

Příklad 1.

Určete, zda je zadaný neorientovaný graf strom.

Příklad 2.

Mějme adresářový strom. Najděte soubor, který je vnořený pod nejvíce podadresáři a spočítejte, kolik jich je.

Příklad 3.

Jakou časovou složitost bude mít prohledávání do šířky, pokud jako reprezentaci grafu použijeme matici sousednosti?

Příklad 4.

Náš robot uvíznul v rozpadajícím se bludišti reprezentovatelném čtvercovou mřížkou. O každém políčku víme, za kolik kroků přestane být průchozí (nula kroků znamená zeď). Může se robot dostat ven? Pokud ano, kolik kroků mu to potrvá?

Příklad 5.

Hrajete deskovku, kde se pohybujete svou postavičkou po mapě tvořící strom. V každém vrcholu se nachází nějaké množství zlaťáků, které si můžete „převzít“. Pokud tak však učiníte, už si nemůžete vzít zlaťáky ze sousedních vrcholů (protože jste právě upozornili místní hlídače).

Najděte vrcholy, ze kterých si zlaťáky „převezmete“ tak, abyste získali co nejvíce zlaťáků.

Příklad 6.

Najděte algoritmus, kterým zjistíte, zda je graf bipartitní.

Příklad 7.

Vraťme se k RAMu s konstantně rychlými neomezeně velkými buňkami. Ukažte, jak reprezentovat vektory, abychom uměli přičíst konkrétní složku, sečíst dva vektory nebo násobit skalárem v konstantním čase. Pro zjednodušení nejprve předpokládejte, že složka vektoru nikdy nepřesáhne konstantu 2^m .