Zadanie 1. (6 punktów) Słowo w nazwiemy anagramem słowa v jeśli w można otrzymać z v poprzez zamianę kolejności liter. Niech

$$anagram(L) = \{w : w \text{ jest anagaramem } v \text{ dla pewnego } v \in L\}.$$

(a) Czy jeśli L jest regularny, to $anagram(L)$ też?

(b) Czy jeśli L jest bezkontekstowy, to $anagram(L)$ też?

Rozwiązanie. Rozważmy regularny język $L = (abc)^*$. Nietrudno zauważyć, że $anagram(L)$ to zbiór słów, gdzie wszystkie litery a, b, c występują tyle samo razy. Pokażemy, że język $anagram(L)$ nie jest nawet bezkontekstowy, a więc odpowiedź na oba podpunkty brzmi „nie". Gdyby język $anagram(L)$ był bezkontekstowy, to bezkontekstowy byłby język

$$K = anagram(L) \cap a^*b^*c^* = \{a^n b^n c^n : n \in \mathbb{N}\},$$

jako iloczyn języka bezkontekstowego i regularnego. Dowód, że K nie jest bezkontekstowy był na wykładzie.

Zadanie 2. (6 punktów) Czy jeśli L jest bezkontekstowy, to $\{w : ww^R \in L\}$ też?
Rozwiązanie. Nie. Rozważmy język

$$L = \{a^n b^n c^{2m} b^m a^k : n, m, k \in \mathbb{N}\}.$$

Język L jest bezkontekstowy. Generuje go poniższa gramatyka.

$$\begin{array}{rcl} S & \to & XYZ \\ X & \to & aXb|\epsilon \\ Y & \to & ccYb|\epsilon \\ Z & \to & aZ|\epsilon \end{array}$$

Nietrudno zauważyć, że

$$\{w : ww^R \in L\} = \{a^n b^n c^n : n \in \mathbb{N}\},$$

a więc język ten nie jest bezkontekstowy.

Zadanie 3. (6 punktów) Załóżmy, że rozpatrujemy maszyny Turinga akceptujące słowa nad alfabetem $\{a, b\}$. Dla każdego z poniższych problemów ustalić, czy jest rozstrzygalny i czy jest częściowo rozstrzygalny.

(a) Dany jest kod maszyny Turinga. Pytanie: czy maszyna akceptuje wyłącznie słowa składające się z samych liter a?

(b) Dany jest kod maszyny Turinga. Pytanie: czy maszyna akceptuje wyłącznie słowa zawierające literę b?

(c) Dany jest kod maszyny Turinga. Pytanie: czy maszyna akceptuje pewne słowo zawierające literę b?

Rozwiązanie. Będziemy korzystać w redukcjach z dopełnienia problemu stopu:

> (NS) Dany jest kod maszyny Turinga. Pytanie: czy maszyna *nie* akceptuje słowa pustego?

Problem (NS) nie jest nawet częściowo rozstrzygalny. (Gdyby był, to problem stopu by był rozstrzygalny.)

Pokażemy, najpierw, że (a) nie jest nawet częściowo rozstrzygalny. Dla kodu maszyny Turinga M nietrudno wyprodukować kod maszyny M' taki, że

- Jeśli M akceptuje słowo puste, to $L(M') = b^+$.
- Jeśli M nie akceptuje słowa pustego, to $L(M') = \emptyset$.

Funkcja f, która zamienia kod maszyny M na kod maszyny M' jest obliczalna. Wystarczy do kodu maszyny M dopisać wstęp, który sprawdza, czy dane na wejściu słowo należy do b^+. Jeśli tak, wycieramy wejście (zostawiając słowo puste) i uruchamiamy M. Jeśli nie, odrzucamy.

Nietrudno zobaczyć, że funkcja f jest redukcją problemu (NS) do (a), a więc (a) nie może być nawet częściowo rozstrzygalny.

Teraz pokażemy, że (b) też nie jest częściowo rozstrzygalny. Rozważmy funkcję obliczalną g, która działa podobnie jak maszyna f, z tą różnicą, że zamiast b^+ jest a^+. Funkcja g jest redukcją problemu (NS) do (a).

