

# Projektuppgift i DV1490/DV1538

## Inledning

OR (Operations Research<sup>1</sup>) är ett vetenskapligt förhållningssätt att lösa komplexa problem. Exempel på användningsområde är beslutsstödst teori, schemaläggning, logistikproblem och många fler. Läs mer på soaf.se vid intresse. Detta projekt handlar om schemaläggning av kirurgiska operationer, och hur man kan göra det på ett så resurseffektivt sätt som möjligt.

Kirurgiska operationer kan delas in i olika specialiteter så som ortopedi, gynekologi, generell kirurgi mm. Vidare kan varje specialitet indelas i subspecialiteter, ex kan ortopedi indelas i höft, rygg, axel mm. I detta projekt ska ni studera och finna en lösning på planering av ortopediska operationer. Vi antar nu att det finns ett antal ortopediska operationer i en lista som ska schemaläggas till ett antal operationssalar. Varje operationssal kan schemaläggas från ett bestämt klockslag och fram till ett bestämt klockslag, dessa klockslag kan variera mellan olika operationssalar. Utöver dessa klockslag är det inte längre möjligt att schemalägga någon operation. Varje operation har ett estimat över tidsåtgång, angivet i minuter. Målet är att schemalägga operationssalarna med så många operationer som möjligt av de som finns i listan, utan att överskrida klockslaget för stängning av respektive operationssal. Det är heller inte fördelaktigt att låta operationssalar stå outnyttjade.

## Förberedelse

Detta problem beskrivs i litteraturen som *the Bin packing problem*, se kap 10.1.3 i kursboken. En genomgång av projektuppgiften görs av Marie Netz på eftermiddagen måndagen den 7 maj 2018, se aktuellt kursschema.

## Implementation (med uppgift 1 och 2)

Projektuppgiften ska göras individuellt eller i grupp om max två studenter. Varje student ska dock bidra väsentligt till utvecklingen och implementeringen av algoritmen samt kunna redogöra för hela projektet. Den skriftliga redovisningen ska göras individuellt.

Du har tillgång till en textfil med en lista över de operationer som ska schemaläggas. Listan är konstruerad med id, därefter subspecialitet följt av ett estimat över tidsåtgång enligt följande princip:

1, arm, 35  
2, fot, 120  
3, rygg, 250  
4, hand, 60  
osv.

Tips: Läs radvis med hjälp av `getline()`; Exempel på text att undersöka "Mrs#Smith#1938"  
`int pos = line.find("#"); //ger 3`  
`string title = line.substr(0,pos); //ger "Mrs"`  
`string restOfLine = line.substr(pos+1); //ger "Smith#1938"`  
osv.

---

<sup>1</sup> <https://soaf.se/vad-ar-operationsanalys/>

Dessa operationer ska på ett intelligent sätt schemaläggas (fördelas) till tillgängliga operationssalar enligt ovan beskrivna förutsättningar. Det innebär att du ska konstruera ett program som innehåller en egen implementation av en relevant algoritm och nödvändiga datastrukturer som krävs för att läsa in en indatafil med information om operationerna och schemalägga dessa. Du får alltså inte importera och använda färdiga bibliotek för att utföra schemaläggningen. Ditt program ska även kunna skriva ut minst tre relevanta och tillåtna lösningar på schemalägningsproblemet, dvs. specifikation av vilka operationer som schemalagts i respektive operationssal, förslagsvis:

Lösning 1:

Operationssal 1: 1,4,6,7

Operationssal 2: 2,8,10

Operationssal 3: 3,5,9

Lösning 2:

osv.

**Uppgift 1.** Tre stycken operationssalar ska schemaläggas med ett visst antal operationer. Operationerna finns angivna i textfilerna "Operationer\_1a.txt" och "Operationer\_1b.txt". Varje operationssal är tillgänglig (dvs. kan schemaläggas) från kl. 8.00 fram till kl. 19.00. Din algoritm ska kunna hantera mer generella problem än det specifika fallet med schemaläggning av ett fördefinierat antal operationer i tre specifika operationssalar, dvs. algoritmen ska även kunna hantera en annan indata-fil innehållande andra operationer och ett annat antal operationssalar.

