczalny.

b. Czy dopełnienie zbioru T jest rekurencyjnie przeliczalne?

6 Maszyny Turinga

Zadanie 63. Udowodnij, że zastąpienie w definicji maszyny Turinga taśmy nieskończoną płaszczyzną nie zmieni klasy funkcji obliczalnych (w tym i następnych zadaniach należy dość dokładnie podać ideę konstrukcji, ale nie wymagam napisania odpowiedniego kompilatora).

Zadanie 64. Skonstruuj maszynę Turinga rozpoznającą język $\{ww^R : w \in \{0,1\}^*\}$.

Zadanie 65. Skonstruuj dwutaśmową maszynę Turinga która mając jako dane na jednej taśmie pewne słowo w zaś na drugiej taśmie słowo v postaci

$$\#q_{i_1}a_1q'_{i_1}a'_1D_1\#q_{i_2}a_2q'_{i_2}a'_2D_2\#\ldots\#q_{i_n}a_nq'_{i_n}a'_nD_n\#q_0\#q_F$$

(gdzie każde D_i jest albo R albo L) zwróci rezultat 1 wtedy i tylko wtedy gdy jednotaśmowa maszyna której ciągiem instrukcji jest słowo v (stanem początkowym jest q_0 a końcowym q_F) zwróci rezultat 1.

Zadanie 66. (za 2 punkty) Zauważ, że maszyna podobna do dwukierunkowego automatu ze stosem, lecz posiadająca dwa stosy, potrafi rozpoznać te same języki co maszyna Turinga. Udowodnij, że pozostaje to prawdą jeśli dysponujemy tylko jednym symbolem stosowym (oprócz symbolu dna stosu). Maszynę taką nazywamy *maszyną z dwoma licznikami. Wskazówka: najpierw udowodnij, że wystarczą cztery liczniki, potem spróbuj zakodować ich stan przy pomocy jednego licznika, drugiego będziesz mógł/mogła użyć do manipulowania pierwszym.*

7 Nierozstrzygalność

Zadanie 67. Czy istnieje algorytm rozstrzygający, dla danej gramatyki bezkontekstowej G, czy istnieje słowo w, takie że $ww^Rw \in L(G)$?

Zadanie 68. Powtórz podany na wykładzie dowód nierozstrzygalności Problemu Odpowiedniości Posta.

Zadanie 69. Dla gramatyki bezkontekstowej G niech L_G oznacza generowany przez nią język. Skorzystaj z nierozstrzygalności problemu odpowiedniosci Posta aby pokazać, że zbiór tych par gramatyk G, H dla których zachodzi $L_G \cap L_H \neq \emptyset$ nie jest rekurencyjny. Czy jest on rekurencyjnie przeliczalny ?

Zadanie 70. (za 2 punkty)
Udowodnij, że nie istnieje algorytm rozstrzygający, dla danej gramatyki bezkontekstowej G i alfabetu A, czy $A^* = L(G)$.

Zadanie 71. Czy istnieje algorytm rozstrzygający, dla danych dwóch gramatyk bezkontekstowych G i H, czy $L(G) = L(H)$?

Zadanie 72. Jak każdy pamięta, deterministyczny automat ze stosem, to urządzenie zadane przez skończony zbiór instrukcji w formacie: *jeśli widzisz na taśmie wejściowej a, jesteś w stanie q, a z czubka stosu zdjąłeś b, to przejdź do stanu q', a na czubek stosu włóż słowo w.* Taki automat czyta słowo wejściowe literka po literce, zmieniając przy tym stan jak zwykły automat skończony, a do tego jeszcze buduje sobie stos. Czy istnieje algorytm odpowiadający, dla danych dwóch deterministycznych automatów ze stosem, czy istnieje niepuste słowo wejściowe, po przeczytaniu którego oba te automaty będą miały na swoich stosach takie same ciągi symboli?

Zadanie 73. Dla danych funkcji $f, g, h : \{0, 1, \ldots, p\} \to \{0, 1, \ldots, p\}$, i danego nieskończonego ciągu liczb naturalnych $(a_1, a_2, a_3 \ldots)$, niech $F_{f,g,h}(a_1, a_2, \ldots)$ będzie ciągiem liczb naturalnych którego i-ty element jest równy $f(a_{i-1}) +_p g(a_i) +_p h(a_{i+1})$, gdzie $+_p$ oznacza dodawanie modulo p (przyjmujemy , że $a_0 = 0$).

Udowodnij, że problem:

dane funkcje f, g i h, oraz skończone ciągi $(b_1, b_2, \ldots b_k)$ i $(c_1, c_2, \ldots c_l)$. Czy istnieje liczba iteracji n taka, że $F^n_{f,g,h}(b_1, b_2, \ldots b_k, 0, 0, 0 \ldots) = (c_1, c_2, \ldots c_l, 0, 0, 0 \ldots)$

jest nierozstrzygalny.

