INSTITUT FOR DATALOGI, AARHUS UNIVERSITET

Science and Technology
EKSAMEN
Algoritmer og Datastrukturer 2 (2003-ordning)
Antal sider i opgavesættet (incl. forsiden): 9 (ni)
Eksamensdag: Torsdag den 22. august 2013, kl. 9.00-13.00
Tilladte medbragte hjælpemidler: Alle sædvanlige hjælpemidler (lærebøger, notater, lommeregner). Computer må ikke medbringes.
Materiale der udleveres til eksaminanden:

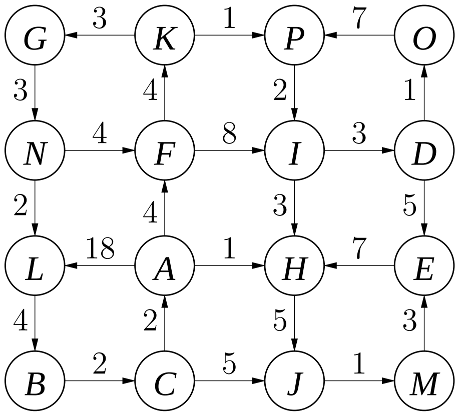
OPGAVETEKSTEN
BEGYNDER
PÅ NÆSTE SIDE

—oOo—

Opgave 1 (25%)

Bemærk: *Bagerst i eksamenssættet findes sider til at angive svarene til opgave 1 på og som afleveres som del af eksamensbesvarelsen.*

Betragt nedenstående vægtede orienterede graf. Det antages, at grafen er givet ved *incidenslister*, hvor incidenslisterne er sorteret alfabetisk.

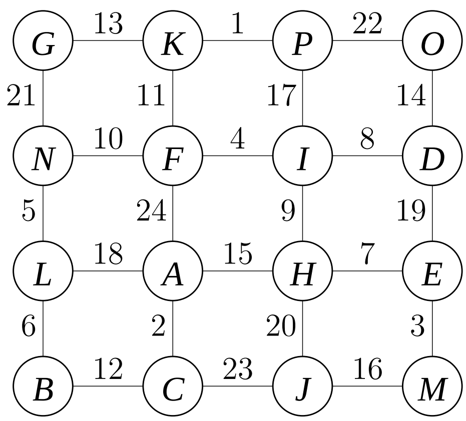


Spørgsmål a: Angiv et BFS-træ for ovenstående graf, hvor BFS-gennemløbet starter i knuden A. Angiv kanterne i BFS-træet og BFS-numrene for knuderne. ☐

Spørgsmål b: Angiv et DFS-træ for ovenstående graf, hvor DFS-gennemløbet starter i knuden A, og en DFS-nummerering af knuderne. Angiv for hver knude "discovery time" og "finishing time". ☐

Spørgsmål c: Angiv et korteste veje (SSSP) træ for ovenstående graf, når korteste veje beregningen sker med hensyn til startknuden A. For hver knude v angiv også afstanden fra startknuden A til v. ☐

Spørgsmål d: Angiv de stærke sammenhængskomponenter i ovenstående graf. For hver komponent angiv knuderne i komponenten. ☐



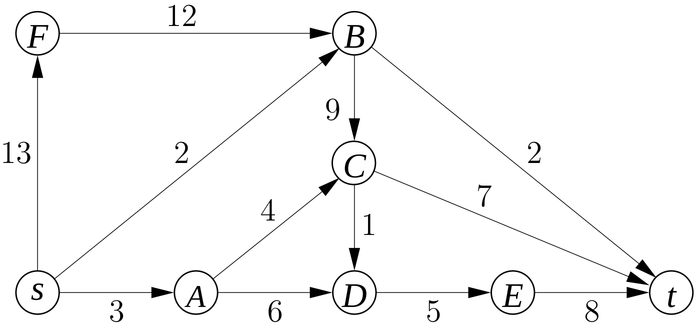
Spørgsmål e: Angiv kanterne i et minimum udspændende træ (MST) for ovenstående uorienterede graf med vægte på kanterne. ☐

(Opgavesættet fortsætter)

Opgave 2 (15%)

Bemærk: *Bagerst i eksamenssættet findes en side til at angive svarene til opgave 2 på og som afleveres som del af eksamensbesvarelsen.*

Betragt nedenstående netværk med de angivne kapaciteter på kanterne.



