INSTITUT FOR DATALOGI, AARHUS UNIVERSITET

Science and Technology
EKSAMEN
Algoritmer og Datastrukturer 2 (2003-ordning)
Antal sider i opgavesættet (incl. forsiden): 11 (elleve)
Eksamensdag: Mandag den 21. august 2017, kl. 9.00-13.00
Tilladte medbragte hjælpemidler: Alle sædvanlige hjælpemidler (lærebøger, notater, lommeregner). Computer må ikke medbringes.
Materiale der udleveres til eksaminanden:

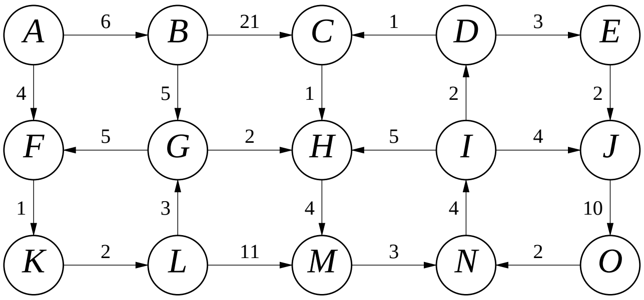
OPGAVETEKSTEN
BEGYNDER
PÅ NÆSTE SIDE

—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.

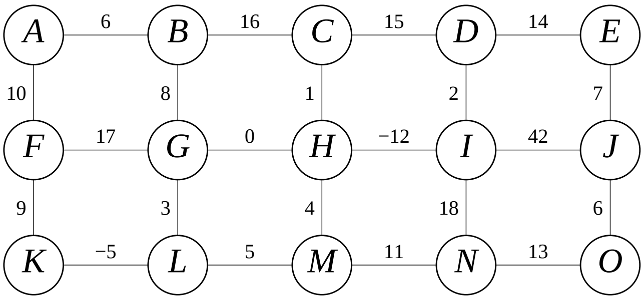


Spørgsmål a: Angiv et BFS-træ for ovenstående graf, hvor BFS-gennemløbet starter i **knuden** A. Angiv kanterne i BFS-træet, BFS-numrene for knuderne, og rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. □

Spørgsmål b: Angiv et DFS-træ for ovenstående graf, hvor DFS-gennemløbet starter i **knuden** A, 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** A. For hver knude v angiv også afstanden fra startknuden A til v, og angiv rækkefølgen knuderne tages ud af prioritetskøen Q i Dijkstra's algoritme. □

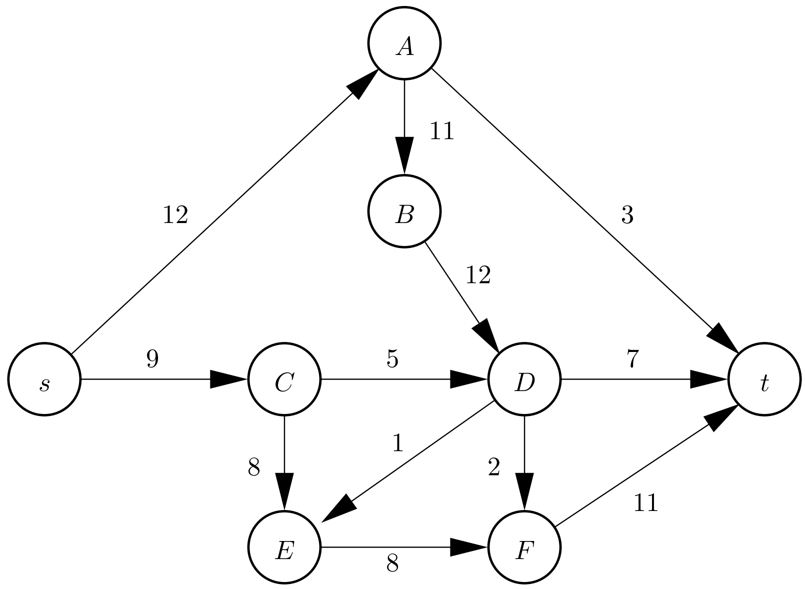


Spørgsmål e: Angiv kanterne i et minimum udspændende træ (MST) for ovenstående uorienterede graf med vægte på kanterne. Angiv i hvilken rækkefølge knuderne bliver taget ud af prioritetskøen Q i Prim's algoritme, når denne starter i **knuden** A. □

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.