Zadanie 74. Powiemy, że semiproces Thuego Π, jest bezkontekstowy jeśli dla każdej pary $[w, v] \in \Pi$ słowo w składa się z jednej litery a słowo v jest niepuste. Czy problem słów dla bezkontekstowych semiprocesów Thuego jest rozstrzygalny?

Zadanie 75. Powiemy, że semiproces Thuego Π, jest prawie bezkontekstowy jeśli dla każdej pary $[w, v] \in \Pi$ jedno ze słów w i v składa się tylko z jednej litery, drugie zaś z dwóch liter. Czy problem słów dla prawie bezkontekstowych semiprocesów Thuego jest rozstrzygalny?
Uwaga. Użyta w tym i poprzednim zadaniu nomenklatura wymyślona została tylko by wygodniej było sformułować te zadania, i nie ma nic wspólnego z żadnym standardem

Zadanie 76. (za 2 punkty)
Udowodnij nierozstrzygalność problemu sprawdzenia dla danego procesu Thuego Π i słowa w czy zbiór $A_w = \{v : w \overset{*\Pi}{\leftrightarrow} v\}$ jest skończony.
Wskazówka: Rozważ maszyny Turinga z dodanym gdzieś na taśmie licznikiem, który jest zwiększany o jeden przy każdym ruchu wykonywanym przez maszynę. Naśladuj dowód nierozstrzygalności problemu słów.

Zadanie 77. (za 2 punkty) Rozpatrzmy skończony zbiór par słów P i binarną relację $\to$ na słowach zdefiniowaną jak następuje: $w \to_P v$ wtedy i tylko wtedy gdy istnieje para $< a, b > \in P$ taka że $w = ax$ i $v = xb$ gdzie x jest pewnym słowem. Niech $\overset{*}{\to}_P$ będzie przechodnim domknięciem $\to_P$ (to znaczy najmniejszą relacją przechodnią zawiarającą $\to_P$).

Czy problem: *dane* P, x, y, *czy* $x \overset{*}{\to}_P y$ *?* jest rozstrzygalny ?

Zadanie 78. Czy istnieje algorytm który, dla danej gramatyki bezkontekstowej G o alfabecie symboli terminalnych składającym się z zera, jedynki i dwójki, rozstrzyga czy w $L(G)$ istnieje palindrom mający tyle samo wystąpień symbolu 0 co symbolu 1?

Zadanie 79. (trudne, za 2 punkty)
Funkcję $f : \mathcal{N} \rightarrow \mathcal{N}$ nazywamy funkcją Conway'a jeśli istnieją liczby naturalne $p, a_1, b_1, a_2, b_2, \ldots a_p, b_p$ takie że dla każdego n jeśli $n = k \bmod p$ to $f(n) = na_k/b_k$. Pokaż, że nie ma algorytmu który dla danych $p, a_1, b_1, a_2, b_2, \ldots a_p, b_p$ odpowiadałby, czy dla definiowanej przez te współczynniki funkcji Conway'a istnieje m takie, że $f^m(2) = 1$, gdzie f^m oznacza funkcję f złożoną m razy ze sobą.

Zadanie 80. (za 2 punkty)
Udowodnij nierozstrzygalność następującego problemu: dany jest skończony zbiór kolorów C, zawierający co najmniej kolory: czerwony i biały, oraz zbiór $N \subseteq C^4$ czwórek kolorów, uznanych za *nieestetyczne*. Mamy nieskończenie wiele kwadratowych kafelków każdego koloru o boku długości 1. Czy istnieje kwadrat (o całkowitych wymiarach i boku nie mniejszym niż 2), który można wypełnić kafelkami w taki sposób, by w lewym dolnym i w lewym górnym narożniku znalazł się czerwony kafelek, pozostałe kafelki dolnego i górnego brzegu były białe, oraz by w całym kwadracie nie pojawiła się nieestetyczna sekwencja kafelków, tj. cztery sąsiadujące kafelki:

c_1	c_2
c_3	c_4

takie że $(c_1, c_2, c_3, c_4) \in N$.

Zadanie 81. Automat *niedeterministycznie ruszający dwiema nogami* zdefiniujemy sobie jako piątkę $\langle \Sigma, Q, q_o, F, \delta \rangle$, gdzie Σ jest skończonym alfabetem, Q skończonym zbiorem stanów, $q_0 \in Q$ jest stanem początkowym, $F \subseteq Q$ jest zbiorem stanów akceptujących a $\delta \subseteq Q \times \Sigma \times \Sigma \times Q \times \{0,1\} \times \{0,1\}$ jest relacją przejścia.

