1. Hledejme vrcholové pokrytí následujícím hladovým algoritmem. V každém kroku vybereme vrchol nejvyššího stupně, přidáme ho do pokrytí a odstraníme ho z grafu i se všemi již pokrytými hranami. Je nalezené pokrytí nejmenší? Nebo alespoň $\mathcal{O}(1)$-aproximace nejmenšího?

2. Vymyslete 2-aproximační algoritmus pro vrcholové pokrytí.

3. *Problém MaxCut:* vrcholy zadaného grafu chceme rozdělit do dvou množin tak, aby mezi množinami vedlo co nejvíce hran. Jinými slovy chceme nalézt bipartitní podgraf s co nejvíce hranami. Rozhodovací verze tohoto problému je NP-úplná, optimalizační verzi zkuste v polynomiálním čase 2-aproximovat.

4. Uvažujme následující algoritmus pro nejmenší vrcholové pokrytí grafu. Graf projdeme do hloubky, do výstupu vložíme všechny vrcholy vzniklého DFS stromu kromě listů. Dokažte, že vznikne vrcholové pokrytí a že 2-aproximuje to nejmenší.

5. V daném orientovaném grafu hledáme acyklický podgraf s co nejvíce hranami. Navrhněte polynomiální 2-aproximační algoritmus.

6. Řešení problému obchodního cestujícího hrubou silou by prohledávalo graf do hloubky a zkoušelo všechny hamiltonovské kružnice. To může v grafu na n vrcholech trvat až $n!$ kroků. Pokuste se najít rychlejší algoritmus. Dynamickým programováním lze dosáhnout složitosti $\mathcal{O}(2^n \cdot n^k)$ pro konstantní k. To je sice exponenciální, ale stále mnohem lepší než faktoriál.