

# Kurs-PM HI1024 8,0 hp HT2020 TIDAA

Kursplan: <http://www.kth.se/student/kurser/kurs/HI1024/>

Kursmaterial i Canvas: <https://kth.instructure.com/courses/20458>

Kursutveckling och tidigare års material inklusive tentor: <https://www.kth.se/social/course/HI1024/>

Kurslitteratur:

K N King, C Programming A Modern Approach, 2 ed, WW Norton & Co, ISBN: 9780393979503

Frivillig extralitteratur främst som stöd om du har svårt med engelskan:

Jan Skansholm, C från början, Studentlitteratur AB, ISBN: 9789144114583

Examinator: Nicklas Brandefelt, [bfelt@kth.se](mailto:bfelt@kth.se), 087909738

Lärare: Nicklas Brandefelt

## *Genomförande av kursen*

Det absolut första du måste göra på kursen är att se till att du har en fungerande programmeringsmiljö på din dator. Förhoppningsvis har du gjort detta på introkursen (HF1010). För att denna kurs ska bli kul och gå bra är det oerhört viktigt att du programmerar mycket under hela kursen. Totalt behöver du antagligen lägga ca 50h/vecka på dina studier vid kth. Bäst är att göra detta mån-fre och sedan vara helt ledig fredag kväll och lördag för att sedan söndag eftermiddag se över om man behöver göra lite extra inför veckan som kommer.

Till varje föreläsning finns läsanvisningar till boken och programmeringsuppgifter i planeringen. Det är viktigt att ni läser och programmerar så att ni ligger i fas. Ge inte upp om det går trögt i början. Det tar tid att lära sig programmera men det är oerhört kul när man kan. Efter varje föreläsning är det 2h övning med övningsassistent (lärare är med via zoom). Utnyttja dessa till max. Sitt helst i skolan och programmera även övrig tid så att ni kan hjälpa och sporra varandra.

Kom ihåg att de tre föreläsningarna från intro-kursen ingår som självstudie i kursen. Om du inte deltog i intro-kursen behöver du läsa igenom dessa och ställa frågor om något är oklart.

## **Examination**

### **LAB1 (2hp)**

Momentet består av tre laborationer. För laboration 1 och laboration 2 finns två redovisningstillfälle till vardera. Blir man godkänd på dessa behöver man inte göra delar av laboration 3. För laboration tre finns två redovisningstillfällen under ordinarie kurs. Blir man inte godkänd vid något av dessa tillfällen måste man redovisa hela lab 3 vid omtentatillfället i p2 eller tentatillfället i p3 (man slipper alltså inga delar även om man gjort lab 1 och/eller lab 2). När laboration 3 är godkänd är man godkänd på momentet (betyget PASS). Blir man inte klar med momentet under läsåret måste hela momentet göras om nästa läsår. Tider för redovisningar meddelas och bokas via canvas (gäller även p2 och p3).

### *Plagiering*

Ni får/bör hjälpa varandra när ni lär er programmera. Diskutera och fråga varandra hur man löser olika problem. Ni kan också hjälpa varandra att hitta buggar när ni arbetar med övningsuppgifter men inte när ni

arbetar med laborationerna. Då ska ni arbeta helt självständigt. Diskutera endast problem mera generellt med varandra och undvik att ens titta på varandras kod. Behöver ni kod för diskussionen skriv separat exempelkod istället. Ni får aldrig föra över kod eller skriva in kod åt varandra när ni arbetar med examinationsmomenten (det är inte en bra ide annars heller). Du måste förstå vad du programmerar – vad varje rad gör! -det är fusk att ta över andras kod på examinationsmoment. Att förstå vad varje rad gör är avgörande för att du ska lära dig att programmera och därmed klara tentamen men framförallt för att du ska bli en bra ingenjör!

Det har förekommit att studenter har kopierat delar av kod men sedan bytt ut variabelnamn, kastat om vissa rader och lagt in nya kommentarer. När detta har upptäckts har studenterna blivit avstängda av disciplinnämnden. Senast blev även den vars kod blivit plagierad avstängd.

### **Redovisning**

Senast 15.00 dagen innan redovisning av laboration skall man via Canvas boka en tid för redovisning. Senast **20 minuter innan sin redovisningstid** skall man ha loggat in i rätt zoomrum med datorn på och programmet körbart redo att redovisa. För laboration 3 gäller att man dessutom senast 15.00 dagen innan redovisning skall ha skickat in programmet via <https://maceo.sth.kth.se> för avancerad plagiatkontroll. **Observera att man inte får boka tid för lab 3 innan man skickat in den kod man ska redovisa!** Laboration 1 och 2 redovisas två och två. Vänligen respektera dessa något omständliga regler. De är till för att vi ska hinna med att ta emot så många redovisningar och ändå förhoppningsvis hinna ge feedback.

