

Antagningsprov i programmering till Mattekollo 2026

Mattekollo – en rolig utmaning!

Mattekollo är ett läger för dig som gillar matematik och programmering! För att alla deltagare ska få ut så mycket som möjligt av lägret, har vi ett antagningsprov med kluriga problem. Vi ser fram emot att ta emot din ansökan!

För att vi ska kunna bedöma din ansökan på rätt sätt är det viktigt att du löser uppgifterna själv, utan hjälp från föräldrar, kompisar, syskon osv. Notera att det inte heller är tillåtet att använda AI-verktyg som t.ex. ChatGPT eller Claude Code, och det är heller inte tillåtet att googla efter lösningar till problemen.* Det handlar inte om att vara bäst – vi vill bara att du visar vad just du kan! Om vi upptäcker att någon fått hjälp med uppgifterna eller samarbetat med andra kan vi tyvärr inte erbjuda den en plats på kollot.

* Däremot är det okej att söka upp dokumentation eller begrepp du inte känner till, exempelvis om du inte vet vad *tidskomplexitet* är eller har glömt hur man läser in indata i programmeringsspråket du använder. Om du är osäker på reglerna, så kan du fråga oss på antagningsprov@mattekollo.se.

Hur antagningen fungerar

- Vi tar in ungefär **11 elever per årskurs** (åk 6, 7, 8, 9, gy1, gy2) baserat på resultaten från antagningsproven (i matematik och programmering).
- **Programmeringsprofilen:** Ca 15-25 % av deltagarna antas via programmeringsprovet.
- **Direktkvalificering:** Om du placerat dig topp 15 i årets PO-final får du en garanterad plats på programmeringsprofilen och behöver inte skicka in något prov.

Hur du skickar in dina lösningar

Skicka dina lösningar till antagningsprov@mattekollo.se senast den **12 april 2026**.

Vad du ska skicka in för varje uppgift:

- **Ett körbart program** i valfritt programmeringsspråk. Programmet kan bestå av en eller flera filer.
- **En kort förklarande text** där du beskriver:
 - Hur din lösning fungerar.
 - En förklaring av vilken tidskomplexitet programmet har.
 - Potentiella förbättringar som går att göra, men som du inte lyckades implementera själv.

Så här namnger och skickar du filer

- Bifoga alla lösningar samtidigt i ett och samma e-postmeddelande.
- Använd tydliga filnamn så att vi enkelt kan se vilken uppgift varje fil tillhör.
- Om din e-postklient blockerar scriptfiler som bilagor, döp om filen till `.txt` så brukar det fungera.
- Filstorleken får inte överstiga **30 MB**.

Mejlets ämnesrad

Använd följande format för ämnesraden i ditt mejl:
Programmeringsprov [Förnamn Efternamn] [Årskurs]

Exempel:

- Programmeringsprov Anna Svensson åk 8
- Programmeringsprov Abraham Swift gymnasiet åk 2

Om du behöver skicka in en ny version av dina lösningar – inga problem! Skicka ett nytt mejl med samma ämnesrad och bifoga alla dina lösningar igen. Vi bedömer alltid den senaste versionen du skickar in.

Tips för att öka dina chanser

- Förklara tydligt hur koden fungerar! Ju bättre förklaringar, desto bättre bedömning.
- Beskriv din idé även om du inte löser uppgiften helt. En genomtänkt lösningsstrategi kan ge delpoäng.
- Du behöver inte lösa alla uppgifter! Gör ditt bästa och visa hur du tänker. Lös så mycket du klarar av.

När får du besked?

Antagningsbesked skickas ut senast den 1 maj. Reservplatser meddelas i slutet av maj.

Glöm inte att även anmäla dig på mattekollo.se senast den 12 april!

Om du inte blir antagen i år – misströsta inte! Vi hoppas att du söker igen nästa år. Oavsett vad, fortsätt ha kul med programmering!



1. Brädspel

Många på mattekollo gillar att spela brädspel. Programmeringsledarna funderar därför på att köpa 5 nya brädspel. Tillsammans har de samlat in en lista av populära brädspel och hur många stjärnor de har fått i betyg (1-5). Vilka 5 brädspel har bäst betyg? Skriv ut deras namn och betyg.

Input

- Ett heltal n , antalet brädspel som ledarna har att välja mellan.
- n stycken rader. Varje rad innehåller namnet på ett brädspel, och ett betyg för det spelet. Betyget är ett flyttal som kan vara mellan 1 och 5.

Output

5 stycken rader med de brädspel som ledarna borde köpa. För varje brädspel, skriv ut både namn och betyg.

