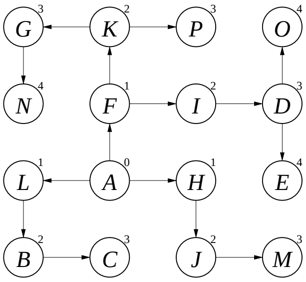
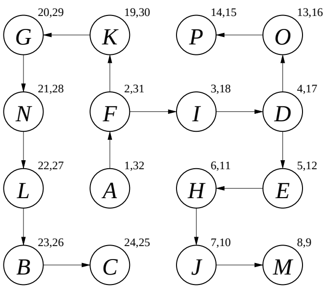
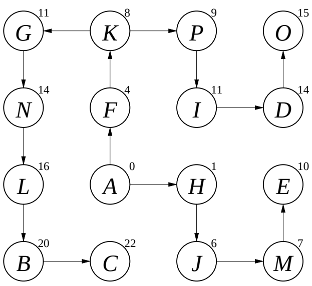
1a



1b



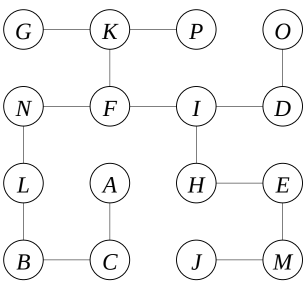
1c



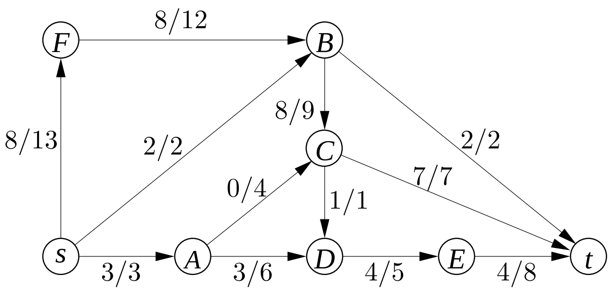
1d

$\{A, B, C, F, G, K, L, N\}, \{E, J, H, M\}, \{D, I, O, P\}$

1e



2a



Maximal strømning = 13.
Snit med kapacitet 13: $(\{s, B, C, F\}, \{A, D, E, t\})$

2b

Forbedring	Sti
2	sBt
3	$sACt$
4	$sFBCt$
1	$sFBCDEt$
3	$sFBCADEt$

3a

Lav $k+1$ kopier af grafens knuder, dvs. for en knude v i input grafen har vi knuder $v_0, v_1, \ldots, v_k$. For en kant (u, v) med vægt w laver vi kanter (u_i, v_{i+1}), dvs. $(u_0, v_1), (u_1, v_2), \ldots, (u_{k-1}, v_k)$, alle med vægt w. Kør DAG shortest path på denne graf for at finde afstanden fra s_0 til t_k. Tid $O((n+m)k)$.

3b

Udvid grafen fra a) med kanter (u_k, v_k) med vægt w hvis (u, v) er en kant i input-grafen med vægt w. Kør Dijkstra's algoritme på denne graf for at finde afstanden fra s_0 til t_k. Tid $O((n+m)k\log((n+m)k))$.

4a

$K(19) = 3$, $19 = 3^2 + 3^2 + 1^2$

4b

```
K[0]=0
A[0]=0
for i=1 to n
  K[i]=i
  A[i]=1
  a=1
  while a*a <= i
    if K[i]>K[i-a*a]+1 then
       K[i]=K[i-a*a]+1
       A[i]=a
    a=a+1
return K[n]
```

Tid $O(n\sqrt{n})$.

4c

```
<< første 11 linier af 4b >>
r=n
while (r>0)
  print A[r]
  r=r-A[r]*A[r]
```

Tid $O(n\sqrt{n})$.

5a

$k = 8$

$S_1 = \mathtt{A\,B\,B} = T_2[11..13]$
$S_2 = \mathtt{A\,B} \quad = T_2[3..4]$
$S_3 = \mathtt{C\,A} \quad = T_2[10..11]$
$S_4 = \mathtt{A\,B\,B} = T_2[11..13]$
$S_5 = \mathtt{C\,A} \quad = T_2[10..11]$
$S_6 = \mathtt{C\,B\,B} = T_2[6..8]$
$S_7 = \mathtt{A\,C\,B} = T_2[5..7]$
$S_8 = \mathtt{A\,B} \quad = T_2[3..4]$

5b

Konstruer suffikstræet for T_2. Lav nu en grådig algoritme der først finder S_1, derefter S_2, etc. For at finde S_1 starter vi med at læse T_1 fra ventre mod højre, startende i $T_1[1]$, og følger stien i suffikstræet for T_2 indtil vi har fundet det længste præfiks af et suffiks i T_2 som også er et præfiks af T_1. Denne streng er S_1. For at finde S_2 starter vi en ny søgning i suffikstræet startende i T_1 ved position $1 + |S_1|$. Tid $O(n_1 + n_2)$.