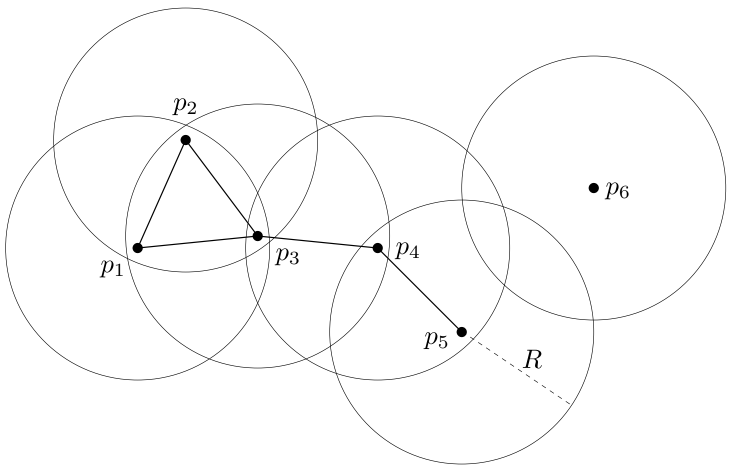


Spørgsmål a: Angiv en maksimal strømning (maximum flow) fra s til t i netværket (angiv for hver kant strømningen langs kanten), angiv værdien af en maksimal strømning, og angiv et snit (cut) (dvs. opdeling af knuderne i to disjunkte mængder) hvor kapaciteten af snittet er lig værdien af en maksimal strømning.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

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%)

I denne opgave antages at vi har givet n punkter $\mathcal{P} = \{p_1, \ldots, p_n\}$ i planen. For alle par af punkter (p_i, p_j) lader vi $d_{i,j}$ betegne afstanden mellem punkterne. For en givet *kommunikationsradius* R, så kan p_i og p_j *kommunikere direkte* hvis $d_{i,j} \leq R$, og p_i og p_j kan *kommunikere* (direkte eller indirekte) hvis der findes en sti $p_i, q_1, \ldots, q_k, p_j$ blandt punkterne i $\mathcal{P}$ således at alle par af hinanden efterfølgende punkter på stien kan kommunikere direkte. I nedenstående eksempel er direkte kommunikation (for kommunikationsradiusen R) angivet ved kanter mellem punkterne. p_1 og p_5 kan kommunikere langs stien p_1, p_3, p_4, p_5, hvorimod p_2 og p_6 ikke kan kommunikere da der ingen sti er fra p_2 til p_6.



Spørgsmål a: Beskriv en algoritme der givet n punkter $\mathcal{P}$, deres indbyrdes afstande, og en kommunikationsradius R, og to punkter p_i og p_j i $\mathcal{P}$, beregner den korteste kommunikationssti mellem p_i og p_j, d.v.s. en sti med færrest mulig punkter med direkte kommunikation. Angiv algoritmens udførselstid. □

Spørgsmål b: Beskriv en algoritme der givet n punkter $\mathcal{P}$ og deres indbyrdes afstande, finder den mindste kommunikationsradius R, således at alle par af punkter kan kommunikkere sammen. Angiv algoritmens udførselstid. □

Spørgsmål c: Beskriv en algoritme der givet n punkter $\mathcal{P}$, deres indbyrdes afstande, og en kommunikationsradius R, og to punkter p_i og p_j i $\mathcal{P}$ uden direkte kommunikation, afgør om der findes to kommunikationsstier mellem p_i og p_j som ingen punkter har tilfælles (ud over p_i og p_j). Angiv algoritmens udførselstid. □

Spørgsmål d: Beskriv en algoritme der givet n punkter $\mathcal{P}$, deres indbyrdes afstande, og to punkter p_i og p_j i $\mathcal{P}$, finder den mindste kommunikationsradius R, således at p_i og p_j kan kommunikere direkte eller der findes to kommunikationsstier mellem p_i og p_j som ingen punkter har tilfælles (ud over p_i og p_j). Angiv algoritmens udførselstid. □

Opgave 4 (25%)

Givet to strenge $P[1..m]$ og $T[1..n]$, så forekommer P som en *delsekvens* af T, hvis der findes positioner $i_1, i_2, \ldots, i_m$, i T hvor $1 \leq i_1 < i_2 < \cdots < i_m \leq n$ og $P[j] = T[i_j]$ for $j = 1..m$. F.eks. er `abc` en delsekvens af $\texttt{\underline{a}x\underline{bc}y}$, med $i_1 = 1$, $i_2 = 3$ og $i_3 = 4$.