Relacja przejścia jest rozumiana następująco: jeśli $\delta(q, a_1, a_2, q', b_1, b_2)$, to gdy automat jest w stanie q, lewą nogę ma na symbolu a_1 a prawą nogę na symbolu a_2, to może przejśc do stanu q', przesunąć lewą nogę o b_1 symboli w prawo i przesunąć prawą nogę o b_2 symboli w prawo.

Automat rozpoznaje pary słów w_l, w_p, z których każde jest postaci $a(\Sigma \setminus \{a, z\})^* z$, dla pewnych ustalonych symboli $a, z \in \Sigma$

Na początku działania automat jest w stanie q_0 i ma lewą nogę na pierwszej literze (czyli a) słowa w_l, a prawą na pierwszej literze (czyli a) słowa w_p. Automat akceptuje parę słów, jeśli możliwe jest aby znalazł się obiema nogami na literach z i przeszedł wtedy w stan $q \in F$.

Czy problem totalności automatu niedeterministycznie ruszającego dwiema nogami jest rozstrzygalny? Przez problem totalności rozumiemy tu dopełnienie problemu istnienia jakiejkolwiek pary słów, o podanej powyżej postaci, ale nie akceptowanej przez dany automat.

8 Niedeterministyczne maszyny Turinga i klasa NP

Zadanie 82. Pokaż, że wielomianową maszynę niedeterministyczną można przerobić tak, aby zgadywała rozwiązanie wcześniej niż pozna dane. Dokładniej rzecz ujmując, udowodnij, że jeśli zbiór A należy do klasy $\mathcal{NP}$ to istnieją wielomiany p, q oraz niedeterministyczna maszyna Turinga M rozpoznająca A działająca, dla danego n w następujący sposób: M wyznacza na taśmie blok klatek o długości $p(|n|)$, (zatem interesuje ją wielkość n, ale nie jego dokładna wartość) po czym niedeterministycznie i nie czytając n zapełnia ten blok ciągiem zer i jedynek. Dopiero następnie czyta n i przechodzi do fazy w której obliczenie jest już deterministyczne i nie zabiera więcej niż $q(|n|)$ kroków.

Zadanie 83. Pokaż, że jeśli zbiór $A \subset \mathcal{N}^2$ jest w $\mathcal{P}$ i p jest wielomianem to zbiór $\{n : \exists m \ \ |m| \leq p(|n|) \ \ i \ \ [n,m] \in A\}$, czyli rodzaj rzutu A na pierwszą oś, jest w $\mathcal{NP}$.

Zadanie 84. Pokaż, że każdy zbiór w $\mathcal{NP}$ jest rzutem pewnego zbioru z $\mathcal{P}$, to znaczy jeśli B jest w $\mathcal{NP}$ to istnieje wielomian p i zbiór $A \subset \mathcal{N}^2$ należący do $\mathcal{P}$ i taki, że $B = \{n : \exists m \ \ |m| \leq p(|n|) \ \ i \ \ [n,m] \in A\}$.

9 Redukcje wielomianowe i NP-trudność

Zadanie 85. Pokaż że 5SAT$<_P$3SAT.

Zadanie 86. (za 2 punkty)
Problem STASI (ach, dziś już studenci nie wiedzą co to było STASI, niech się dowiedzą to zobaczą czemu tak nazwałem tu ten problem. Naprawdę nazywa się on problemem zbioru dominującego.) jest taki: mamy dany graf nieskierowany i liczbę k. Czy da się rozstawić k agentów w wierzchołkach grafu tak aby każdy wierzchołek w którym nie stoi agent miał (co najmniej jednego) agenta za sąsiada? Pokaż, że 3SAT $<_P$ STASI.

Wskazówka: To nie jest trudne. Idea jest podobna jak przy dowodzie faktu, że $3SAT <_P 3COL$, który był na wykładzie. Tylko łatwiej.

Zadanie 87. (za 2 punkty) Niech H oznacza problem cyklu Hamiltona dla grafów nieskierowanych (to znaczy język tych wszystkich grafów nieskierowanych w których istnieje ścieżka zamknięta przechodzaca dokładnie raz przez każdy wierzchołek).

Niech H_d oznacza problem cyklu Hamiltona dla grafów skierowanych. Pokaż, że $H <_P H_d$ i $H_d <_P H$. *Wskazówka: W trudniejszą stronę trzeba każdy wierzchołek odpowiednio zastapić trzema.*

Zadanie 88. (za 2 punkty) Klauzula nazywa się *hornowska* jeśli co najwyżej jeden z jej literałów jest niezanegowany. Pokaż, że problem HORNSAT spełnialności formuł w postaci CNF których każda klauzula jest hornowska jest w $\mathcal{P}$.