För att bli godkänd på en laboration krävs att programmet uppfyller specifikationen, att programmet är välskrivet och att du kan förklara hur programmet fungerar.

### **Tentamen**

Det kommer att finnas två tillfällen att skriva tentor: tentaperioden i p1 och omtentaperioden i p2. Förutom dessa tillfällen har Medicinsk teknik tenta och omtenta som man kan skriva men dessa riskerar då att krocka med andra tentor.

### **TENA(3 hp)**

TENA är en teoretisk tentamen som denna gång antagligen kommer att göras digitalt hemifrån. Inför TENA finns det ett antal instuderingsfrågor. Dessa är tänkta att besvaras med hjälp av boken. Gör gärna dessa kontinuerligt under kursen efter det att du har gjort programmeringsuppgifterna. Själva tentamen kommer bestå av ett antal teorifrågor och några kodexempel. Teorifrågorna kommer att vara på boken (liknande instuderingsfrågorna), på föreläsningarna och på föreläsningarna från introkursen (HF1010). Kodexempeluppgifterna kan vara att man ska hitta fel i kod eller skriva kod eller på annat sätt visa programmeringskunskaper.

### **TEN2 (3 hp)**

TEN2 är en salsskrivning där ni på er egen dator bootar upp en speciell tentamensmiljö och därmed inte kommer att ha tillgång till internet. **Man måste alltså ta med sig sin laptop till skrivningen.** Man får ta med sig sitt exemplar av kursboken: K. N. King, "C Programming A Modern Approach" och även Jan Skansholm, "C från början". Dessa måste vara helt rena utan anteckningar eller understrykningar! Inget övrigt får medtagas (förutom penna). Absolut inga anteckningar. Du kommer att skriva tentan på din dator men bootad från en usb-sticka så att du inte kan nå internet. I denna bootning kommer Visual Studio Code, CodeLite och CodeBlocks att vara installerad så se till att du behärskar någon av dessa även om du använder en annan programmeringsmiljö i vanliga fall.

### **Bonus**

Till de flesta övningstillfällen finns det uppgifter som man kan redovisa i klassrummet. Om du gör, redovisar och blir godkänd vid minst 8 tillfällen behöver du inte göra första uppgiften på TEN2. Den räknas då som full poäng. Observera att varje uppgift måste redovisas vid rätt övningstillfälle. Det

finns marginal för att missa några tillfällen om du skulle vara sjuk. Obs att du inte får gå till skolan om du är sjuk. Om du är sjuk men frisk nog att redovisa kontakta ansvarig lärare (Nicklas) före redovisningstillfället börjar. För mer information om hur redovisningarna går till läs aktuell information på canvas.

## Lärandemål

Efter kursen ska studenten kunna:

1. Utifrån en given avgränsad problemställning skapa ett fungerande program där koden är strukturerad, lätt att läsa och vid behov uppdelad i funktioner.
2. Analysera en större programmeringsuppgift och till denna skapa ett väl fungerande program där koden är bra strukturerad, lätt att läsa och uppdelad i funktioner på ett strukturerat sätt.
3. Använda externa filer för datalagring.
4. Läsa och förstå samt felsöka kod.
5. Känna till några grundläggande begrepp inom datakunskap relevant för programmering.
6. Förstå olika tekniker för problemlösning.

## Betygskriterier

Lärandemål 2-5 examineras med betygsskalan P,F. Lärandemål 1 examineras med betygsskalan A, B, C, D, E, FX, F:

E - Utifrån en given avgränsad enklare problemställning på väl tilltagen tid skapa ett fungerande program där koden är strukturerad, lätt att läsa och vid behov uppdelad i funktioner.

C – Utifrån en given avgränsad problemställning på väl tilltagen tid skapa ett väl fungerande program där koden är väl strukturerad, lätt att läsa och vid behov uppdelad i funktioner på ett strukturerat sätt.

A – Utifrån en given avgränsad något svårare problemställning som kräver ett visst mått av kreativitet på en något begränsad tid skapa ett väl fungerande program där koden är väl strukturerad, lätt att läsa och vid behov uppdelad i funktioner på ett strukturerat sätt.

Betyget Fx ges om man är mycket nära att uppnå betyget E. Betyget D ges om man delvis uppnår kraven för betyget C. Betyget B ges om man delvis uppnår kraven för betyget A.

TEN1 examinerar främst mål 1 och 4-6 med betyget P/F. LAB1 examinerar främst mål 1, 2 och 4 med betyg P/F. TEN2 examinerar främst mål 1 med betyget A-F. TEN2 kommer att ha två delar: en C-del och en A-del. På C-delen kan man beroende på resultat få betyget C, D, E, Fx eller F. A-delen bedöms endast om man visat att man når upp till betyget C på C-delen och bedöms då med betyget A, B eller C.

För godkänd kurs krävs godkänt på samtliga moment. Slutbetyget grundas på momentet TEN2 med betygsskalan A-F.