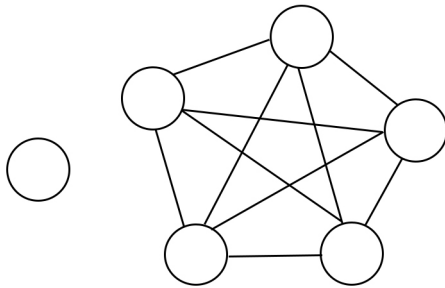
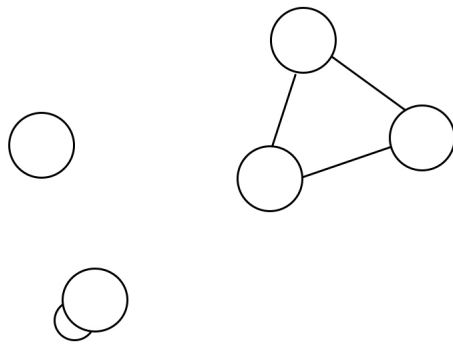


Graf: 6 hörn, 10 kanter och 2 sammanhängande komponenter.

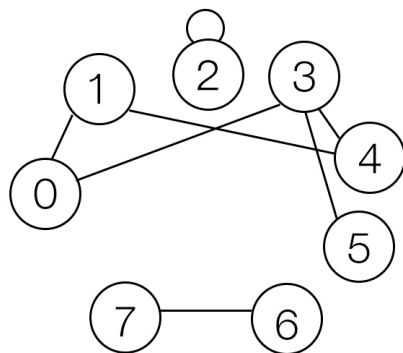


Graf: 5 hörn, 4 kanter och 3 komponenter



Låt G vara en oriktad graf som består av 8 hörn numrerade från 0 till 7 och kantmängden $\{(0,1), (0,3), (1,4), (2,2), (3,4), (3,5), (6,7)\}$.

Rita G .



Fråga: Ange ordningen som hörnen besöks vid en djupetförstsökning (DFS) med start i hörn 0.

Ordningen som hörnen besöks i är: 0 1 4 3 5

Fråga: Ange ordningen som hörnen besöks vid en breddenförstsökning (BFS) med start i hörn 0.

Ordningen som hörnen besöks i är: 0 1 3 4 5

Fråga: Skulle du representera en graf med hjälp av en närhetsmatris eller med hjälp av närhetslistor i följande fall? Motivera dina svar.

Fråga: Grafen har 1000 hörn och 2000 kanter och det är viktigt att vara sparsam med minnet.

Matrisen skulle innehålla en miljon element och många av dessa är tomma, därför är listan en vinnare.

Fråga: Grafen har 1000 hörn och 50000 kanter och det är viktigt att vara sparsam med minnet.

Matrisen innehåller fortfarande en miljon element, visserligen är det färre nollor i detta scenario men listor är ändå mer sparsamma.

Fråga: Det är viktigt att snabbt (på konstant tid) kunna avgöra om två hörn är grannar. Om möjligt vill du också vara sparsam med minnet.

När det gäller grannar är matriser markant mycket snabbare, detta eftersom att de möjliggör direkt tillgång av grannar. Tillskillnad från listor där det krävs att man loopar igenom listan tills att man finner svaret.

Fråga: Förklara varför DFS tar $\Theta(n^2)$ tid för en sammanhängande graf med n hörn om grafen representeras med en närhetsmatris.

För varje iteration behövs ett element tills att det inte längre finns några kvar. Summan blir då:

$$n + n - 1 + n - 2 \cdots 2 + 1 + 0$$

vilket är lika med $\frac{n \times n}{2}$ eftersom att konstanter inte gör någon skillnad vid benämning av ordo-notation blir tidskomplexiteten därför kvadratisk.