Spørgsmål a: Angiv en algoritme der afgør om en streng $P[1..m]$ er en delsekvens af en streng $T[1..n]$. Angiv algoritmens udførselstid. □

Givet to strenge $P[1..m]$ og $T[1..n]$ ønsker vi at finde positioner $i_1 < i_2 < \cdots < i_m$ for en forekomst af P som delsekvens af T, hvor P forekommer som et *minimalt antal sammenhængende blokke*, dvs. således at $\{j \mid 1 \leq j < m \ \wedge \ i_j + 1 < i_{j+1}\}$ er mindst mulig. F.eks. så forekommer $P =$ `abcd` som delsekvens af $T =$ `aabxccd` på følgende fire måder: $\texttt{\underline{a}a\underline{b}x\underline{c}c\underline{d}}$, $\texttt{a\underline{ab}x\underline{c}c\underline{d}}$, $\texttt{\underline{a}a\underline{b}xc\underline{cd}}$, og $\texttt{a\underline{ab}xc\underline{cd}}$. Disse består af en opdeling af P i henholdsvis 4, 3, 3 og 2 blokke, hvorfor det er den sidste forekomst af P vi er interesseret i at finde.

For $0 \leq i \leq m$ og $0 \leq j \leq n$ lader vi $C(i, j)$ betegne det minimale antal blokke $P[1..i]$ skal deles op i for at forekomme som en blokket delsekvens af $T[1..j]$. Hvis $P[1..i]$ ikke forekommer som en delsekvens af $T[1..j]$, så lader vi $C(i, j) = +\infty$.

$C(i, j)$ kan beskrives ved følgende rekursionsformel:

$$C(i,j) = \begin{cases} 0 & \text{hvis} \quad i = 0 \\ +\infty & \text{hvis} \quad i > j \\ C(i, j-1) & \text{hvis} \quad 1 \leq i \leq j \wedge P[i] \neq T[j] \\ \min(C(i, j-1),\ 1 + C(i-d, j-d)) & \text{hvis} \quad 1 \leq i \leq j \wedge P[i] = T[j] \\ \qquad \text{hvor} \quad d = \max\{\ell \mid 1 \leq \ell \leq i \ \wedge \ P[i-\ell+1..i] = T[j-\ell+1..j]\} & \end{cases}$$

Spørgsmål b: Udfyld nedenstående tabel for $C(i, j)$ for $P =$ `abc` og $T =$ `bacbabbc`.

$C(i,j)$	0	1	2	3	4	5	6	7	8
0									
1									
2									
3									

□

Spørgsmål c: Angiv en algoritme baseret på dynamisk programmering, der givet strenge $P[1..m]$ og $T[1..n]$, beregner det minimale antal blokke i en forekomst af P som delsekvens i T, dvs. beregner $C(m, n)$. Angiv algoritmens udførselstid. □

Spørgsmål d: Udvid algoritmen til at rapportere positionerne i T for en minimal blokket forekomst af P. Angiv algoritmens udførselstid. □

Opgave 5 (15%)

Givet to strenge A og B af længde n, så siges B at være et *præfiks-suffiks swap* af A, hvis $B = A[k..n]A[1..k-1]$ for et k, hvor $1 \leq k \leq n$. Specielt er $B = A$ et præfiks-suffiks swap af A. F.eks. er $B =$ `deabc` et præfiks-suffiks swap af $A =$ `abcde`.