Zadanie 89. (za 2 punkty) Pokaż, że problem spełnialności formuł w koniunkcyjnej postaci normalnej w których każda klauzula jest alternatywą co najwyżej dwóch literałów jest w klasie $\mathcal{P}$. (Patrz definicja na stronie 375 polskiego wydania książki Hopcrofta i Ullmana. Tłumaczka z bożej łaski tłumaczy CNF jako PNK).

Zadanie 90. Pokaż, że 3SAT $<_P$ $3SAT_3$. To ostatnie to 3SAT ograniczony tylko do formuł w których żadna zmienna nie występuje więcej niż 3 razy.

Zadanie 91. (za 2 punkty) Udowodnij, że problem cyklu Hamiltona NP-zupełny.

Zadanie 92. Problem komiwojażera jest taki: dany jest graf nieskierowany pełny, którego krawędzie etykietowane są liczbami całkowitymi. Waga drogi w grafie jest definiowana jako suma wag krawedzi należących do tej drogi. Dana liczba k. Czy istnieje w grafie cykl Hamiltona o wadze mniejszej niż k?

Pokaż, że problem komiwojażera jest NP zupełny. Wolno skorzystać z NP-zupełności problemu Hamiltona.

Komentarz: Problem komiwojażera to jedyny kawałek teorii informatyki, który trafił do kultury masowej, stając się w ten sposób kolegą Myszki Miki.

Zadanie 93. (za 2 punkty) Pokaż, że jeśli istnieje wielomianowy algorytm wskazujący, dla danego przykładu problemu komiwojażera, cykl nie więcej niz dwa razy dłuższy od optymalnego, to $\mathcal{P}$=NP.

Wskazówka: To wcale nie jest trudne zadanie, z całą pewnością nie zasługuje na dwie gwiazdki. Podobnie jak w poprzednim trzeba sie odwołać do NP-zupełności problemu Hamiltona.

Zadanie 94. Pokaż, że jeśli ograniczymy się do przykładów problemu komiwojażera w którym wagi krawędzi spełniają nierówność trójkąta, to znaczy dla każdych wierzchołków v_1, v_2, v_3 zachodzi $d(v_1, v_2) + d(v_2, v_3) \geq d(v_1, v_3)$ to istnieje wielomianowy algorytm znajdujący cykl Hamiltona o wadze nie więcej niz dwa razy większej od optymalnej.

Zadanie 95. Jaka jest złozoność problemu 3SAT$_2$, tzn. problemu spełnialnosci formuł w postaci 3CNF w których żadna zmienna nie występuje więcej niż 2 razy?

Zadanie 96. Jaka jest złozoność problemu SAT$_2$, tzn. problemu spełnialnosci formuł w postaci CNF w których żadna zmienna nie występuje więcej niż 2 razy?

Zadanie 97. (za 2 punkty) Udowodnij, że problem istnienia w danym grafie o n wierzchołkach kliki mającej $n/2$ wierzchołków jest NP-zupełny.

Zadanie 98. Udowodnij, że jeśli istnieje wielomianowy algorytm znajdujący w danym grafie klikę wielkości co najmniej połowy kliki maksymalnej to istnieje również wielomianowy algorytm znajdujący w danym grafie klikę wielkości co najmniej $1/\sqrt{2}$ kliki maksymalnej.

Zadanie 99. Rozważmy następujący *problem spełnialności dla zdecydowanej większości klauzul*: dane różne klauzule rachunku zdań $\phi_1, \phi_2 \dots \phi_n$. Czy można podstawić wartości 0 i 1 za zmienne zdaniowe tak aby więcej niż 9/10 spośród klauzul $\phi_1, \phi_2 \dots \phi_n$ przyjeła wartość logiczną 1 ? Udowodnij, że *problem spełnialności dla zdecydowanej większości klauzul* jest NP-zupełny. Przypominam, że klauzulą nazywamy formułę postaci $l_1 \vee l_2 \vee \dots \vee l_k$, gdzie l_i są literałami, to znaczy zmiennymi zdaniowymi lub ich negacjami. *Wskazówka: skorzystaj z NP-zupełności SAT.*

Zadanie 100. Niech $\mathcal{P}_1$ będzie zbiorem grafów nieskierowanych niespójnych, niech $\mathcal{P}_2$ będzie zbiorem wszystkich grafów nieskierowanych zawierających jakiś wierzchołek stopnia mniejszego od 2 i niech wreszcie $\mathcal{P}_3$ będzie zbiorem tych grafów nieskierowanych, w których jest droga Hamiltona przechodząca przez przynajmniej połowę wierzchołków.

Czy zbiór $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2 \cup \mathcal{P}_3$ jest NP-zupełny?

