Implementace Goldberga

Příklad 1: Uvažujme verzi Goldberga zvedající nejvyšší vrchol s přebytkem. Jaké informace potřebujeme udržovat, aby přístupy k hranám a vrcholům s přebytkem měly správnou složitost? Jak tyto informace udržovat při zvedání vrcholů a převádění po hranách tak, aby byla zachována složitost algoritmu?

Hradlové sítě

Pojmy: hradlo, vrstva, hloubka

Příklad 2: Dokažte, že n-bitový OR nelze spočítat v menší než logaritmické hloubce.

Příklad 3: Sestrojte hradlovou síť počítající majoritní funkci 4 bitů.

Příklad 4: Navrhněte hradlovou výhybku, tedy obvod s jedním i-bitovým vstupem p, 2^i k-bitovými vstupy $x_1, \ldots, x_{2^i}$, a jedním k-bitovým výstupem na který vydá x_p.

Příklad 5: Ukažte, že libovolnou funkci s k bitovými vstupy lze počítat hradlovou sítí o hloubce $O(k)$.

Důsledek: místo bitových vstupů můžeme uvažovat k-bitové, hloubka vzroste pouze konstantně-krát.

Příklad 6: Analyzujte hradlovou síť pro sčítání n-bitových čísel v hloubce $O(log n)$.

Příklad 7: Navrněte hradlovou síť počítající součet m n-bitových čísel v hloubce $\mathcal{O}(log(n) + log(m))$

Příklad 8: Navrhněte síť na násobení čísel, a násobení matic.

Příklad 9: Navrhněte síť, která dostane na vstupu matici sousednosti (neorientovaného) grafu, a v rozhodne, zda je graf souvislý.

Příklad 10: Ukažte, že pokud simulujeme hradlovou síť (třeba na RAM), její velikost souvisí s potřebném časem, a její hloubka s potřebným prostorem.

Komparátorové sítě

Příklad 11: Navrhněte a analyzujte komparátorový insert-sort a bubble-sort.

Příklad 12: Dokažte, že komparátorová třídící síť (nebo obecně třídící algoritmus v porovnávacím modelu) je korektní právě tehdy když správně funguje pro všechny 0/1-vstupy.

Implementace Goldberga

Příklad 1: Uvažujme verzi Goldberga zvedající nejvyšší vrchol s přebytkem. Jaké informace potřebujeme udržovat, aby přístupy k hranám a vrcholům s přebytkem měly správnou složitost? Jak tyto informace udržovat při zvedání vrcholů a převádění po hranách tak, aby byla zachována složitost algoritmu?

Hradlové sítě

Pojmy: hradlo, vrstva, hloubka

Příklad 2: Dokažte, že n-bitový OR nelze spočítat v menší než logaritmické hloubce.

Příklad 3: Sestrojte hradlovou síť počítající majoritní funkci 4 bitů.

Příklad 4: Navrhněte hradlovou výhybku, tedy obvod s jedním i-bitovým vstupem p, 2^i k-bitovými vstupy $x_1, \ldots, x_{2^i}$, a jedním k-bitovým výstupem na který vydá x_p.

Příklad 5: Ukažte, že libovolnou funkci s k bitovými vstupy lze počítat hradlovou sítí o hloubce $O(k)$.

Důsledek: místo bitových vstupů můžeme uvažovat k-bitové, hloubka vzroste pouze konstantně-krát.

Příklad 6: Analyzujte hradlovou síť pro sčítání n-bitových čísel v hloubce $O(logn)$.

Příklad 7: Navrněte hradlovou síť počítající součet m n-bitových čísel v hloubce $\mathcal{O}(log(n) + log(m))$

Příklad 8: Navrhněte síť na násobení čísel, a násobení matic.

Příklad 9: Navrhněte síť, která dostane na vstupu matici sousednosti (neorientovaného) grafu, a v rozhodne, zda je graf souvislý.

Příklad 10: Ukažte, že pokud simulujeme hradlovou síť (třeba na RAM), její velikost souvisí s potřebném časem, a její hloubka s potřebným prostorem.

Komparátorové sítě

Příklad 11: Navrhněte a analyzujte komparátorový insert-sort a bubble-sort.

Příklad 12: Dokažte, že komparátorová třídící síť (nebo obecně třídící algoritmus v porovnávacím modelu) je korektní právě tehdy když správně funguje pro všechny 0/1-vstupy.

Domácí úkol 4: komparátor

Navrhněte hradlovou síť, která přijme dvě k-bitová čísla x, y jako bity $x_1, x_2, \ldots, x_k$ a $y_1, y_2, \ldots, y_k$, a na dva k-bitové výstupy m, v (taktéž reprezentované jako sady bitů) vydá menší a větší z čísel.

Analyzujte velikost a hloubku sítě.

Domácí úkol 4: komparátor

Navrhněte hradlovou síť, která přijme dvě k-bitová čísla x, y jako bity $x_1, x_2, \ldots, x_k$ a $y_1, y_2, \ldots, y_k$, a na dva k-bitové výstupy m, v (taktéž reprezentované jako sady bitů) vydá menší a větší z čísel.

Analyzujte velikost a hloubku sítě.