Problem (c) jest częściowo rozstrzygalny. Algorytm zgaduje niedeterministycznie słowo zawierające literę b, a następnie uruchamia maszynę na tym słowie. Problem (c) nie jest rozstrzygalny, ponieważ jest dopełnieniem problemu (a), który nie jest rozstrzygalny.

Zadanie 4. (6 punktów *) Rozważmy automaty ze stosem z następującym warunkiem akceptacji: „stos należy do L”, gdzie L jest bezkontekstowy. Czy istnieje język, który nie jest bezkontekstowy, a jest rozpoznawany przez rozważany rodzaj automatu?

Rozwiązanie. Dla zwiększenia obrazowości rozwiązania i zmniejszenia liczby oznaczeń poszczególne alfabety występujące w rozwiązaniu będziemy oznaczać różnymi kolorami.

Niech A - automat rozpoznający L z niebieskim stosem, z jednym stanem, stos początkowy „niebieska pinezka”, operacje „*read(czerwona litera lub epsilon); pop(niebieska litera); push(niebieskie słowo)*”, akceptacja przez pusty niebieski stos (taki właśnie automat powstaje po konwersji z gramatyki bezkontekstowej).

Niech B - automat rozpoznający nasz główny język, stos początkowy pusty, operacje „*read(biała litera)*”, „*push(zielona litera)*”, „*pop(zielona litera)*”, akceptuje, jeśli słowo będące na stosie, przekolorowane na czerwono, jest w języku L.

Teraz zmieniamy B w następujący sposób:

Zamiast robić „*push(zielona litera)*”, myślimy o tym, czy automat B tą literę będzie chciał kiedykolwiek zdjąć – (tzn. niedeterministycznie zgadujemy) – jeśli nie, to od razu tą literę przekolorowujemy na czerwono. Zatem stos wygląda tak, że najpierw ma czerwone litery (które tam zostaną już do końca, dla automatu A), a potem zielone (tymczasowe pomocnicze dla B), nigdy nie ma zielonej przed czerwoną, do zaakceptowania cały stos musi być czerwony, a żeby literę zdjąć, musi ona być zielona.

Teraz zamiast wrzucać te czerwone litery, przekazujemy je bezpośrednio do A, czyli nasz stos będzie teraz wyglądał tak, że na początku są niebieskie litery pomocnicze dla A, a potem zielone pomocnicze dla B. Czyli wrzucamy na początku niebieską pinezkę i zamiast wrzucać czerwone litery na stos, traktujemy je jako wejście dla automatu A (czyli patrzymy, co zrobiłby automat A, jakby wczytał taką literę i wykonujemy odpowiednią operację na niebieskim stosie (możemy to zrobic, bo w takim momencie nie ma zielonych na wierzchu); potem ewentualnie wykonujemy jeszcze ϵ-przejścia automatu A). Akceptujemy przez pusty stos.

(9 punktów) Wybierz 9 z poniższych pytań i odpowiedz na nie. Za prawidłowe odpowiedzi dajemy +1 punkt, za złe -1 punkt. Punkty policzymy za 9 najgorszych odpowiedzi, czyli nie opłaca się odpowiadać na więcej niż 9 pytań.

W każdym pytaniu podana jest modyfikacja znanego modelu. Trzeba powiedzieć, czy model zmodyfikowany ma tę samą siłę wyrazu co niezmodyfikowany.

1. Maszyna Turinga, która zawsze przesuwa głowicę w prawo.
   Odpowiedź: Model nie jest równoważny. 83 poprawnych na 86 prób.
   Krótkie uzasadnienie: Taka maszyna jest równoważna automatowi skończonemu. Pisanie na taśmie niczego nie wnosi, podobnie wyjście poza ostatnią literę słowa.

2. Maszyna Turinga, która ma prawo pisać tylko na słowie wejściowym, a nie na symbolach "blank".
   Odpowiedź: Model nie jest równoważny. 66 poprawnych na 69 prób.
   Krótkie uzasadnienie: Dla każdej takiej maszyny problem "czy maszyna akceptuje dane słowo w" jest rozstrzygalny (a nie jest rozstrzygalny dla niektórych maszyn Turinga), bo ilość konfiguracji użytych w biegu jest ograniczona przez długość słowa wejściowego w.

