INSTITUT FOR DATALOGI, AARHUS UNIVERSITET

Science and Technology
EKSAMEN
Algoritmer og Datastrukturer 2 (2003-ordning)
Antal sider i opgavesættet (incl. forsiden): 11 (elleve)
Eksamensdag: Torsdag den 18. juni 2015, kl. 9.00-13.00
Tilladte medbragte hjælpemidler: Alle sædvanlige hjælpemidler (lærebøger, notater, lommeregner). Computer må ikke medbringes.
Materiale der udleveres til eksaminanden:

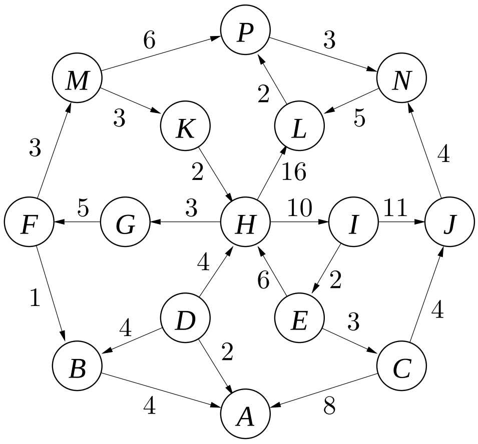
OPGAVETEKSTEN
BEGYNDER
PÅ NÆSTE SIDE

—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.

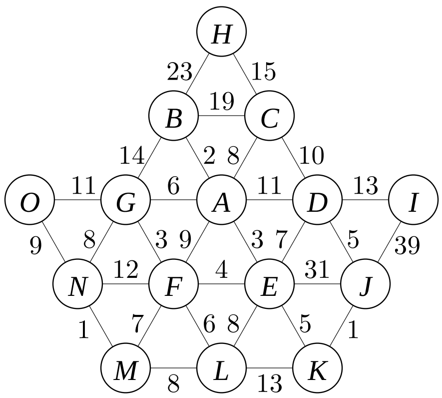


Spørgsmål a: Angiv et BFS-træ for ovenstående graf, hvor BFS-gennemløbet starter i **knuden** H. Angiv kanterne i BFS-træet, BFS-numrene for knuderne, og rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. □

Spørgsmål b: Angiv et DFS-træ for ovenstående graf, hvor DFS-gennemløbet starter i **knuden** H, og en DFS-nummerering af knuderne. Angiv for hver knude "discovery time" og "finishing time". Marker for hver kant om det er en "tree edge" (T), "back edge" (B), "cross edge" (C) eller "forward edge" (F). □

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

Spørgsmål d: Angiv et korteste veje (SSSP) træ for ovenstående graf, når korteste veje beregningen sker med hensyn til **startknuden** H. For hver knude v angiv også afstanden fra startknuden H til v, og angiv rækkefølgen knuderne tages ud af prioritetskøen Q i Dijkstra's algoritme. □

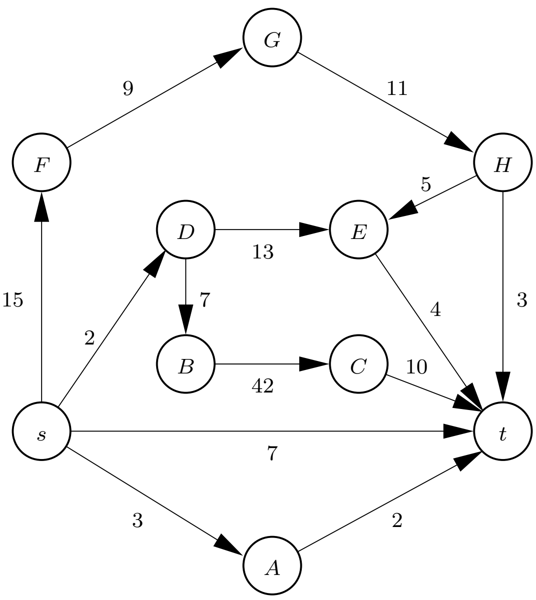


Spørgsmål e: Angiv kanterne i et minimum udspændende træ (MST) for ovenstående uorienterede graf med vægte på kanterne. Angiv i hvilken rækkefølge knuderne bliver taget ud af prioritetskøen Q i Prim's algoritme, når denne starter i **knuden** A. □

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%)

Betragt en uorienteret graf $G = (V, E)$ med $n = |V|$ knuder og $m = |E|$ kanter, og hvor hver kant $e \in E$ har en ikke negativ heltallig længde $w(e)$. Længden af en sti i grafen er summen af længderne af kanterne på stien, og afstanden $d(u, v)$ mellem to knuder $u \in V$ og $v \in V$ er længden af en korteste sti mellem u og v i grafen.