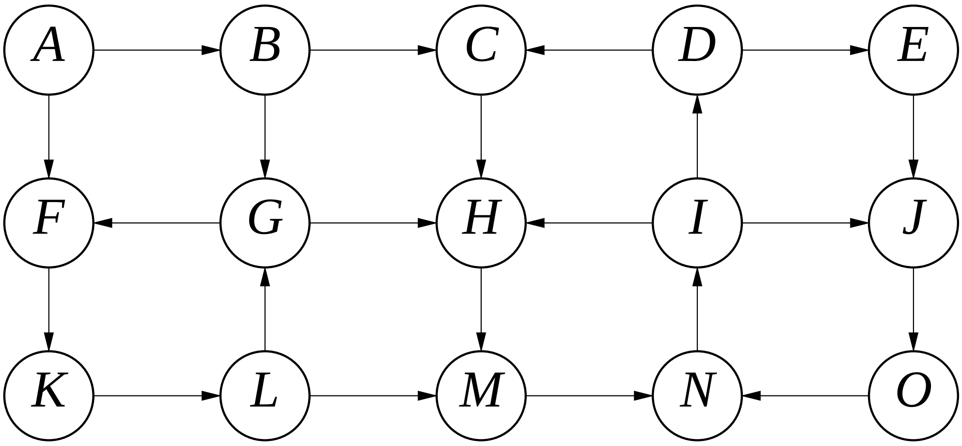
I det følgende kan det antages at strengene er over et alfabet med $O(1)$ tegn og at et suffikstræ for en streng af længde n over et alfabet med $O(1)$ tegn kan konstrueres i $O(n)$ tid.

Spørgsmål a: Beskriv en algoritme, der givet to strenge A og B af længde n, afgør om B er et præfiks-suffiks swap af A. Angiv algoritmens udførselstid. En ideel besvarelse opnår udførselstid $O(n)$. □

Spørgsmål b: Beskriv en algoritme, der givet to strenge P og T af henholdsvis længde m og n, $m \leq n$, afgør om et præfiks-suffiks swap af P forekommer som en delstreng i T. F.eks. forekommer et præfiks-suffix swap af $P =$ `abcde` i $T =$ `abcdeabced`. Angiv algoritmens udførselstid. En ideel besvarelse opnår udførselstid $O(n)$. □

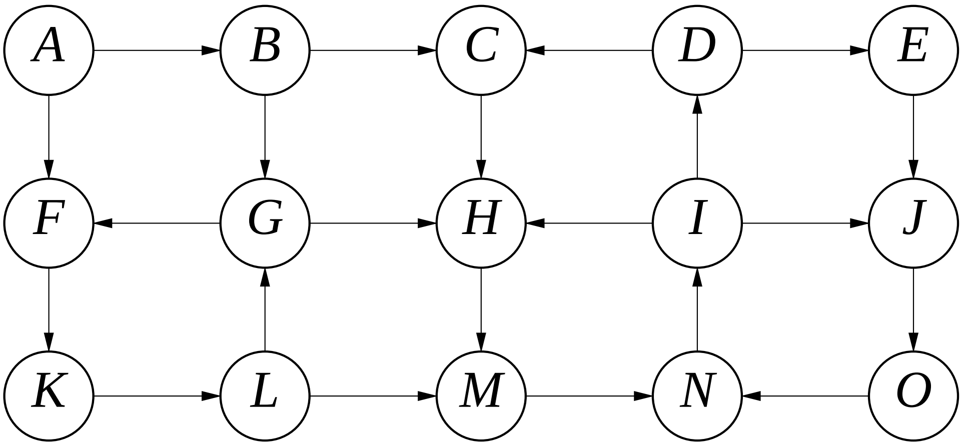
Opgave 1 — Svar

Spørgsmål a: BFS



Indsættelser i Q: ______________________________

Spørgsmål b: DFS



(blank side)

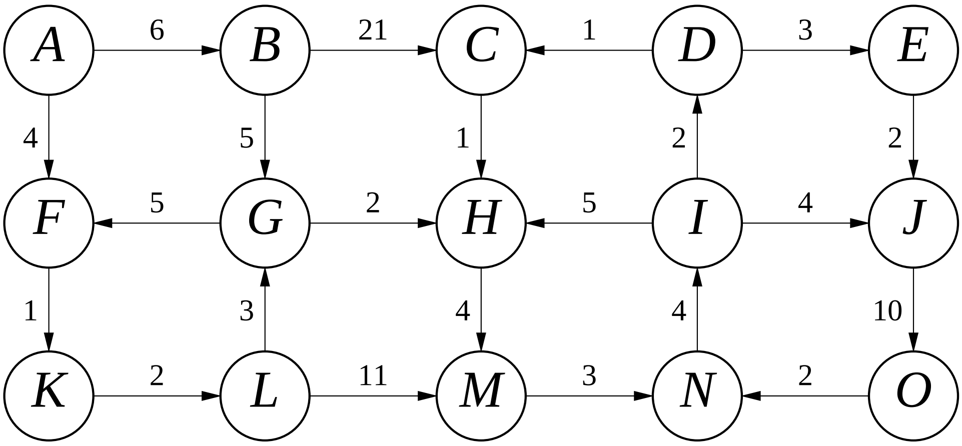
Årskort:	Navn:

Opgave 1 — Svar

Spørgsmål c:

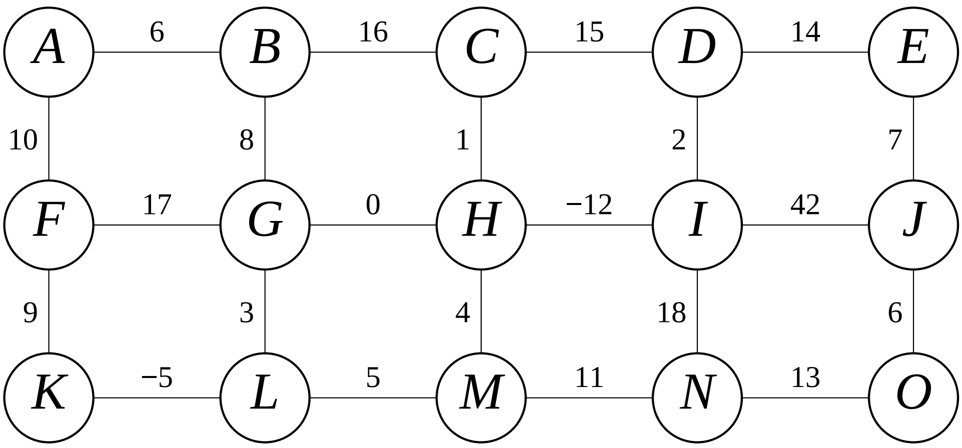
Stærke sammenhængskomponenter: ____________________

Spørgsmål d: SSSP



Udtagelse fra Q: ____________________

Spørgsmål e: MST



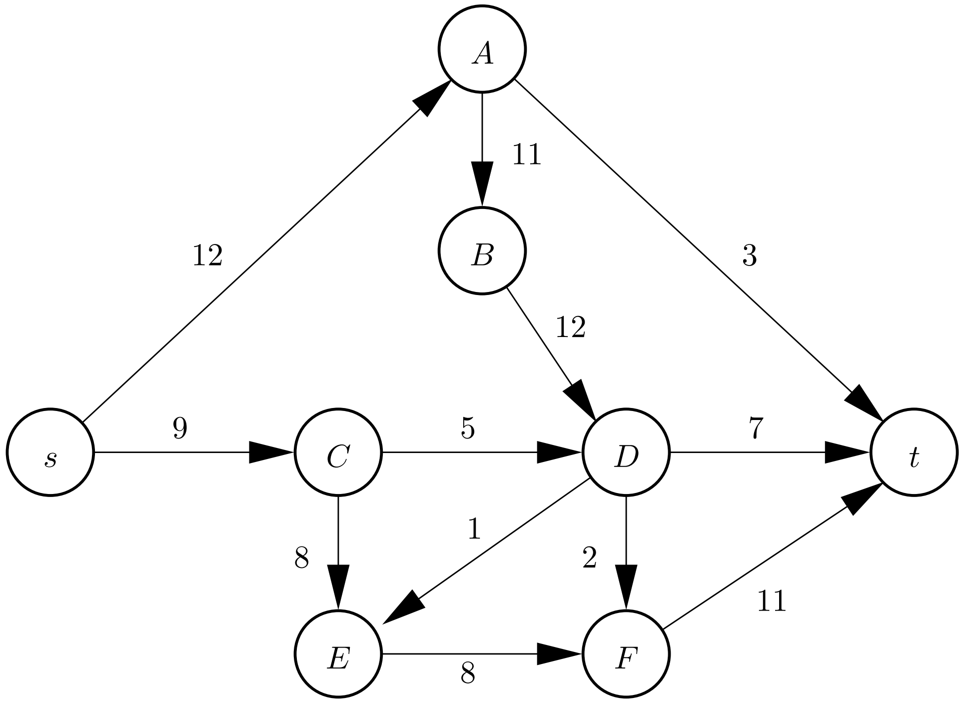
Udtagelse fra Q: ____________________

(blank side)

Årskort:	Navn:

Opgave 2 — Svar

Spørgsmål a:



Værdien af strømning: ______________________

Minimal snit: ______________________

Spørgsmål b:

Forbedring	Forbedrende sti