Zadanie 101. Niech KLIKA$_c$ będzie problemem istnienia w danym grafie o n wierzchołkach kliki zawierającej nie mniej niż n/c wierzchołków. Pokaż że dla każdych c, c' zachodzi KLIKA$_c$ $\leq_P$KLIKA$_{c'}$.

Zadanie 102. Rozważmy nastepujący *problem smutnych strażników.* Dany jest pewien zbiór strażników $s_1, s_2, \dots s_l$. Strzegą oni obiektów $a_1, a_2, \dots a_k$. Odbywa się to tak, że każdy strażnik s_i ma w swoim kantorku dwa ekrany telewizyjne E_i i F_i, i na każdym z tych ekranów widzi, za pośrednictwem nieruchomej kamery, pewien niezmienny zbiór obiektów (odpowiednio Z_{E_i} i Z_{F_i}, zbiory te oczywiście niekoniecznie muszą być parami rozłączne). Powiemy że *strażników można rozweselić* jeśli da się przestawić u każdego z nich jeden telewizor na kanał gdzie akurat transmitują mecz, ale w taki sposób, że każdy obiekt ze zbioru $\{a_1, a_2, \dots a_k\}$ pozostaje pod obserwacją na jakimś nieprzestawionym telewizorze.

Pokaż że problem rozstrzygnięcia, dla danego przykładu problemu smutnych strażników, czy strażników tych da się rozweselić, jest NP-zupełny.

Zadanie 103. Udowodnij, że następujący *problem podziału n harcerzy na 4 drużyny* jest NP-zupełny.

Dany jest hufiec H harcerzy, i lista $E \subset H^2$ par harcerzy którzy się nie lubią. Czy da się podzielić H na cztery drużyny tak, aby spełnione były warunki:
– drużyny mają być z grubsza równej wielkości: do każdej z nich musi należeć przynajmniej jedna piąta wszystkich harcerzy z H.
– w żadnej drużynie nie mogą jednocześnie znaleźć się dwaj harcerze, którzy się nie lubią.

Zadanie 104. W Pewnej Wschodniej Krainie wszystkim rządzą trzej gangsterzy, pan R, pan G i pan B. Każda firma, która chce mieć spokój i dostawać koncesje i kontrakty, musi mieć wśród członków swojej rady nadzorczej przyjaciół przynajmniej dwóch spośród tych trzech gangsterów. Kłopot polega jednak na tym, że:

– gangsterzy za sobą nawzajem nie przepadają, więc każdy członek rady może przyjaźnić się z co najwyżej jednym gangsterem;

– rady nadzorcze różnych firm niekoniecznie muszą być rozłączne.

Udowodnij, że problem:
Dane listy członków rad nadzorczych pewnej ilości firm, czy da się osoby figurujące na tych listach pozaprzyjaźniać z panami R, G i B, w taki sposób, aby wszystkie z tych firm miały spokój?

Jest NP-zupełny.

Wskazówka: Zauważ, że warunki zadania implikują w szczególności, że rada nadzorcza firmy która chce mieć spokój, musi liczyć nie mniej niż dwóch członków. Okazuje się, że przy odpowiedniej redukcji wystarczy ograniczyć się do takich właśnie dwuosobowych rad.

Zadanie 105. Jaka jest złozoność następującego *problemu klasy udającej się na wycieczkę.* Klasa ma udać się na wycieczkę do Świeradowa. Jednak ze względu na trawiący ją wewnętrzny konflikt, niektórzy z młodych ludzi obwarowują kwestię swojego wyjazdu pewnymi warunkami. Warunki te mają następującą postać:

Ja (jadę|nie jadę) jeśli X (jedzie|nie jedzie)

gdzie X przebiega zbiór uczniów klasy. Każdy uczeń może przedstawić Pani Wychowawczyni dowolną ilość takich warunków. Jaka jest złożoność problemu sprawdzenia, czy da się zorganizować wycieczkę w sposób uwzględniający wszystkie postawione warunki? Wielkością zadania jest tu łączna ilość warunków.

Zadanie 106. Udowodnij że problem istnienia, dla danego grafu nieskierowanego, takiego kolorowania wierzchołków tego grafu trzema kolorami, aby każdy wierzchołek sąsiadował z co najwyżej jednym wierzchołkiem tego samego koloru, jest NP-zupełny.

Zadanie 107. Jaka jest złożoność problemu spełnialności formuł boolowskich w postaci CNF, jeśli ograniczymy sie do formuł w których:
a) w każdej klauzuli jest co najwyżej jedna niezanegowana zmienna,
b) w każdej klauzuli jest co najwyżej jedna niezanegowana zmienna albo co najwyżej jedna zanegowana zmienna.