Spørgsmål a: Angiv en maksimal strømning (maximum flow) fra s til t i netværket (angiv for hver kant strømningen langs kanten), angiv værdien af en maksimal strømning, og angiv et snit (cut) (dvs. opdeling af knuderne i to disjunkte mængder) hvor kapaciteten af snittet er lig værdien af en maksimal strømning. □

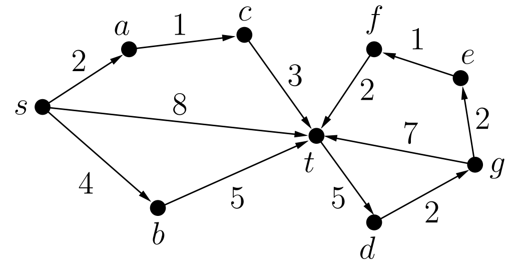
Spørgsmål b: Betragt Edmonds-Karp algoritmen anvendt på ovenstående graf til beregning af en maksimal strømning. Angiv de forbedrende stier (augmenting paths) der anvendes under udførslen af Edmonds-Karp algoritmen. For hver forbedrende sti angiv knuderne på stien og strømningen, man forbedrer med, langs stien. □

Opgave 3 (20%)

I denne opgave betragter vi korteste veje i positivt vægtede orienterede grafer. Input er en positivt vægtet orienteret graf og to knuder s og t. Vi lader n betegne antallet af knuder i grafen og m antallet af kanter.

Givet et heltal k, ønsker vi i denne opgave at finde korteste stier fra s til t som indeholder henholdvis *præcis* k kanter eller *mindst* k kanter.

I nedenstående graf er den korteste vej fra s til t med præcis 5 kanter stien s-b-t-d-g-t, som har længde 23. Den korteste sti fra s til t med mindst 9 kanter er stien s-a-c-t-d-g-e-f-t-d-g-e-f-t, som indeholder 13 kanter og har længde 30. Bemærk at en sti kan godt besøge en knude flere gange.



Spørgsmål a: Angiv en algoritme til at beregne længden af den korteste sti fra s til t med *præcis* k kanter. Angiv algoritmens udførselstid som funktion af n, m og k. □

Spørgsmål b: Angiv en algoritme til at beregne længden af den korteste sti fra s til t med *mindst* k kanter. Angiv algoritmens udførselstid som funktion af n, m og k. □

Opgave 4 (20%)

Lagrange's Fire-Kvadrat Sætning (fra 1770) siger at ethvert ikke-negativt heltal n kan skrives som summen af kvadraterne på højest fire positive heltal. F.eks. kan 33 og 7 skrives som $33 = 4^2 + 4^2 + 1^2 = 4^2 + 3^2 + 2^2 + 2^2$ og $7 = 2^2 + 1^2 + 1^2 + 1^2$.

I denne opgave ønsker vi givet et ikke negativt heltal n at finde det mindste tal k, således at n kan skrives som summen af kvadraterne af k positive heltal, dvs. $n = a_1^2 + \cdots + a_k^2$ hvor $a_1, \ldots, a_k$ er positive heltal. For et givet n, lader vi $K(n)$ betegne dette k. F.eks. er $K(33) = 3$ og $K(7) = 4$.