Exempelinput

```
6
Avalon 4.8
Crazy Time 5
Fia med knuff 1.5
Schack 3.8
Ricochet robots 4.157
Coup 3.2
```

Exempeloutput

```
Crazy Time 5
Avalon 4.8
Ricochet robots 4.157
Schack 3.8
Coup 3.2
```

2. Föreläsningseentusiasten

Under ett läger finns det n stycken schemalagda föreläsningar. Som deltagare vill du såklart gå på så många föreläsningar som möjligt. Givet start- och sluttider för alla föreläsningar, hur många föreläsningar kan du gå på som mest? Du kan inte gå på två föreläsningar som hålls samtidigt. Men ifall en föreläsning börjar samma minut som en annan slutar, så kan du gå på båda dem om du vill.

Input

- Ett heltal n , antalet föreläsningar som är schemalagda.
- n stycken rader med start- och sluttider. Alla tider ges på formatet hh:mm.

Output

Ett heltal, antalet föreläsningar du som mest kan gå på.

Exempelinput

```
3
08:15 08:45
08:10 08:50
08:45 09:00
```

Exempeloutput

```
2
```

3. Föreläsningseentusiasten 2

Innan du går på alla föreläsningar så får du plötsligt en viktig insikt: du lär dig inte lika mycket på alla föreläsningar. Därför har du gått runt till alla ledare och frågat vad de kommer handla om. På så sätt har du lyckats lista ut hur mycket värde du får ut av varje föreläsning. Du vill få så stor summa av föreläsningvärden som möjligt. Hur stor summa kan du få på en dag?

Input

- Ett heltal n , antalet föreläsningar som är schemalagda.
- n stycken rader med föreläsningar. Varje rad består utav en starttid, sluttid och ett heltal som representerar hur mycket värde du får ut av den föreläsningen.

Output

Ett heltal med det största totala värdet du kan få ut av ett smart urval av föreläsningar.

Exempelinput

```
3
08:15 08:45 5
08:10 08:50 10
08:45 09:00 4
```

Exempeloutput

```
10
```

4. Skattjakten

En tradition på mattekollo är den så kallade skattjakten. Det fungerar lite som orientering, fast vid varje kontroll så måste någon slags utmaning klaras av. I början av skattjakten får ni en karta, och varje kontroll är märkt med en bokstav A, B, C osv. Du och ditt lag har kommit fram till att ta alla kontroller i bokstavsordning. Om ni alltid tar den kortaste vägen för att komma till en kontroll, hur långt måste ni gå då?

Kartan är given som ett rutnät. Ni börjar längst upp till väster och ska sedan ta er till stället med A, sedan B, sedan C osv. Vissa delar av kartan har brännässlor som är markerade med symbolen #. Dessa rutor får ni inte gå igenom. För resterande rutor får ni gå hur ni vill, så länge ni bara förflyttar er i en av de fyra riktningarna: vänster, höger, uppåt, neråt.

Input

- En rad med två heltal h , b : höjden och bredden på kartan.
- h rader. Varje rad består utav b stycken tecken som antingen är '#', '.', eller en bokstav A – Z.

Output

Ett heltal: antalet steg som ni måste gå.

Exempelinput

```
. . ## . B . # . . #  
# . . . # . . . . # C  
# . # . A # . # . . .  
# D . . . . # . . . .
```

Exempeloutput

34

5. Skattjakten 2

Gör inte denna uppgift ifall du inte klarade av förra uppgiften.

Ifall man får ta kontrollerna i vilken ordning som helst, hur långt måste ni gå då?

6. Spel

Den här uppgiften har en annan stil än de tidigare uppgifterna. Istället för ett litet problem, ska du skriva en egen version av det världsberömda spelet Pacman. Då detta antagligen kommer vara mer tidskrävande för dig än de tidigare uppgifterna så måste du inte implementera en lösning för alla de förslag som nämns här. Gör så mycket du kan/hinner med. Se till att vi som kommer läsa din kod kan förstå den, och även att vi kan testköra spelet på våra datorer.

Då det numera finns massor av versioner av Pacman online så får du själv lite frihet i exakt hur spelet ska se ut och fungera. Ett par minimumkrav som spelet bör uppfylla är:

- Det bör finnas ett fönster som ritar upp spelet.
- Man ska kunna kontrollera spelet med tangentbordet.
- Det bör finnas spöken som rör på sig, och man ska förlora ett liv eller få game over om pacman går in i ett spöke.

Ifall du klarar av att få ett fungerande spel så kan du passa på att visa upp dina kunskaper genom att lägga till lite coola saker till spelet. Här är ett par exempel, men du kan även hitta på egna coola features:

- Implementera samma algoritmer för spökens AI som i originalspelet. En beskrivning av spökens AI finns t.ex. här:
<https://aighost.co.uk/how-pac-man-ghost-ai-works-the-classic-chase-algorithms/>.
- Uppdatera programmet så att pacman styrs av datorn. Hur datorn kontrollerar pacman får du själv välja. Det kan vara allt från ett par if-satser om var spökena finns till ett neuralt nätverk, eller någonting däremellan.