Zadanie 108. Przykładem problemu pokrycia zbioru podzbiorami rozłącznymi (PZPR) jest skończona rodzina $A = \{A_1, A_2, \dots A_n\}$ skończonych zbiorów. $A \in$ PZPR jeśli istnieje rodzina $B \subseteq A$ zbiorów rozłącznych, taka że suma wszystkich zbiorów z B jest równa sumie wszystkich zbiorów z A. Udowodnij, że 3SAT$\leq_P$PZPR. *Wskazówka: pokaż, że* 3SAT3$\leq_P$PZPR *gdzie* 3SAT3 *to problem spełnialności dla formuł w postaci 3CNF w których każda znienna występuje najwyżej 3 razy.*

Zadanie 109. Jaka jest złożoność problemu istnienia, dla danej formuły boolowskiej w postaci CNF, wartościowania przy którym w każdej klauzuli wszystkie literały przyjmują wartość 1 albo wszystkie literały przyjmuja wartość 0? *Wskazówka: co wiesz o złożoności problemu 2CNF?.*

Zadanie 110. Niech $f : \{0,1\}^* \to \{0,1\}^*$ będzie bijekcją obliczalną w czasie wielomianowym. Czy wynika z tego, że f^{-1} też jest bijekcją obliczalną w czasie wielomianowym?

Zadanie 111. Jaka jest złożoność następującego problemu: Dany graf nieskierowany. Czy istnieje taki sposób pokolorowania jego krawędzi dwoma kolorami, czerwonym i niebieskim, aby każda krawędź byla pokolorowana i aby nie pojawił się żaden niebieski cykl nieparzystej długości ani żaden czerwony cykl nieparzystej długości?

10 Inne klasy złozoności obliczeniowej

Zadanie 112. Udowodnij, że problem, czy dane wyrażenie regularne opisuje wszystkie słowa nad danym alfabetem, jest w PSPACE.

Zadanie 113. (za 3 punkty) Instancja Gry w Kamienie to: skończony zbiór X (zbiór pól), relacja $R \subseteq X^3$, zbiór $Y \subset X$ i element $f \in X$.

Grę toczą dwaj gracze wykonujący na przemian ruchy. Przed kazdym ruchem na niektórych polach ze zbioru X znajdują się kamienie: przed pierwszym ruchem pierwszego gracza mamy po jednym kamieniu w każdym polu zbioru Y, który to zbiór wyznacza w ten sposób pozycje początkową w grze. W swoim ruchu kazdy z graczy przesuwa jeden kamień zgodnie z regułą, że jeśli zachodzi $R(x, y, z)$, oraz w x i w y sa kamienie a w z nie ma kamienia to wolno przesunąć kamień z x do z. Wygrywa ten kto pierwszy postawi kamień w f.

Jaka jest złożoność problemu: dana instancja gry w kamienie. Czy gracz pierwszy ma strategię wygrywającą?

Zadanie 114. Udowodnij, że problem prawdziwości formuł postaci $\exists! x_1 \exists! x_2 \dots \exists! x_n \;\; \phi(x_1, x_2 \dots x_n)$ jest w PSPACE. Zmienne $x_1, x_2 \dots . x_l$ przebiegają tu zbiór $\{0, 1, 2\}$. Kwantyfikator $\exists!$ oznacza *istnieje dokładnie jeden*, zaś ϕ jest formułą bez kwantyfikatorów, z n zmiennymi, zbudowaną przy pomocy spójników boolowskich i symboli arytmetyki modulo 3, to znaczy symboli dodawania i mnożenia modulo 3, symbolu równości i stałych $\{0, 1, 2\}$.

Zadanie 115. Jaka jest złożoność problemu istnienia, dla danej formuły boolowskiej ϕ w postaci 2CNF wartościowania spełniającego ϕ i przyporządkowującego wartość logiczną 1 przynajmniej trzem spośród zmiennych występujących w ϕ?

Zadanie 116. (za 2 punkty)
Udowodnij, że są języki rekurencyjne które nie są w PSPACE.

Zadanie 117. Udowodnij, że problem prawdziwości formuł postaci $Q_1 \; x_1 \; Q_2 \; x_2 \; \dots Q_n \; x_n \; \phi(x_1, x_2 \dots x_n)$, gdzie każdy Q_i to albo $\exists!$, oznaczający *istnieje dokładnie jeden*, albo $\exists$ (czyli *istnieje*), albo $\forall$ (czyli *dla każdego*), zaś ϕ jest formułą boolowską, jest PSPACE-zupełny.