Lad $S \subseteq V$ være en mængde af *startknuder* og $T \subseteq V$ være en mængde af *målknuder*, ikke nødvendigvis disjunkte.

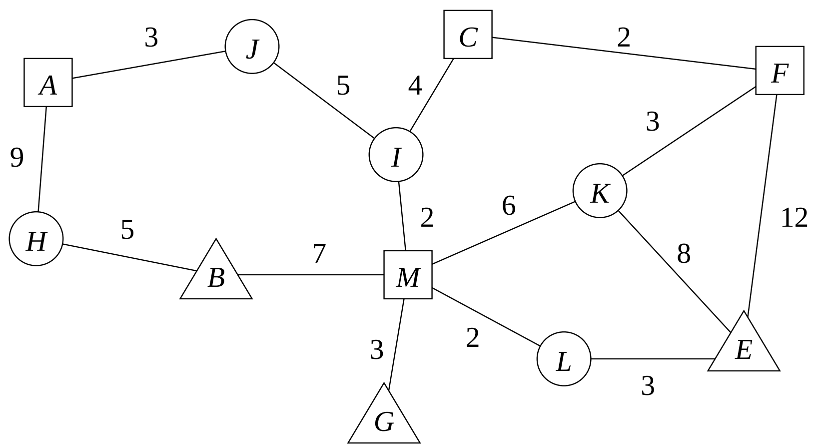
Vi antager at der i hver startknude starter præcis én agent, og hver agent højest kan bevæge sig en afstand D, hvor D er et positivt heltal der er identisk for alle agenter.

I nedenstående eksempel er $S = \{B, E, G\}$ og $T = \{A, C, M, F\}$ angivet med henholdsvis trekantede og firkantede knuder. Hvis vi for $v \in S$ lader $T_v \subseteq T$ betegne de målknuder man kan nå fra v med afstand D, så har vi for $D = 10$ at

$$T_B = \{M\},\ T_E = \{M\},\ \text{og}\ T_G = \{C, M\}.$$

Vi ønsker at flytte agenterne rundt, således at ingen agent flyttes mere end afstand D, og hvor der efterfølgende opnås et maksimalt antal målknuder hvor der står mindst én agent på. Vi lader k betegne dette maksimale antal målknuder der kan nås.

I nedenstående eksempel kan vi flytte agenten fra G til C og agenten fra E til M, således at vi har $k = 2$. Agenten i B bliver stående.



Spørgsmål a: Beskriv en algoritme der for alle startknuder $v \in S$ bestemmer hvilke målknuder T_v man kan nå fra v inden for afstand højest D. Angiv algoritmens udførselstid. Bemærk at output kan have størrelse op til n^2 når alle knuder er startknuder og disse kan nå alle knuder inden for afstand D. □

Spørgsmål b: Beskriv en algoritme der finder det maksimale antal målknuder k der kan nås med agenter, når der starter præcis én agent i hver startknude. Angiv algoritmens udførselstid. □

Opgave 4 (20%)

For to strenge $S[1..n]$ og $T[1..m]$ af længde n og m er en *fælles koncentreret delsekvens* en sekvens der forekommer som en delsekvens i både i S og T, og hvor der imellem to udvalgte positioner fra S (henholdsvis T) højest findes én position der ikke er udvalgt.

I denne opgave ønsker vi at finde en længste fælles koncentreret delsekvens. Mere præcist ønsker vi at finde det maksimale k, hvor der findes $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ og $1 \leq j_1 < j_2 < \cdots < j_k \leq m$ således at $S[i_1] = T[j_1], S[i_2] = T[j_2], \ldots, S[i_k] = T[j_k]$ og $i_{\ell+1} - i_\ell \leq 2$ og $j_{\ell+1} - j_\ell \leq 2$ for $1 \leq \ell < k$.

For eksempel har strengene $S = \texttt{abxy}\underline{\texttt{ab}}\texttt{x}\underline{\texttt{c}}\texttt{x}\underline{\texttt{da}}\texttt{x}$ og $T = \texttt{abxazd}\underline{\texttt{a}}\texttt{b}\underline{\texttt{b}}\texttt{c}\underline{\texttt{cd}}\texttt{y}\underline{\texttt{a}}\texttt{z}$ en længste fælles koncentreret delsekvens `abcda` af længde 5.

Vi lader $L(i,j)$ betegne længden af en længste fælles koncentreret delsekvens hvor $S[i]$ og $T[j]$ indgår som det sidste element i delsekvensen. Hvis $S[i] \neq T[j]$ så er $L(i,j) = 0$.