3. Maszyna Turinga (niedeterministyczna), która może przesuwać głowicę tylko o 1 komórkę w prawo, lub na początek taśmy.
   Odpowiedź: Model jest równoważny. 46 poprawnych na 49 prób.
   Krótkie uzasadnienie: Dowolną maszynę symulujemy M maszyną N ograniczonego rodzaju w następujący sposób. Załóżmy, że bieg maszyny M składał się z konfiguracji $c_1, c_2, \ldots, c_n$. Symulując jeden krok maszyny M, maszyna N zostawia na taśmie zapis konfiguracji c_i, wraz z położeniem głowicy. Aby przejść z zapisu c_i do c_{i+1}, należy przejechać całe słowo, przesunąć znacznik głowicy i wrócić na początek. Przy przesuwaniu znacznika w lewo przydaje się niedeterminizm, choć można to nawet zrobić deterministycznie.

4. Automat ze stosem, który może przesuwać się po wejściu w dwie strony. (Przykładowa tranzycja: w stanie q, z a na wejściu i b na czubku stosu, zmień stan na p, zdejmij ze stosu literę b i przesuń głowicę w lewo.)
   Odpowiedź: Model nie jest równoważny. 51 poprawnych na 53 prób.
   Krótkie uzasadnienie: Taki automat potrafi rozpoznać język $\{a^n b^n c^n : n \in \mathbb{N}\}$, który nie jest bezkontekstowy. Najpierw, w zwykły sposób, maszyn sprawdza, czy jest tyle samo a co b. Potem wraca do pierwszego b i sprawdza, czy jest tyle samo b co c. Ogólniej, problem niepustości jest nierozstrzygalny dla takich maszyn.

5. Automat ze stosem, gdzie operacja "zdejmij ze stosu literę b" jest uogólniona do "zdejmij ze stosu słowo z języka L", dla L języka regularnego.
   Odpowiedź: Model jest równoważny. 28 poprawnych na 30 prób.
   Krótkie uzasadnienie: Taką operację symulujemy poprzez zjadanie ze stosu pamiętając stan automatu skończonego dla L.

6. Automat ze stosem, który może wykonać przejście postaci “jeśli zawartość stosu należy do L, to zmień stan z p na q”, dla L języka bezkontekstowego.
Odpowiedź: Model nie jest równoważny. 36 poprawnych na 37 prób.
Krótkie uzasadnienie: Taki automat potrafi rozpoznać język $\{a^n b^n c^n : n \in \mathbb{N}\}$, który nie jest bezkontekstowy. Przy czytaniu prefiksu $a^* b^*$, wrzucamy go na stos. Pomiędzy ostatnim b a pierszym c, sprawdzamy czy stos ma tyle samo a co b. Potem czytamy litery c, zdejmując dla każdej c jedno b ze stosu. Na końcu mają być na stosie same a.

7. Automat ze stosem, z operacją “podwój stos”, która zamienia stos w na stos ww.
Odpowiedź: Model nie jest równoważny. 67 poprawnych na 72 prób.
Krótkie uzasadnienie: Taki automat potrafi rozpoznać język $\{a^n b^n c^n : n \in \mathbb{N}\}$, który nie jest bezkontekstowy. Przy czytaniu prefiksu a^*, wrzucamy go na stos. Przed przeczytanim pierwszego b, wrzucamy na stos $\#$, a potem stos podwajamy. W ten sposób liczbę n mamy dwa razy na stosie, co pozwala nam sprawdzić długość bloku liter b oraz długość bloku liter c.

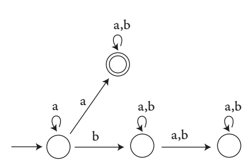
8. Gramatyka bezkontekstowa, gdzie prawe strony reguł mogą zawierać wyrażenia regularne, np. $X \to (X^* a Y)^*$.
Odpowiedź: Model jest równoważny. 75 poprawnych na 77 prób.
Krótkie uzasadnienie: Na przykład regułę $X \to (X^* a Y)^*$ symulujemy przez dodanie nowych nieterminali A, B oraz reguł

$$X \to A \qquad A \to AA|BaY \qquad B \to BB|X.$$

9. Wyrażenia regularne, gdzie dozwolona jest operacja różnicy $L - K$.
Odpowiedź: Model jest równoważny. 53 poprawnych na 59 prób.
Krótkie uzasadnienie: Język regularne są zamknięte na dopełnienie.