Zadanie 118. (za 2 punkty)
Funkcja różnowartościowa $f : \mathcal{N} \to \mathcal{N}$, i taka, że dla każdego n zachodzi $|n| = |f(n)|$ jest *jednostronna* jeśli istnieje wielomianowy algorytm obliczający f ale nie ma wielomianowego algorytmu obliczającego f^{-1}. Pokaż, że jeśli istnieje jakaś funkcja jednostronna to $\mathcal{NP} \cap$ co-$\mathcal{NP} \neq \mathcal{P}$.

Wskazówka: Rozważ zbiór $\{< x, y >: f^{-1}(x) < y\}$

Definicja . Powiemy że język A należy do klasy altPTIME jeśli istnieją wielomian p i język $B \in$ PTIME, takie, że zachodzi równoważność:

$w \in A$ w.t.w. gdy gracz pierwszy ma strategię wygrywającą w opisanej poniżej grze $Gra(w, p, B)$

$Gra(w, p, B)$ ma następujące reguły. Zaczyna się od napisania na taśmie słowa $s_1 = w\#$. Następnie, w rundzie i-tej, najpierw gracz pierwszy dopisuje do aktualnie zapisanego słowa s_i wybrany przez siebie sufiks $w_i\#$ a następnie gracz drugi dopisuje do $s_i w_i\#$ pewien wybrany przez siebie sufiks $v_i\#$, tworząc w ten sposób słowo s_{i+1}. Żąda się przy tym aby długości w_i i v_i były obie równe $p(n)$, gdzie n jest długością słowa w. Gracz pierwszy wygrywa gdy $s_{p(n)} \in B$.

Definicja. Powiemy że język A należy do klasy altPTIME jeśli istnieją wielomian p i język $B \in$ PTIME, takie, że zachodzi równoważność:

$w \in A$ w.t.w. gdy gracz pierwszy ma strategię wygrywającą w opisanej poniżej grze $Gra(w, p, B)$

$Gra(w, p, B)$ ma następujące reguły. Zaczyna się od napisania na taśmie słowa $s_1 = w\#$. Następnie, w rundzie i-tej, najpierw gracz pierwszy dopisuje do aktualnie zapisanego słowa s_i wybrany przez siebie sufiks $w_i\#$ a następnie gracz drugi dopisuje do $s_i w_i\#$ pewien wybrany przez siebie sufiks $v_i\#$, tworząc w ten sposób słowo s_{i+1}. Żąda się przy tym aby długości w_i i v_i były obie równe $p(n)$, gdzie n jest długością słowa w. Gracz pierwszy wygrywa gdy $s_{p(n)} \in B$.

Zadanie 119. (za 2 punkty) Udowodnij że altPTIME=PSPACE.

Definicja . Powiemy że język A należy do klasy altPSPACE jeśli istnieją wielomian p oraz języki $B, C \in$ PTIME, takie, że zachodzi równoważność:

$w \in A$ w.t.w. gdy gracz pierwszy ma strategię wygrywającą w opisanej poniżej grze $Gra2(w, p, B, C)$

$Gra2(w, p, B)$ ma następujące reguły. Zaczyna się od napisania na taśmie słowa $w\#$. Następnie, w pierwszej rundzie, najpierw gracz pierwszy dopisuje do $w\#$ wybrany przez siebie sufiks $w_1\#$ tworząc w ten sposób $t_1 = w\#w_1\#$, a następnie gracz drugi dopisuje do t_1 pewien wybrany przez siebie sufiks $v_1\#$, tworząc w ten sposób słowo s_1. W i-tej rundzie najpierw gracz pierwszy wymazuje z taśmy słowo w_{i-1} zastępując je wybranym przez siebie w_i (powstałe w ten sposób słowo nazywamy t_i) a następnie drugi gracz wymazuje z taśmy słowo v_{i-1} zastępując je przez v_i. Powstałe słowo (równe $w\#w_i\#v_i$) oznaczamy przez s_i. Żąda się przy tym aby długości w_i i v_i były obie równe $p(n)$, gdzie n jest długością słowa w. Gra kończy się porażką pierwszego gracza, gdy w którejś rundzie pojawi się słowo t_i takie że $t_i \notin B$. Gra kończy się porażką drugiego gracza, gdy w którejś rundzie pojawi się słowo s_i takie że $s_i \notin C$.

Zadanie 120. Udowodnij, że jeśli któryś z graczy ma strategię wygrywajacą w grze $Gra2(w, p, B, C)$, to może doprowadzić do zwycięstwa nie dalej, niż po wykładniczej względem długości w ilości rund.

Zadanie 121. (za 2 punkty) Udowodnij, że $altPSPACE \subseteq EXPTIME$.

Zadanie 122. za 3 punkty Udowodnij, że $EXPTIME \subseteq altPSPACE$.

Zadanie 123. (za 2 punkty) Wyobraźmy sobie prostokątną tabelkę o 17 wierszach i 5 kolumnach, w każde pole której wolno wpisać 0 lub 1. Ponadto wyobraźmy sobie formułę zdaniową KROK, w ktorej występuje 10 zmiennych.