For ovenstående eksempel har vi $L(8,11) = 3$, da `abc` er den længste fælles koncentreret delsekvens som indeholder det sidste tegn i $S[1..8] = \texttt{abxy}\underline{\texttt{ab}}\texttt{x}\underline{\texttt{c}}$ og $T[1..11] = \texttt{abxazd}\underline{\texttt{a}}\texttt{b}\underline{\texttt{b}}\texttt{c}\underline{\texttt{c}}$,

$L(i,j)$ kan bestemmes ved følgende rekursionsformel, hvor vi antager at $-1 \leq i \leq n$ og $-1 \leq j \leq m$. Bemærk at for at simplicere rekursionsformlen tillader vi i og j at antage værdierne $\{0, -1\}$, som begge angiver at vi betragter det tomme præfiks af en af strengene.

$$L(i,j) = \begin{cases} 0 \quad \text{hvis } i \leq 0 \ \vee \ j \leq 0 \ \vee \ S[i] \neq T[j] \\ 1 + \max\{L(i-1,j-1), L(i-2,j-1), L(i-1,j-2), L(i-2,j-2)\} \\ \hfill \text{hvis } 1 \leq i \leq n \ \wedge \ 1 \leq j \leq m \ \wedge \ S[i] = T[j] \end{cases}$$

Spørgsmål a: Udfyld nedenstående tabel for strengene $S = \texttt{abbc}$ og $T = \texttt{acbca}$.

$L(i,j)$	-1	0	1	2	3	4	5
-1							
0							
1							
2							
3							
4							

□

Spørgsmål b: Angiv en algoritme baseret på dynamisk programmering, der givet to strenge S og T af længde henholdsvis n og m, beregner længden af en længste fælles koncentreret delsekvens. Angiv algoritmens udførselstid. □

Spørgsmål c: Udvid algoritmen fra spørgsmål b) til at udskrive en længste fælles koncentreret delsekvens for to strenge S og T. Angiv algoritmens udførselstid. □

Opgave 5 (20%)

For en streng T af længde n over et alfabet med $O(1)$ tegn, ønsker vi at finde det mindste heltal k, således at alle delstrenge af T med længde k er forskellige. For $T =$ `abcabde` får vi at $k = 3$, da alle delstrengene af længde 3 (`abc`, `bca`, `cab`, `abd` og `bde`) er forskellige, hvorimod der findes to identiske delstrenge af længde 2, nemlig $T[1..2] = T[4..5] =$ `ab`.

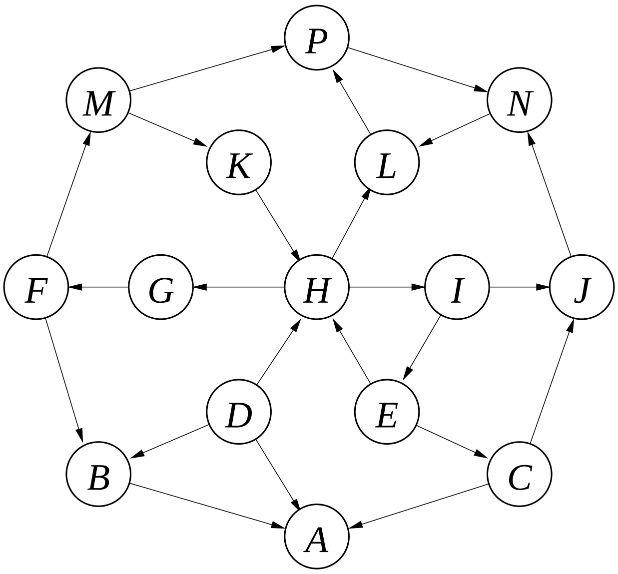
I det følgende kan det antages at et suffikstræ for en streng af længde n over et alfabet med $O(1)$ tegn kan konstrueres i $O(n)$ tid.

Spørgsmål a: Beskriv en algoritme der givet en streng af længde n, finder det mindste k, således at alle delstrenge af T af længde k er forskellige. Angiv algoritmens udførselstid. En ideel løsning opnår udførselstid $O(n)$. □

Spørgsmål b: Beskriv en algoritme der givet en streng af længde n, finder det mindste k, således at der findes en delstreng P af T af længde k, som forekommer præcis én gang i T. Angiv algoritmens udførselstid. En ideel løsning opnår udførselstid $O(n)$. □

