

1.

$$\frac{f(n)}{g(n)} = \frac{n^{\frac{1}{100}} 2^{100}}{2^{\frac{1}{100}n} n^{100}} = 2^{100} \frac{1}{2^{\frac{1}{100}n} n^{9999/100}} \rightarrow 0$$

$$\frac{f(n)}{h(n)} = \frac{n^{\frac{1}{100}} 2^{100}}{100n^{\frac{1}{100}n}} = \frac{2^{100}}{100} \frac{1}{n^{\frac{1}{100}n - \frac{1}{100}}} \rightarrow 0$$

Ez alapján $f(n) \neq \Omega(g(n))$, $g(n) = \Omega(f(n))$, $f(n) = O(h(n))$, $h(n) \neq O(f(n))$.

2. Legyen A a 2 egység sugarú körvonal pontjaiból álló halmaz. B_1, B_2 pedig a két diszjunkt egységsugarú körvonal pontjaiból álló halmaz. $B = B_1 \cup B_2$ az uniójuk. Megadunk egy bijekciót A és B között, ezzel bizonyítva, hogy a két halmaz számossága azonos (valójában mindkettő continuum számosságú). A körvonal egy pontját egyértelműen meghatározza a középpontból a pontba mutató vektornak a vízszintessel bezárt φ szöge, ahol $0^\circ \leq \varphi < 360^\circ$. Legyen a megfeleltetés a következő: egy φ szögű $x \in A$ pontnak feleljen meg a 2φ szögű $f(x)$ pont B_1 -ben ha $0^\circ \leq \varphi < 180^\circ$ illetve a $2\varphi - 360^\circ$ szögű pont B_2 -ben ha $180^\circ \leq \varphi < 360^\circ$. Ez az f megfeleltetés könnyen láthatóan egy bijekció A és B között.

Megjegyzés: A körök sugara lényegtelen a feladatban.

3. (a) Pl. 0010 szóra: $q_0 0100 \vdash \bar{0} q_1 100 \vdash^3 \bar{0} 100 q_1 \sqcup \vdash \bar{0} 10 q_2 0 \vdash \bar{0} 1 q_3 0 \bar{0} \vdash^2 q_3 \bar{0} 10 \bar{0} \vdash 0 q_0 10 \bar{0} \vdash \bar{0} \bar{1} q_5 0 \bar{0} \vdash \bar{0} \bar{1} q_6 0 \bar{0} \vdash 0 q_7 \bar{1} \bar{1} 0 \vdash 00 q_0 \bar{1} 0 \vdash 00 q_i 10$.

- (b) Az \mathcal{M} TG megfordítja a szavakat. (Tehát $f: u \mapsto u^{-1}$.) Legyen $u \in \{0, 1\}^*$ egy input szó. Egy ciklus előtti szalag tartalom $z^{-1}yx^{-1}$, ahol $u = xyz$, $x, y, z \in \{0, 1\}^*$ és $\ell(x) = \ell(z)$, x^{-1} első betűje meg van jelölve, a TG q_0 állapotban van és író-olvasó fej y első betűjén áll. Egy ciklusban a TG megjelöli y első betűjét és megjegyzi egy állapotában, megkeresi x^{-1} első betűjét, leveszi róla a jelölést és 1-et visszalép. Az ott olvasott betűt felülírja a megjegyzett betűvel és megjelöli, a kitörölt betűt állapotában megjegyzi, majd visszamegy y első betűjére, kitörli és felülírja a megjegyzett betűvel, majd jobbra lép. Tehát az történt, hogy kicseréltük y első és utolsó betűjét, a jelölés eggyel balra csúszott (x^{-1} első betűjéről y utolsó, azaz a csere után már az eredetileg első, betűjére), az író-olvasó fej új pozíciója pedig immár y második betűje. Tehát ha a módosított y első és utolsó betűjét most már z^{-1} illetve x^{-1} részének tekintjük, akkor a ciklus végeztével az a helyzet, mint az elején, de most már egy 2-vel rövidebb y -nal. A végén ha páros hosszú volt a szó q_0 -ban, ha páratlan hosszú akkor q_2 -ben vagy q_6 -ban eltüntetjük az egyetlen megjelölt betű jelölését és megállunk. Nem kell ilyen részletesen a 4 pontért.

- (c) Egy ciklusban a lépések száma $2\ell(y) + 1$. Tehát a TG időigénye egy n hosszúságú inputon: $h(n) = 2n + 1 + 2(n-2) + 1 + 2(n-4) + 1 \dots h(n) = (n+2)(n+1)/2(-1) = O(n^2)$ (a -1 akkor, ha n páros). Tehát pl. $k = 2$ jó választás.

4. (a) Egy ötlet 2 szalagosra: Végigmegyünk a szón miközben minden a -t átmásolunk a 2. szalagra, b esetén csak az első szalagon lépünk, majd visszafele haladva minden b -re az első szalagon, a -ra a másodikon L, L -et, minden a -ra az első szalagon L, S -t lépünk. Ha a végén egyszerre olvasunk a két szalagon \sqcup -t, akkor elfogadjuk a szót. Átmenetdiagram talán később...

- (b) A szalagon kétszer megyünk végig, tehát a TG időigénye $\Theta(n)$.

5. Visszavezetjük a problémára a TG-ek megállási problémáját az üres szón. Indirekt tegyük fel, hogy egy \mathcal{S} Turing-gép el tudja dönteni egy tetszőleges \mathcal{M} TG-ről, hogy minden szón megáll-e. Készítünk egy \mathcal{S}' TG-et, mely \mathcal{M} -ről eldönti, hogy megáll-e az üres szón. \mathcal{S}' egy \mathcal{M} inputon (valójában persze az \mathcal{M} TG kódja az input) a következőt csinálja: Először legyártja \mathcal{M} -ből azt az \mathcal{M}' TG-et, mely először úgy működik, hogy letörli az input szót, bármi is legyen az, majd pontosan úgy működik, ahogy \mathcal{M} . \mathcal{S}' ezek után elindítja az \mathcal{S} TG-et az \mathcal{M}' inputtal. Az indirekt feltevésünk szerint az \mathcal{S} TG ad egy választ, hogy \mathcal{M}' megáll-e minden inputon, vagy sem. Vegyük észre, hogy \mathcal{M}' konstrukciója alapján \mathcal{M}' pontosan akkor áll meg minden inputon, ha \mathcal{M} megáll az üres szón. Végül, \mathcal{S}' válaszoljon igennel (azaz menjen q_i állapotba), ha \mathcal{S} igen választ adott, nemmel, ha \mathcal{S} nem választ adott. Tehát az \mathcal{S}' TG egy olyan TG, amely eldönti, hogy egy tetszőleges \mathcal{M} TG megáll-e az üres szón, vagy sem. De erről a gyakorlaton/előadáson láttuk, hogy ilyen nincs.