$K(n)$ kan bestemmes ved følgende rekursionsformel:

$$K(n) = \begin{cases} 0 & \text{hvis } n = 0 \\ 1 + \min\{K(n - a^2) \mid a \text{ er et positivt heltal hvor } a^2 \leq n\} & \text{ellers} \end{cases}$$

Spørgsmål a: Angiv $K(19)$, og angiv 19 som summen af $K(19)$ kvadrater. □

Spørgsmål b: Angiv en algoritme baseret på dynamisk programmering, der givet et ikke-negativt heltal n, beregner $K(n)$. Angiv algoritmens udførselstid. □

Spørgsmål c: Udvid algoritmen fra spørgsmål b) til også at rapportere $K(n)$ heltal $a_1, \ldots, a_{K(n)}$ hvor summen af kvadraterne er lig n. Angiv algoritmens samlede udførselstid. □

Opgave 5 (20%)

Antag at vi er givet to strenge T_1 og T_2 af længde henholdsvis n_1 og n_2. Vi siger T_1 kan *genereres* ud fra T_2, hvis der findes strenge $S_1, \ldots, S_k$, således at T_1 er konkatenationen af $S_1, \ldots, S_k$, og hvert S_i er en delstreng af T_2. F.eks. kan nedenstående streng T_1 genereres som konkatenationen af $k = 6$ delstrenge af strengen T_2.

$$\begin{array}{l} \quad\;\; {\scriptstyle 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12\ 13} \\ T_1 = \underline{\mathtt{ABC}}\,\underline{\mathtt{BBA}}\,\underline{\mathtt{A}}\,\underline{\mathtt{BA}}\,\underline{\mathtt{C}}\,\underline{\mathtt{BBA}} \end{array}$$

$$T_2 = \mathtt{BBABC}$$

$$\begin{aligned} S_1 &= \mathtt{ABC} &&= T_2[3..5] \\ S_2 &= \mathtt{BBA} &&= T_2[1..3] \\ S_3 &= \mathtt{A} &&= T_2[3..3] \\ S_4 &= \mathtt{BA} &&= T_2[2..3] \\ S_5 &= \mathtt{C} &&= T_2[5..5] \\ S_6 &= \mathtt{BBA} &&= T_2[1..3] \end{aligned}$$

Hvis der findes et tegn i T_1 der ikke findes i T_2 er det ikke muligt at generere T_1 ud fra T_2. Modsat, hvis alle tegn i T_1 også forekommer i T_2 kan vi altid generere T_1 ud fra T_2 med $k = n_1$ delstrenge af længde 1.

I det følgende er vi interesseret i at finde det mindste k, således at T_1 kan genereres ud fra k delstrenge af T_2. Det antages at alle tegn i T_1 også forekommer i T_2, og at T_1 og T_2 er over et alfabet med $O(1)$ tegn.

Spørgsmål a: Angiv det mindste k og delstrenge $S_1, \ldots, S_k$ af T_2, således at T_1 kan skrives som konkatenationen $T_1 = S_1 \cdots S_k$.

$$\begin{array}{l} \quad\;\; {\scriptstyle 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12\ 13\ 14\ 15\ 16\ 17\ 18\ 19\ 20} \\ T_1 = \mathtt{ABBABCAABBCACBBACBAB} \end{array}$$