10. Wyrażenia regularne, gdzie każda litera alfabetu występuje $\leqslant 10$ razy.
Odpowiedź: Model nie jest równoważny. 81 poprawnych na 84 prób.
Krótkie uzasadnienie: Dla ustalonego alfabetu, np. alfabetu $\{a\}$, takich wyrażeń jest skończenie wiele (nierównoważnych), a języków regularnych jest nieskończenie wiele. Inny argument: nie można opisać języka $\{a^1 1\}$.

11. Automat skończony, gdzie warunek akceptacji to: “co najmniej połowa biegów jest akceptująca”.
Odpowiedź: Model nie jest równoważny. 13 poprawnych na 24 prób.
Krótkie uzasadnienie: Rozważmy następujący automat.



Na słowie $a^n b^m$ ma on n biegów akceptujących (ukośną tranzycją można wczytać dowolną literę a) oraz $m-1$ biegów nieakceptujących (prawą tranzycją poziomą można wczytać dowolną literę b oprócz pierwszej). Słowo takie zostanie zaakceptowane wtedy i tylko wtedy gdy $n \geqslant m-1$. Skądinąd wiadomo (nierywialny dowód), że nierozstrzygalny jest problem: czy dla danego automatu $\mathcal{A}$ istnieje słowo wejściowe gdzie automat ma co najmniej połowę biegów akceptujących.

12. Automat skończony, który może wykonać instrukcję "wróć do początku słowa w stanie q".
Odpowiedź: Model jest równoważny. 21 poprawnych na 32 prób.
Krótkie uzasadnienie: Aby symulować taki automat, wystarczy śledzić jego bieg od początku w każdym z możliwych stanów.

13. Automat skończony, który może wykonać instrukcję "usuń ostatnią literę czytanego słowa". Automat nie dowiaduje się jaka litera zniknęła. Na przykład, jeśli wykona tę instrukcję czytając pierwszą literę słowa *abbac*, to zniknie *c*.
Odpowiedź: Model nie jest równoważny. 33 poprawnych na 51 prób.
Krótkie uzasadnienie: Takim automatem można rozpoznać nieregularny język $\{a^n b(a+b)^n : n \in \mathbb{N}\}$. Czytając litery a na początku, usuwamy ostatnie litery. Akceptujemy jeśli po pierwszej literze b jest koniec wejścia.

14. Automat ze stosem, który może wykonać instrukcję "usuń ostatnią literę czytanego słowa".
Odpowiedź: Model nie jest równoważny. 5 poprawnych na 17 prób.
Krótkie uzasadnienie: Taki automat potrafi rozpoznać język $\{a^n b^n a(a+b)^n : n \in \mathbb{N}\}$, który nie jest bezkontekstowy. Najpierw, w zwykły sposób, maszyn sprawdza, czy jest tyle samo a co b, ale przy czytaniu liter b automat usuwa ostatnią literę słowa wejściowego. Aby zaakeptować, maszyna musi skończyć po wczytaniu pierwszego a po bloku liter b.

15. Maszyna Turinga, która się blokuje (odrzucając wejście) jak tylko drugi raz odwiedzi tę samą konfigurację.
Odpowiedź: Model jest równoważny. 32 poprawnych na 35 prób.
Krótkie uzasadnienie: Każdą maszynę Turinga można zmodyfikować tak, by nie odwiedzała dwa razy tej samej konfiguracji. Maszyna na specjalnej taśmie (można i bez dodawania taśmy) trzyma licznik, który zwiększa po wykonaniu każdego kroku. Inne rozwiązanie: maszyny nie trzeba modyfikować, bo biegi odwiedzające dwa razy tę samą konfigurację nic nie wnoszą do akceptacji.