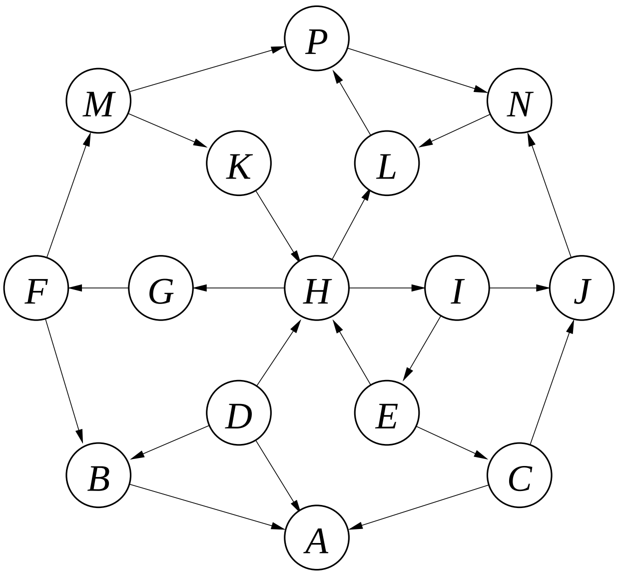
Opgave 1 — Svar

Spørgsmål a: BFS



Indsættelser i Q: ______________________________

Spørgsmål b: DFS



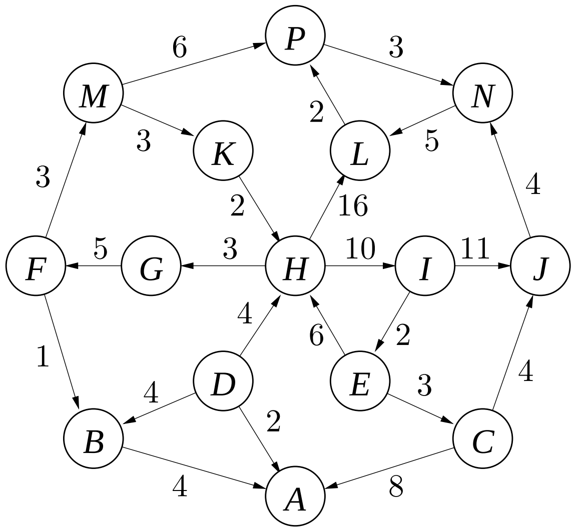
(blank side)

Opgave 1 — Svar

Spørgsmål c:

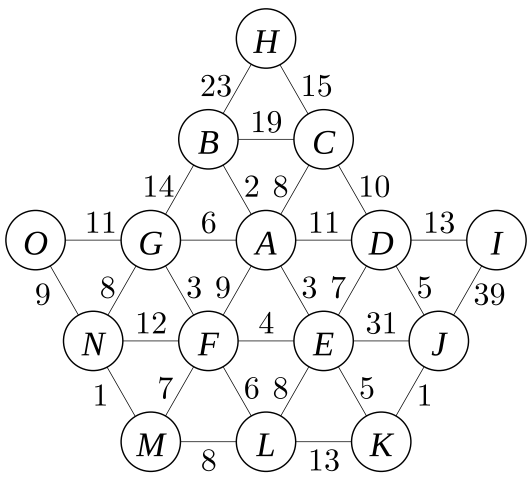
Stærke sammenhængskomponenter: ______________________

Spørgsmål d: SSSP



Udtagelse fra Q: ______________________

Spørgsmål e: MST



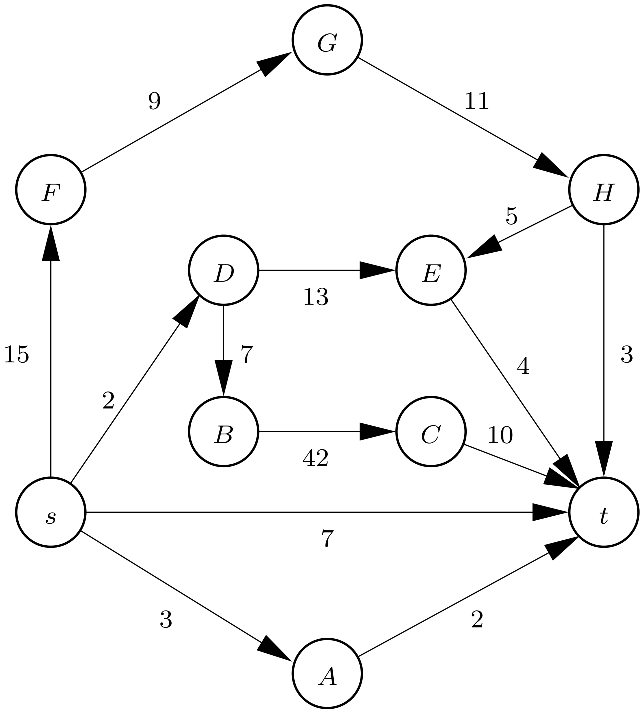
Udtagelse fra Q: ______________________

(blank side)

Årskort:	Navn:

Opgave 2 — Svar

Spørgsmål a:



Værdien af strømning: ______________________

Minimal snit: ______________________

Spørgsmål b:

Forbedring	Forbedrende sti