Mówimy, że tabelka jest poprawnie wypełniona jeśli dla każdych kolejnych dwóch wierszy $a_1a_2a_3a_4a_5$ i $b_1b_2b_3b_4b_5b_6$ zachodzi:

$$\mathrm{KROK}(a_1, a_2, a_3, a_4, a_5, b_1, b_2, b_3, b_4, b_5).$$

Niech $k_1k_2k_3k_4k_5$ i $m_1m_2m_3m_4m_5$ będą dwoma ustalonymi ciągami zerojedynkowymi. Napisz QBF mówiącą, że można poprawnie wypełnić naszą tabelkę zerami i jedynkami w taki sposób, że w pierwszym wierszu jest $k_1k_2k_3k_4k_5$ w ostatnim zaś $m_1m_2m_3m_4m_5$. Formuła nie może zawierać więcej niż 45 zmiennych (nie liczymy $m_1 \dots m_5$ i $k_1 \dots k_5$ - to nie są zmienne). Oczywiście szukana QBF będzie zawierała pewną ilość wystąpień podformuły KROK.

Przez QBF rozumiemy tu formułę zdaniową o k zmiennych poprzedzoną k kwantyfikatorami, wiążącymi te zmienne. To znaczy $\forall p\ \exists q\ (p \vee q) \wedge (\neg p \vee q)$ jest QBF a $\forall p\ [(\exists q\ (p \vee q)) \wedge (\exists q(\neg p \vee q))]$ nie jest.

Zadanie 124. Napisz formułę rachunku predykatów mówiącą, że w danym grafie $\langle V, R\rangle$ istnieje ścieżka prowadząca z danego wierzchołka c do danego wierzchołka k złożona z dokładnie 16 krawędzi. Formuła ta ma mieć przy tym nie więcej niż 10 wystąpień kwantyfikatorów.

Przez formułę rachunku predykatów rozumiemy tu formułę w której używa się kwantyfikatorów wiażących zmienne przebiegające zbiór V, symbolu relacji R, symbolu równości, nawiasów i spójników logicznych.

Wyjaśnienie. Formuła: $\exists x_1\ \exists x_2 \dots \exists x_{15}\ \ R(c, x_1) \wedge R(x_1, x_2) \wedge \dots R(x_{15}, k)$
spełnia wszelkie wymogi zadania, oprócz tego, że ma 15 kwantyfikatorów.

Zadanie 125. Dla danej formuły ϕ zbudowanej, zgodnie ze zwykłymi regułami, ze zmiennych $x_1, x_2, \dots x_n$, $y_1, y_2, \dots y_n$, z liczb naturalnych 0,1 i 2, ze spójników boolowskich oraz z symboli $=_3$, $+_3$ i $\times_3$ (rozumianych jako równość modulo 3, dodawanie modulo 3 i mnożenie modulo 3), ale bez kwantyfikatorów, rozważmy następującą grę $G(\phi)$ między graczami X i Y.

Rozgrywka składa się z n ruchów, w trakcie których zastępuje się zmienne w formule ϕ stałymi. W ruchu i najpierw gracz X wybiera $k \in \{0, 1, 2\}$ i zastępuje wszystkie wystąpienia zmiennej x_1 przez liczbę k, a następnie gracz Y wybiera $l \in \{0, 1, 2\}$ i zastępuje wszystkie wystąpienia zmiennej y_1 przez liczbę l. Gracz X wygrywa jeśli formuła bez zmiennych, w jaką zamieni się ϕ po ostatnim ruchu gracza Y, jest prawdziwa.

Niech 3GRA będzie problemem rozstrzygnięcia, dla danej formuły ϕ, czy gracz X ma strategię wygrywającą w grze $G(\phi)$. Udowodnij, że QBF$\leq_P$3GRA.

Zadanie 126. Jaka jest złożoność problemu prawdziwości kwantyfikowanych formuł boolowskich, w których są co najwyżej dwa wystąpienia kwantyfikatora ogólnego?

Zadanie 127. Jaka jest złożoność problemu rozstrzygnięcia, dla danych dwóch deterministycznych automatów skończonych M_1 i M_2 czy język $L_{M_1} \cap L_{M_2}$ jest niepusty? (wielkością zadania jest tu łączna ilość stanów tych automatów)

Zadanie 128. Jaka jest złożoność problemu rozstrzygnięcia, dla danych deterministycznych automatów skończonych $M_1, M_2 \dots M_k$ czy język $L_{M_1} \cap L_{M_2} \dots \cap L_{M_k}$ jest niepusty? (wielkością zadania jest tu łączna ilość stanów tych automatów)