1. (20 points) Suppose that a network \((V,E)\) has a pair of nodes, \(u,v \in V\), such that 
\((u,v),(v,u) \in E\). That is, there is an edge from \(u\) to \(v\) and an edge from \(v\) to \(u\). Prove or give a counterexample to the following:

**Conjecture**: There must be a maximum flow \(f\) such that \(f(u,v) = 0\) or \(f(v,u) = 0\).
2. (20 points) Suppose that you are given a graph \( G = (V, E) \) and a function \( w : E \to \mathbb{R}^+ \) which associates a positive weight with every edge. We define the distance \( D : V \times V \to \mathbb{R}^+ \) between two vertices as the length of a shortest path between the vertices, and the diameter of \( G \) as the distance between two most distant vertices. That is, \( \text{diameter}(G) = \max_{v, w \in V} D[v, w] \). For example, the diameter of

![Graph Diagram]

is 18 because the distance between \( a \) and \( c \) is 18. Describe an algorithm to determine the diameter of an arbitrary graph \( G \) with worst-case time complexity in \( O(|V|^2) \).
3. (20 points) An instance of the THREESET problem is positive integers \(x_1, \ldots, x_n\), and an integer \(K\). An algorithm to solve the THREESET problem should return \text{true} if \(x_1, \ldots, x_n\) can be partitioned into multisets \(S_1, S_2, S_3\) such that \(\sum_{x \in S_1} x = \sum_{x \in S_2} x = \sum_{x \in S_3} x = K\) and \text{false} otherwise. That is, when presented with \(K=9\) and integers 1,2,2,4,4,5,7, the algorithm should return \text{true} because of the partition into multisets \(\{4,5\}, \{2,7\}\) and \(\{1,2,4\}\), and when presented with \(n=8, K=9\), and integers 1,2,2,4,4,5,7, the algorithm should return \text{false}. Your algorithm should work in time in \(O(nK)\) under the standard computational model.
4. (20 points) Suppose you are given an infinite supply of coins of denominations \( \{c_1, \ldots, c_n\} \).

For example, in the US (ignoring silver dollars) \( n=5 \) and \( c_1 = 1, c_2 = 5, c_3 = 10, c_4 = 25 \) and \( c_5=50 \). Design an algorithm to compute the number of ways to make change for \( a \$ \). For example, for input 

\[ n = 5, \{1, 5, 10, 25, 50\}, a=17 \]

the output would be 6 because of the solutions

\[(10, 5, 1, 1), (10, 1, 1, 1, 1, 1), (5, 5, 5, 1, 1), (5, 5, 1, 1, 1, 1, 1),
(5, 1, 1, 1, 1, 1, 1, 1, 1, 1), (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)\]
5. (20 points) A recursive definition of the binomial coefficients is that
\[
\binom{n}{k} = \begin{cases} 
1, & \text{if } k = 0 \text{ or } k = n \\
\binom{n-1}{k-1} + \binom{n-1}{k}, & \text{otherwise}
\end{cases}
\]
The corresponding recursive program is
\[
C(n,k) \begin{cases} 
\text{if } k = 0 \text{ or } k = n \\
\text{then return } 1 \\
\text{else return } C(n-1,k-1) + C(n-1,k)
\end{cases}
\]
\(a\) What is the time to execute \(C(n,k)\), as a function of \(n\) and \(k\)?

\(b\) Write a dynamic programming algorithm to compute \(C(n,k)\) with an execution time in \(\Theta(nk)\).
1. The CONJECTURE is true. For any flow $f$, the value of $|f|$ doesn’t change if we increase or decrease $f(u,v)$ and $f(v,u)$ by the same amount (while respecting capacity constraints). Let $f$ be any maximum flow. If $f(u,v) > 0$ and $f(v,u) > 0$, then we create a new flow $f^*(x,y) = \begin{cases} f(x,y), & \text{if } \{x,y\} \neq \{u,v\} \\ f(x,y) - \min\{f(u,v), f(v,u)\}, & \text{if } \{x,y\} = \{u,v\} \end{cases}$

That is, $f^*$ is the same as $f$ except through the edges $(u,v),(v,u)$, and in that case they are reduced by the same amount. And $|f| = |f^*|$.  

2. Apply Dijkstra’s algorithm $|V|$ times, once for each $v \in V$ as the source.

3. Given an instance of THREESET, you can call KNAPSACK with capacity $K$ such that the value and weight of object $i$ is $x_i$ for $1 \leq i \leq n$. Using the dynamic programming algorithm to solve KNAPSACK, if there is no packing of value $K$, then THREESET should return false. If there is a packing of value $K$, then remove these elements from the set of possible objects. Invoking the knapsack algorithm again, the remaining objects admit a packing of value $K$ if and only if the answer to THREESET is true.

4. For $0 \leq k \leq n$, $0 \leq x \leq a$, we let $f(k,x)$ denote the number of ways to make change for $x\epsilon$ using coins of denominations $\{c_i, \ldots, c_k\}$. We assume that $f(k,x)=0$ if $k<0$ or $x<0$. We note that we can use $0,1,\ldots,\lfloor x/c_i \rfloor$ coins of denomination $c_i$. If we use exactly $i$ coins of denomination $c_i$, then there are $f(k-1,x-ic_i)$ ways to make change for $(x-ic_i)\epsilon$ using coins of denominations $\{c_i,\ldots,c_{i+1}\}$.

$$f(k,x) = \sum_{0 \leq i \leq \lfloor x/c_i \rfloor} f(k-1,x-ic_i)$$

for $x \leftarrow 1$ to $a$ do

for $k \leftarrow 0$ to $n$ do

f(k,x) \leftarrow f(k-1,x) /* using 0 c_k */

for $i \leftarrow 1$ to $\lfloor x/c_i \rfloor$ do
\[
f(k, x) \leftarrow f(k, x) + f(k - 1, x - i) \quad \text{/* using } i \text{ */}
\]

\[
\text{return } f(n, a)
\]

5. \(a\) Because ultimately the recursion tree causes \(\binom{n}{k}\) \(1\)'s to be added together, computing \(C(n, k)\) makes \(\binom{n}{k}\) calls on \(C\), and has a time complexity \(\Theta\left(\binom{n}{k}\right)\).

\(b\) The values are stored in an array \(C[0..n, 0..k]\).

\[
\text{for } i \leftarrow 0 \text{ to } n \text{ do}
\]

\[
\text{for } j \leftarrow 0 \text{ to } \min(i, k) \text{ do}
\]

\[
\text{if } k = 0 \text{ or } k = n
\]

\[
\text{then } C[i, j] \leftarrow 1
\]

\[
\text{else } C[i, j] \leftarrow C[i - 1, j - 1] + C[i - 1, j]
\]