

Institutt for datateknologi og informatikk

Eksamensoppgave i IDATT2101 Algoritmer og datastrukturer

Faglig kontakt under eksamen: Helge Haffing

Tlf.: 924 386 56

Eksamensdato: 24. november 2020

Eksamenstid (fra-til): 09:00–13:00

Hjelpemiddelkode/Tillatte hjelpemidler: Alle

Annen informasjon:

Målform/språk: bokmål

Antall sider (uten forside): 2

Antall sider vedlegg: 1

Informasjon om trykking av eksamensoppgave			
Originalen er:			
1-sidig	<input type="checkbox"/>	2-sidig	<input checked="" type="checkbox"/>
sort/hvit	<input type="checkbox"/>	farger	<input checked="" type="checkbox"/>
skal ha flervalgskjema		<input type="checkbox"/>	

Kontrollert av

.....
Dato Sign

Oppgave 1

10%

Du har fått et kandidatnr for å skrive på oppgaven og vedlegg. Gang dette kandidatnummeret med 47. (Kalkulator kan brukes.) Tallet du får, blir sifre for denne oppgaven.

- a) Sett sifrene inn i en max-heap. Sett dem inn ett om gangen, i den rekkefølgen de har i tallet. Tegn opp heapen en gang for hvert tall du setter inn.
- b) Sett de samme sifrene inn i et binært søketre. Sett dem inn i den rekkefølgen de har i tallet, og tegn opp søketreet en gang for hvert tall du setter inn.

Oppgave 2

20%

Analyser de følgende programmene og finn kjøretiden. Regn med at alle parametre er større enn eller lik 0. Bruk Θ om mulig, ellers O og Ω .

```
public void oppg_a(int m, int [][] tab) {  
    for (int i = 1; i < m; ++i) {  
        for (int j = m; j > 0; --j) tab[i][j] = i*j;  
    }  
}
```

```
public int oppg_b(int n, int p) {  
    int res;  
    if (n >= p) return -1;  
    for (int i = 0; i < p; i+=n) res += i;  
    return res;  
}
```

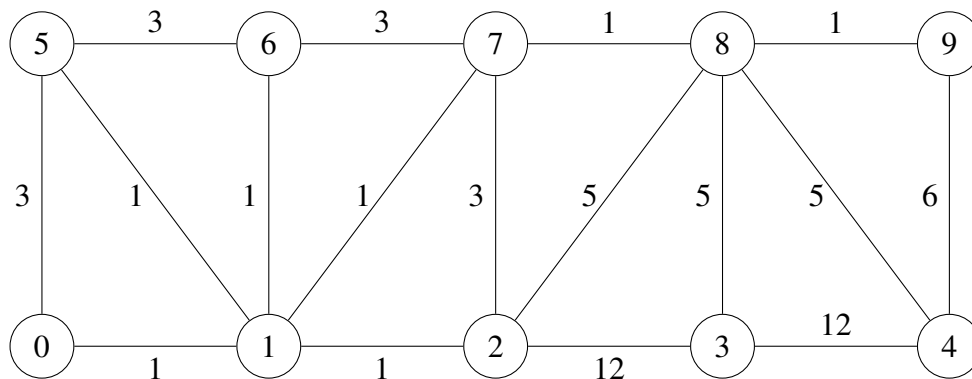
```
public float oppg_c(int n, float[] tab) {  
    float sum = 0.0;  
    if (n <= 0) return sum;  
    for (int i=0; i<n; ++i) sum += tab[i];  
    return sum + oppg_c(n/2, tab);  
}
```

```
public float oppg_d(int m, float[] tab) {  
    float sum = 0.0;  
    if (m <= 0) return sum;  
    for (int i=0; i<m; ++i) {  
        for (int j=i; j<m; ++j) {  
            sum += tab[i] * tab[j];  
        }  
    }  
    sum += oppg_d(m/2, tab);  
    return sum + oppg_d(m/2, tab);  
}
```

Oppgave 3

30%

Gitt denne grafen:

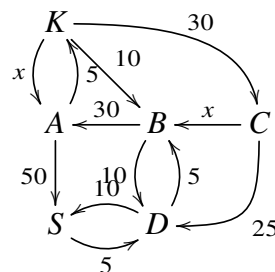


- Velg noden med samme nr. som siste siffer i kandidatnummeret ditt som startnode. Finn korteste vei fra startnode til alle andre noder i grafen, og tegn korteste-vei treet.
- Forklar hvorfor Dijkstras algoritme ikke fungerer korrekt på grafer som har kanter med negative vekter.

Oppgave 4

25%

Bruk denne grafen. Bruk siste siffer i kandidatnummeret ditt, som vekt på de to kantene merket x .



- Finn og tegn et minimalt spennetre med vekt, eller forklar hvorfor det ikke er mulig. (På denne deloppgaven ser du bort fra pilene, så det blir en urettet graf.)
- Finn maksimal flyt fra K til S ved hjelp av flytøkende veier. Skriv opp hvilke flytøkende veier du bruker, og hvor mye du øker flyten langs hver vei. I denne deloppgaven er grafen rettet.

Oppgave 5

15%

- Forklar hva det vil si, at et program eller problem er i kompleksitetsklassen **P**. Gi eksempel på et slikt problem.
- Forklar hva det vil si, at et problem er i kompleksitetsklassen **NP**.
- Kåre vil lagre postnumre i en hashtabell. Tabellen har størrelse 100. Mange postnumre slutter på 0. Vil det være lurt å bruke en hashfunksjon basert på restdivisjon? Hvorfor, eller hvorfor ikke?