Frågor du ska fundera på och besvara i rapporten: *Vilka eventuella operationer kunde inte schemaläggas, och hur prioriterade din algoritm mellan olika operationer? Hur stor andel av tillgänglig tid var resp. operationssal schemalagd? Kan operationssalarna utnyttjas på ett bättre sätt än vad din algoritm föreslår?*

**Uppgift 2.** Samma operationssalar ska schemaläggas, fast för två veckodagar i följd (exempelvis en måndag och efterföljande tisdag). Tillgänglig tid för operationssal 1 är kl. 8.00 - 19.00, operationssal 2 kl. 8.00 - 22.00, och för operationssal 3 kl. 8.00 - 17.00, i övrigt gäller samma restriktioner som tidigare. Textfilen "Operationer\_2.txt" ska användas nu. Jämför beräkningstiden som krävs mellan uppgift 1 och 2.

Tips: # include <ctime>

clock\_t start;

clock\_t end;

start = clock();

//er algoritm

end = clock();

Frågor du ska fundera på och diskutera i rapporten: *Jämför den beräkningstid som krävdes för att finna en tillräckligt bra lösning i uppgift 1 med den beräkningstid som krävdes för att finna en tillräckligt bra lösning i uppgift 2. Hur effektiv tror du att din algoritm är om antalet operationssalar och operationer som ska schemaläggas ökar ännu mer? Fungerar den algoritm du utvecklat och implementerat för att generera en lösning inom rimlig tid?*

## Skriftlig rapport

Rapporten ska ha strukturen av en teknisk rapport. Länk till stöd för hur man skriver en teknisk rapport finner du här:

[http://skrivguiden.se/media/2017/12/Skriva tekniska rapporter 2017 dec.pdf](http://skrivguiden.se/media/2017/12/Skriva_tekniska_rapporter_2017_dec.pdf). Rapporten kan vara på engelska eller svenska, men säkerställ att den är korrekturläst och ni har använt relevant program för stavnings- och grammatikkontroll innan ni lämnar in rapporten. Utöver ditt namn, studentakronym och datum ska rapporten innehålla följande:

- En beskrivning av vem man eventuellt samarbetet med (namn och studentakronym).
- En beskrivning av den fungerande algoritm du utvecklat och implementerat (i form av pseudokod) och de datastrukturer du använt. Motivera utformningen av algoritmen och de datastrukturer du valt att använda.
- En bilaga i rapporten där du klistrar in din/er lösning till uppgift 1 och 2 (dvs. programmets utskrifter)
- En redogörelse och diskussion kring hur du ser på vad som är rimlig tid för att lösa ett problem av denna karaktär, samt skalbarhet när problemet växer, dvs. då antalet operationssalar och operationer ökar. Se stödfrågor under respektive uppgift 1 och 2. Förslag ges nedan på hur du kan redovisa hur lång tid det tar att generera minst tre olika tillåtna, relevanta lösningar för samma problem:



- En redogörelse och diskussion kring hur du ser på att värdera de lösningar som algoritmen föreslår. Värderas exempelvis 20 minuters icke-schemalagd tid i en operationssal som 10 minuters icke-schemalagd tid i två operationssalar? Hur skulle din algoritm kunna hantera prioriteringar mellan operationer, dvs. vissa operationer är kanske viktigare än andra?

## Inlämning

Du ska lämna in den skriftliga rapporten enligt ovan (pdf- eller word-format). Du ska även lämna in de h-filer och cpp-filer som du konstruerat och som krävs för att köra ditt program och lösa uppgift 1 och 2 i projektet. Lämna INTE in en zip- eller rar-fil, dvs. alla filer ska lämnas in som separata filer via Canvas.

## Muntlig redovisning (gäller endast DV1538)

Under den muntliga redovisningen ska ditt program kunna köras och fungera. Du kommer att bli ombedd att besvara en del frågor som berör problemet och hur du/ni har valt att lösa uppgifterna. Redovisning kommer ske inför hela gruppen under schemalagda pass fredagen den 25 maj. Det är därför bra att förbereda en kort ppt-presentation. Om ni är två som arbetar i grupp ska båda vara närvarande vid den muntliga redovisningen. Ytterligare instruktioner följer.

Lycka till!