|  |
| --- |
| Universitetet i Sørøst-Norge  Fakultet for Teknologi, Naturvitenskap og Maritime fag  Institutt for Mikrosystemer  –  Obligatorisk oppgave  Ola Slålien Thoen **04/04/2019**  **Oblig 2 vår DA-ALG1000** |
|  |
|  |
|  |
|  |

Innholdsfortegnelse

[1 Innledning 3](#_Toc5359426)

[1.1 Oppgavetekst 3](#_Toc5359427)

[1.2 Løsning av oppgaven 4](#_Toc5359428)

[2 Metoder 4](#_Toc5359429)

[2.1 UML representasjon av klassen 4](#_Toc5359430)

[2.2 findCost 4](#_Toc5359431)

[2.3 Main 5](#_Toc5359432)

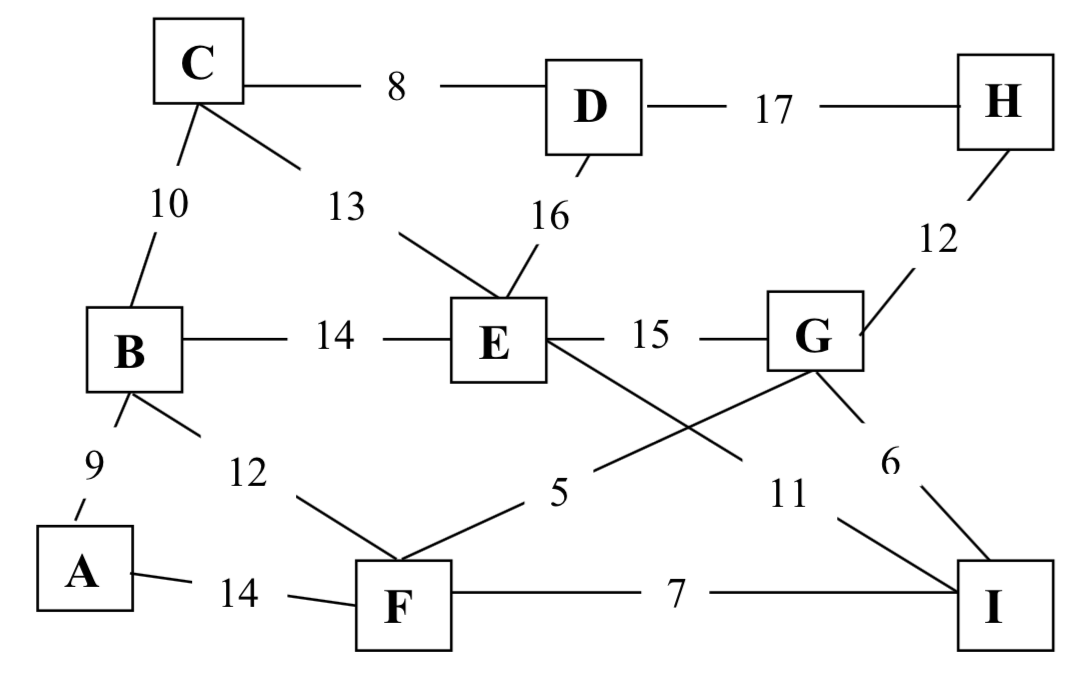
[2.3.1 Spesialtilfeller 5](#_Toc5359433)

[3 Konklusjon 5](#_Toc5359434)

[4 Vedlegg 5](#_Toc5359435)

