# 1

Zakładamy że istnieje DFA A rozpoznający język L, który ma mniej niż 1024 stany.
Bierzemy dwa dowolne różne słowa w1, w2 z alfabetu {0, 1} dł. 10 takie, że po ich przejściu A kończy w tym samym stanie q1. (Słów jest 2^10, a stanów mniej, zasada szufladkowa). Słowa muszą się różnić na jakiejś pozycji i. Następnie dopisujemy do każdego z tych słów (10 - i) zer. Po przejściu wydłużonej tych słów A skończy w stanie q2 (akceptującym bądź nie), takim samym dla w1 i w2. Sprzeczność, gdyż w1 należy do L, a w2 nie (BSO).

# 2

Niech L – język z zadania.
Bierzemy zbiór A3 = { w ∈ {a, b, c}* | |w| ≤ 3 }. |A3| = 16, pokażemy, że tyle stanów musi mieć DFA A rozpoznający L. Załóżmy, że ma mniej, bierzemy w1, w2 ∈ A3 takie że po ich przejściu A kończy w tym samym stanie q1. Słowa muszą się różnić na jakiejś pozycji i ≤ 2. Rozważamy przez przypadki, że zawsze możemy dopisać taki ciąg za w1 i w2, że w1 będzie należeć do L, a w2 nie.

# 3

Udowodnij, że język L = { a^n b^2n | n ∈ N } nie jest regularny.

Z lematu o pompowaniu.
Wikipedia, całkiem niezła:
https://www.wikiwand.com/en/Pumping_lemma_for_regular_languages

Załóżmy, że L jest regularny. Wtedy istnieje takie p ≥ 1, że każde słowo w ∈ L takie że |w| ≥ p może być podzielone na w = xyz, tak że:

- |y| ≥ 1
- |xy| ≤ p
- Dla każdego n ≥ 0 (x y^n z) należy do L.

Intuicja: p - liczba stanów DFA rozpoznającego L.

W tym konkretnym zadaniu:

Załóżmy, że L jest regularny. Wtedy istnieje takie p, weźmy to p. Słowo w = a^p b^2p należy do L. W takim razie istnieje podział, …, niech xyz = w. Wiemy, że |y| > 1 i |xy| ≤ p, w takim razie x = a^|x|, y = a^|y|. Wtedy dla każdego n ≥ 0 (a^|x| a^n|y| a^|2p - |z|| b^2p) należy do L, co daje sprzeczność dla n ≠ 1.

# 4

Źródło:
https://cs.stackexchange.com/questions/10013/if-l-is-a-subset-of-0-then-how-can-we-show-that-l-is-regular

# 35

Rel: https://www.wikiwand.com/en/Synchronizing_word

a. Tak, dla A, dla sync(S) konstruujemy DFA który go rozpoznaje, gdzie stany to zbiory stanów z Q (P(Q)), f. przejścia z A tylko po zbiorach, stan początkowy – zbiór wszystkich stanów S, stany akceptujące – każdy singleton z P(Q).
b. Nie, bo w vw możemy “wypaść” poza S i wtedy to, że w ∈ sync(S) nic nam nie da. Samo wv’ raczej tak. Przydałby się jakiś kontrprzykład.
c. Tak, to co było wcześniej nie ma znaczenia, a później idziemy deterministycznie.

# 36

a. Rel: http://www.math.uni.wroc.pl/~kisiel/auto/volkov-surv.pdf Jeżeli dla danego 2-elementowego S sync(S) niepusty, to istnieje w nim w t. że |w| < |Q|^2. Konstruujemy pomocniczy automat, którego stany to 1- lub 2-elementowe zbiory stanów z A. Jest ich |Q|(|Q| + 1)/2. Stan startowy to S, stany akceptujące to singletony. Z założenia wiemy, że istnieje jakaś ścieżka między stanem startowym a jednym z akceptujących, więc musi być jakaś o długości nie większej niż |Q|^2.
b. Rel:
https://pdfs.semanticscholar.org/7d28/e902a741176f75db63731e74373d9d33e159.pdf
Jeżeli sync(Q) jest niepusty, to istnieje w nim w takie że |w| < |Q|^3. Pokażemy, jak skonstruować takie w. Intuicja – bierzemy dwa stany z Q, i idziemy z nich do jednego ścieżką v, gdzie |v| < |Q|^2, potem z |Q| - 1 takich ścieżek sklejamy w.
    - Q’ := Q
    - w := ε
    - dopóki |Q’| > 1:
        - weź dowolne q1, q2 z Q’
        - znajdź takie v, że v ∈ sync({q1, q2}) ∧ |v| < |Q|^2 (z a.)
        - w := wv
        - Q’ := Q’ ∪ d(q1, v) // albo q2, wszystko jedno

# 37

To może być wprost automat z papera Cernego (rel 36 a.). Ale pewnie można lepiej, bo u nas S ma być tylko jeden dla danego A.

# 38

Chyba tak, argument analogiczny jak w 36 b.

# 39

a. Analogicznie jak 36 a.?
b. Zbiór potęgowy stanów A, pokazujemy że istnieje ścieżka ze zbioru wszystkich stanów do singletonu, w takim razie musi być taka krótsza niż 2^|Q|