Övningsmästerprov 2: Lika vikt

Uppgift: Visa att beslutsproblemet "Lika vikt" är NP fullständigt.

Vad krävs för ett NP-fullständighetsbevis:
För att visa att ett problem Q är NP-fullständigt krävs det att problemet uppfyller 2 krav:

1. Det givna problemet Q ligger i NP, alltså Q $\in$ NP.
2. Att Q är NP-svårt, alltså att alla problem i NP kan reduceras till Q på polynomisk tid.

NP tillhörighet:
För att visa att problemet ligger i NP ska vi kunna verifiera alla ja-instanser som tillhör problemet på polynomisk tid, därför måste en verifieringsalgoritm som körs på polynomisk kunna skapas. En lösning av lika vikt kommer att bestå av två olika mängder som är en uppdelning av de *n* elementen i indatan *v1,....,vn*. Denna indelning kan representeras av en array med *n* platser som innehåller 0:or och 1:or som ska representera i vilken av de två delmängder som talet på den platsen ska vara i. Till exempel om det står en 0:a på index 2 ska element *v2* vara i den första av de två mängderna. Denna array kan man sedan iterera igenom på linjär tid för att kontrollera att summan av de två mängderna är lika.

```
verifiera(v[1,....,n], setnr[1,....,n])
        set1_sum, set2_sum ← 0
        for i ← 1 to n do
                if (setnr[i] == 0) do set1_sum += v[i]
                if (setnr[i] == 1) do set2_sum += v[i]
        if(set1_sum == set2_sum)
                return true
        else
                return false
```

Tidskomplexitet för verifieringsalgoritmen:
Denna algoritm går på polynomisk tid(linjär) tid, alltså *O(n)*, vilket man kan se i for loopen som endast itererar n gånger och endast utför konstant tids operationer. detta visar att lika vikt ligger i NP.

Lika vikt är NP-svårt:
För att visa att Lika vikts problemet är NP-svårt ska vi reducera ett NP-fullständigt problem till Lika vikt, detta problem är delmängdssumma. Delmängdssumma är ett NP-fullständigt problem som givet en lista med positiva heltal *P* och ett mål *K* försöker hitta en möjlig delmängd av *P* som bildar summan *K*.

Vi låter $s = \sum_{i=1}^{n} P_i$, alltså summan av alla tal i *P*. När vi sedan karpreducerar delmängdssumma till lika vikt ska den sökta uppdelningen av vikter vara i två delmängder som båda bildar summan $s/2$. Detta är om vi samtidigt låter talen i P representera vikterna som skickas som indata till lika vikts problemet. Om vårt K från delmängdssumma är exakt lika med $s/2$ kan man direkt använda sig av lika vikt eftersom detta är ekvivalent med att söka efter en delmängd vars summa är hälften av den totala summan av P, vilket i sin tur ger oss en partitionering av prylarna vars vikter är lika.

Men om $K \neq s/2$ får vi två fall som måste hanteras, och detta görs genom att man lägger till ett extra element, x, i vår viktmängd som är anpassad på ett sådant sätt att ifall man kan dela in vikterna i två lika grupper motsvarar det även att man kan hitta en delmängd i P vars summa är lika med K.

1. **$K < s/2$:**
   I detta fall krävs det att man lägger till en extra vikt i den hittade delmängden för att bilda en ekvivalent summa med den resterande mängden. Alltså måste den delen ha vikten $K+x$. Partitioneringen av vikterna i den nya mängden med x kommer att ha en vikt på $(x+s)/2$, vilket ger oss $K+x = (x+s)/2 \Rightarrow x = s - 2K$.
2. **$K > s/2$**
   I detta fall kommer den funna del mängden att ha en summa som är större än hälften av den totala summan delat på två, för att detta ska kunnas lösas av lika vikt krävs det då att man lägger till en vikt på den resterande mängden. Vi vill alltså att ena delen ska vikt K vilket ger $(s+x)/2 = K \Rightarrow x = 2K - s$.

Denna reduktion går på polynomisk eftersom att det endast krävs att man skapar en ny mängd *v[]* från P och i vissa fall lägger till ett extra element x, vilket som i värsta fall går på linjär tid med avseende på storleken av indatan P. Alltså är tidskomplexiteten *O(n)* för reduktionen.

Korrekthet:
För att visa att denna karpreduktion är korrekt krävs det att man uppfyller två krav: alla ja instanser från delmängdssumma problemet ska också bilda en ja instans i lika vikts problemets, på samma sätt ska alla nej instanser från delmängdssumma också motsvara nej instanser i lika vikt. För att visa att en ja instans i delmängdssumma transformeras till en ja instans i lika vikt behöver man visa att om det finns en delmängd i P var summa är lika med K så ska det också finnas en uppsättning av vikter i *v[]* som har samma vikt. Detta ska stämma för de tre fallen angivna i reduktionen, $K = s/2$, $K < s/2$ och $K > s/2$.
För att göra motsvarande bevis fast för nej instanser kan man istället gå det motsatta hållet och visa att för varje ja instans för lika vikt som skapats av reduktionen så är den motsvarande förekomsten av delmängdssumma även en ja instans. Det krävs även här att man visar för de tre fallen som i beviset innan.