

BI-AG1: Seznam Veta otázek (verze 29.11.2021)

- 1) Definujte podgraf a indukovaný podgraf. Definujte izomorfismus grafů a automorfismus grafu.
- 2) Definujte stupeň vrcholu, okolí vrcholu a regulární graf. Dále formulujte větu o principu sudosti a její důsledek pro počet vrcholů lichého stupně.
- 3) Definujte souvislou komponentu neorientovaného grafu a popište alespoň jeden algoritmus pro hledání souvislých komponent.
- 4) Popište algoritmus DFS, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 5) Definujte slabou souvislost a silnou souvislost orientovaného grafu.
- 6) Definujte acyklický orientovaný graf, zdroj a stok. Dále formulujte větu o existenci zdroje.
- 7) Definujte topologické uspořádání orientovaného grafu. Popište algoritmus konstrukce topologického uspořádání orientovaného grafu, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 8) Definujte strom, les, list. Formulujte tvrzení o existenci listů.
- 9) Formulujte větu o trhání listů. Dále formulujte alespoň jednu vlastnost stromů ekvivalentní s definicí stromu.
- 10) Definujte kostru neohodnoceného souvislého grafu a popište alespoň jeden algoritmus její konstrukce a diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 11) Definujte vzdálenost dvou vrcholů v neohodnoceném grafu. Dále popište algoritmus BFS a diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 12) Popište algoritmy SelectSort, BubbleSort a InsertSort a jejich časové složitosti. U jednoho z nich diskutujte jeho korektnost a uveďte jeho časovou složitost.
- 13) Definujte binární strom. Definujte binární minimovou haldu a popište její reprezentaci pomocí pole.
- 14) Popište operace vložení prvku a odebrání minima binární minimové haldy a jejich časové složitosti.
- 15) Popište algoritmus vytvoření binární minimové haldy z neuspořádaného pole (HeapBuild) a uveďte jeho časovou složitost.
- 16) Popište algoritmus HeapSort, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 17) Definujte binomiální strom a binomiální minimovou haldu. Popište operaci sloučení dvou binomiálních hald (BHMerge) a vložení prvku do binomiální haldy a jejich časovou složitost.
- 18) Definujte binomiální strom a binomiální minimovou haldu. Popište operaci odebrání minimálního prvku z binomiální minimové haldy (BHExtractMin) a její časovou složitost.
- 19) Definujte binární strom. Definujte binární vyhledávací strom a popište v něm operace nalezení prvku, následníka a předchůdce a jejich časové složitosti.
- 20) Definujte binární strom. Definujte binární vyhledávací strom a popište v něm operace vložení a odstranění prvku a uveďte jejich časové složitosti.
- 21) Definujte binární strom. Definujte AVL strom a formulujte větu o hloubce AVL stromu.
- 22) Definujte binární stroma AVL strom. Popište jednoduchou rotaci a dvojitou rotaci binárního vyhledávacího stromu a popište základní myšlenku vyvažování při vkládání prvku do AVL stromu.

- 23) Definujte binární strom a AVL strom. Popište jednoduchou rotaci a dvojitou rotaci binárního vyhledávacího stromu a popište základní myšlenku vyvažování při odebírání prvku z AVL stromu.
- 24) Definujte hešovací tabulku a popište principy hešování s řetízky.
- 25) Definujte hešovací tabulku a popište principy hešování s otevřenou adresací s lineárním přidáváním a s dvojitým hešováním.
- 26) Popište algoritmus MergeSort, diskutujte jeho korektnost a uveďte jeho časovou složitost.
- 27) Popište algoritmus rychlého násobení čísel, diskutujte jeho korektnost a uveďte jeho časovou složitost.
- 28) Popište algoritmus QuickSelect (nalezení k-tého nejmenšího prvku neseřazené posloupnosti), diskutujte jeho korektnost a uveďte jeho časovou složitost.
- 29) Popište algoritmus QuickSort, diskutujte jeho korektnost a uveďte jeho časovou složitost.
- 30) Definujte porovnávací model RAM. Vysvětlete větu o dolní mezi složitosti problému vyhledávání v porovnávacím modelu.
- 31) Definujte porovnávací model RAM. Vysvětlete větu o dolní mezi složitosti problému řazení v porovnávacím modelu.
- 32) Popište algoritmus CountingSort, jeho předpoklady, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 33) Popište algoritmus LexCountingSort, jeho předpoklady, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 34) Popište polynomiální algoritmus na výpočet nejdelší rostoucí podposloupnosti (na jaké podproblémy se rozkládá?). Diskutujte korektnost vašeho algoritmu.
- 35) Definujte editační vzdálenost dvou řetězců a popište polynomiální algoritmus na výpočet editační vzdálenosti dvou řetězců (na jaké podproblémy se rozkládá?). Diskutujte korektnost vašeho algoritmu.
- 36) Definujte elementární řez grafu a formulujte lemma o řezech ohodnoceného grafu.
- 37) Definujte minimální kostru ohodnoceného grafu. Popište Jarníkův algoritmus, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 38) Definujte minimální kostru ohodnoceného grafu. Popište Kruskalův algoritmus, diskutujte jeho konečnost a korektnost a uveďte jeho časovou složitost.
- 39) Popište základní myšlenku reprezentace struktury Union-Find pomocí keříků a uveďte časovou složitost operací Union a Find.
- 40) Popište Dijkstrův algoritmus pro hledání nejkratších cest v ohodnoceném orientovaném grafu, jeho předpoklady, diskutujte jeho korektnost a uveďte jeho časovou složitost.
- 41) Popište Bellman-Fordův algoritmus pro hledání nejkratších cest v ohodnoceném orientovaném grafu, jeho předpoklady, diskutujte jeho korektnost a uveďte jeho časovou složitost.

Vysvětlující komentář: Tímto upřesňujeme požadavky na ústní zkoušku. U otázek týkajících se popisu algoritmů a jejich konečnosti a korektnosti očekáváme od studenta stručný popis algoritmu – slovní nebo v pseudokódu – a stručné vysvětlení, proč se algoritmus v konečném počtu kroků zastaví (např. algoritmus otevře každý vrchol nejvýše jednou) a proč funguje správně (např. sloučením dvou seřazených posloupností vznikne seřazená posloupnost a to se děje rekurzivně). Pokud je časová složitost algoritmu závislá na hodnotách vstupních dat, měl by student uvést nejen složitost v nejhorším případě, ale i např. v nejlepším či průměrném, pokud je příslušná analýza součástí přednášek.