$$T_2 = \mathtt{CBABACBBBCABB}$$

□

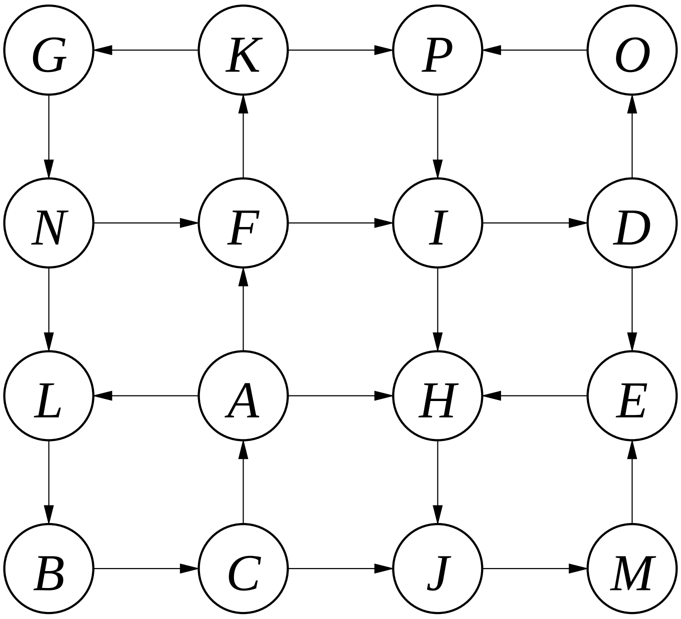
I det følgende kan det antages at et suffikstræ for en streng af længde n over et alfabet med $O(1)$ tegn kan konstrueres i $O(n)$ tid.

Spørgsmål b: Beskriv en algoritme der givet to strenge T_1 og T_2 af længde n_1 og n_2 og over et alfabet med $O(1)$ tegn, finder det mindste k, således at strengen T_1 kan genereres ud fra k delstrenge af T_2. Angiv algoritmens udførselstid. For at få fuld point for besvarelsen skal algoritmens udførselstid være $O(n_1 + n_2)$. □

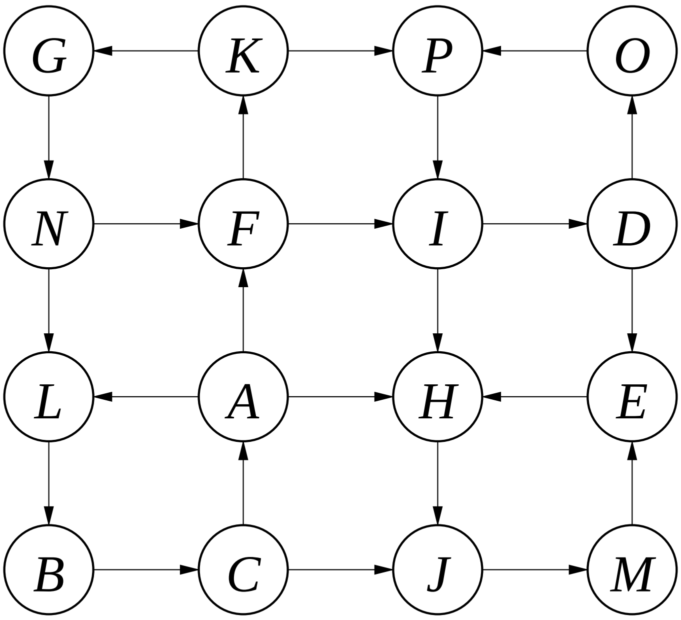
Årskort:	Navn:

Opgave 1 — Svar

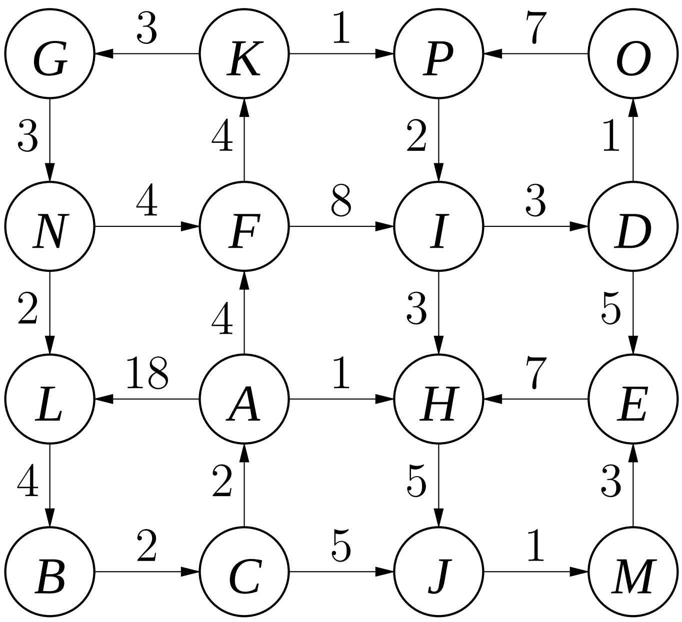
Spørgsmål a: BFS



Spørgsmål b: DFS



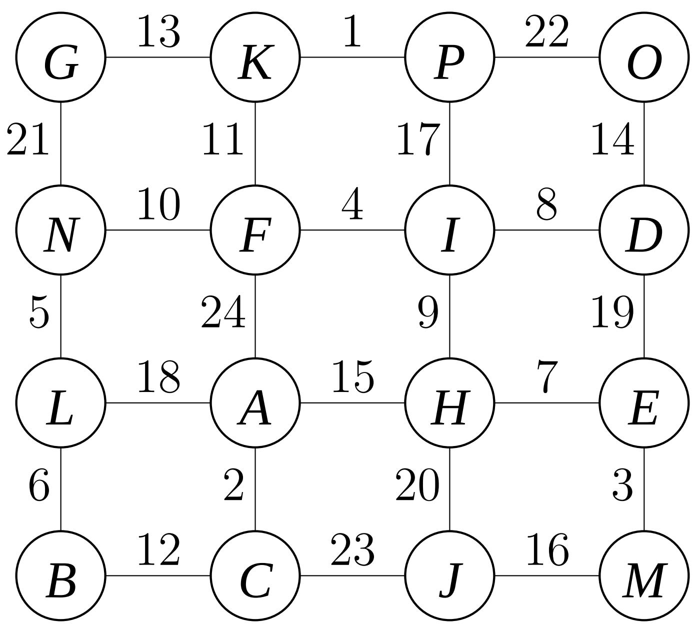
Spørgsmål c: SSSP



Spørgsmål d:

Stærke sammenhængskomponenter: ______________________________

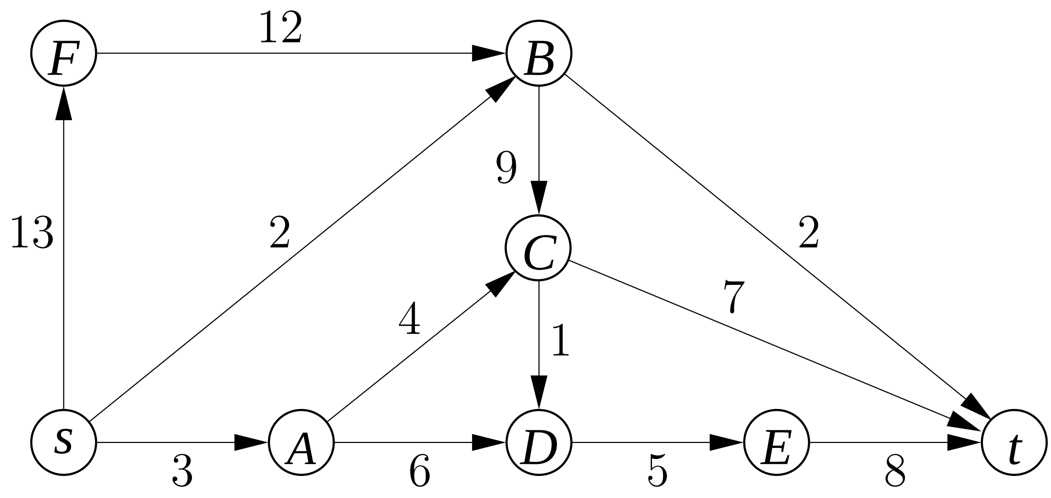
Spørgsmål e: MST



Årskort:	Navn:

Opgave 2 — Svar

Spørgsmål a:



Værdien af strømning: ____________________

Minimal snit: ____________________

Spørgsmål b:

Forbedring	Forbedrende sti