# Innledning

## Oppgavetekst

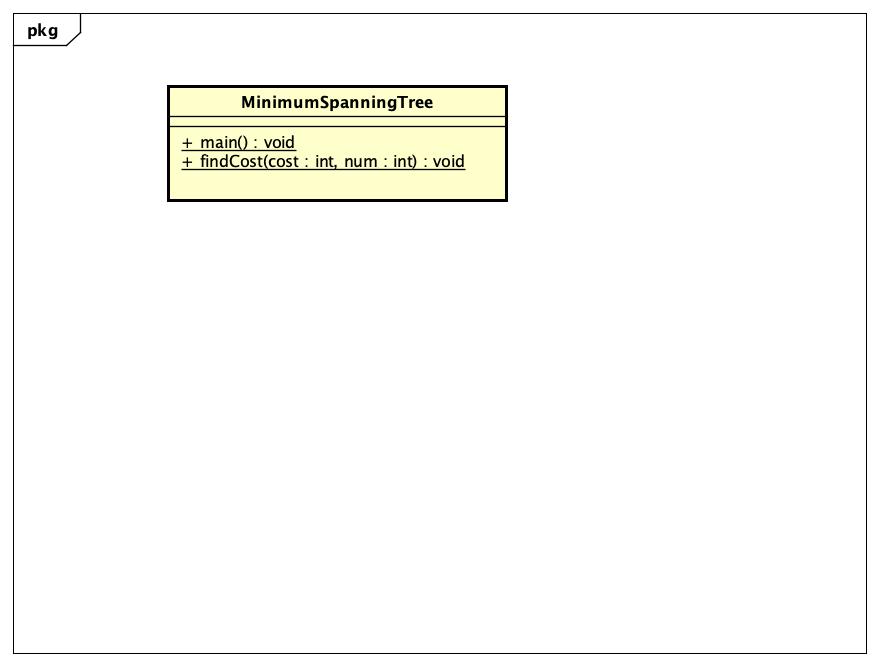
I følgende figur tenker vi oss at nodene A, B, C, D, E, F, G, H og I indikerer hus. Kantene mellom husene indikerer at det kan trekkes en fiberkabel mellom 2 hus til en kostnad av det tallet som er oppgitt ved den tilhørende kanten. Prisen er oppgitt i antall 1000 kr. Programmér en løsning slik at alle hus er tilknyttet og at total kostnad er minst mulig. Man skal kunne finne samme løsning uansett hvilken node/hus man starter i. Programmet skal først skrive ut total kostnad og deretter hvilke hus som parvis velges underveis. Programmet skal ha god informasjon til/med bruker underveis. Dersom programmet oppdager feil, skal dette informeres bruker på en god måte. Ønsker ikke å se en løsning som kan spores til å ha blitt funnet på Internett. Oppgaven leveres som én fil (gjerne .zip) og rapport skal være med (i samme fila!). Oppgaven skal løses vha. en tabell- representasjon av grafen. Dersom dere vil løse den også på en annen måte, er det mulig ved å levere 2 forskjellige løsninger. Lykke til!

## Løsning av oppgaven

Ut ifra følgende gitte oppgave ble det brukt en graf representert som en todimensjonal tabell i løsningen. For de plasseringene det ikke er forbindelse mellom to hus er verdien satt til 32000 for å ha et høyt tall som gjør jobben lettere å finne minste verdien mellom hvert av husa. Hele oppgaven ligger i en klasse og det er brukt én funksjonsmetode for å løse oppgaven.

# Metoder

## UML representasjon av klassen



## findCost

findCost metoden har tabellen med husene og verdiene mellom og tallet hentet fra bruker som parameter. For å vise hvilke hus som er valgt og ikke valgt gjennom metoden brukers det en endimensjonal tabell med boolean verdier for hver av husene. «True» vil si at huset er besøkt, mens «false» vil si at det ikke er besøkt enda. Til å starte med brukes tallet hentet fra brukeren i main til å sette det valgte huset lik true. Dermed starter programmet og sjekke billigste reiserute derifra. Oppgaven er løst ved å gå igjennom tabellen ved hjelp av tre nestede for-løkker. De to innerste går igjennom alle radene og kolonnene og finner den billigste veien mellom to hus, mens den ytterste får dette til å skje 8 ganger for å finne alle forbindelsene. I den innerste for-løkka if-testes det; hvis huset er satt til true i boolean-tabellen og det gjeldende huset ikke er besøkt/kolonnens index er «false», samtidig som den gjeldende verdien er mindre enn en variabel min blir min satt til denne verdien. I tillegg blir to variabler satt til de gjeldende koordinatene for å huske plasseringen til den billigste koblingen mellom to hus. Etter de to innerste er traversert i den ytterste for-løkka blir den billigste verdien for hver kobling lagt til i en variabel for total reisekostnad. Det gjeldende huset blir satt til «true» i boolean-tabellen, variabelen min blir tilbakestilt til 32000 og de valgte husene blir lagt til i en ArrayList. Deretter utenfor for-løkkene skrives den totale reisekostnaden ut fra metoden, og en siste for-løkke går igjennom de valgte husene i ArrayListen og skriver ut hvilke reiseruter som er valgt i rekkefølge.

## Main

main-metoden inneholder den todimensjonale tabellen med alle forbindelsene og kostnadene mellom dem. Bruker blir spurt om hvilket hus han vil starte i representert med tall fra 0 – 8 for hver av de 9 husene. Deretter blir tabellen og tallet valgt av bruker satt som parameter inn i metoden som da skriver ut total kostnad for reisen og skriver ut hvilke hus som parvis blir valgt.

### Spesialtilfeller

Programmet er testet for at bruker-input ikke er gyldig. I tilfelle bruker velger et tall annet enn 0 – 8, vil han bli bedt om å legge inn et nytt tall mellom 0 – 8. Dette vil gå i løkke frem til bruker taster et gyldig tall.

# Konklusjon

Programmet spør bruker om hvor han vil starte og gir en output på den totale kostnaden og hvilke hus som er valgt i hvilken rekkefølge, som var det oppgaven blant annet spurte om. Man får samme totale kostnad uansett hvilket hus man velger å begynne i. Bruker får god beskjed om at han har valgt et tall som ikke er gyldig og må dermed taste inn et nytt. En forbedring kunne vært å returnere noe fra metoden, for å øke funksjonaliteten til eventuelt videre bruk.

# Vedlegg

Kildekoden ligger vedlagt i den leverte zip-fila.