
Instuderingsfrågor inför prov 2 i Programmering 2

Prov 2 omfattar: Lektionerna **29** - **xx** på webben m(o).
Kursboken, **kap 4-5**, **sid 95-197**, inkl.:
Frågor/Övningar: kap 4, sid 132
kap 5, sid 193-197

Svar till alla frågor kan hittas i kursboken [Programmering 2 med C++](#)

Alla program- och sidoreferenser hänvisar till kursboken.

Kap 4 Pekare, sid 95-132

1. Redogör för de två olika betydelserna av ampersand `&` i referenssammanhang och ge exempel.
2. Varför kan man säga att referens en *automatiserad* pekare? Använd bilden på sid 88.
3. Skriv en pekarversion av programmet `Array_Int` (sid 77). Skapa arrayen med

```
int *no = new int[4];
```

och ersätt överallt *index* med *pekarposition*.
4. Vilken parameteröverföringsmetod används i C/C++ när parametrarnas datatyp är pekare?
5. Vad är den mest relevanta skillnaden mellan statisk och dynamisk minnesallokering?
6. Nämn de fem olika platser där datorn lagrar data när ett program kompileras och exekveras.
7. Är de platser du nämnde i frågan ovan (73) hårdvaru- eller mjukvarukomponenter?
8. Vilket är datorns snabbaste minne? Var är det lokaliserat? Skriv kod som skapar en variabel med snabbast möjliga åtkomst.
9. Med satsen `auto float c;` skapas en *automatisk* variabel. I vilken minnesenhet kommer denna variabel att lagras?
10. Blir det en statisk eller dynamisk minnesallokering med satsen `auto float c;` ? Vilket ord i satsen får utelämnas utan att det blir fel?
11. Vad heter den del av RAM som kompilatorn inte har tillgång till?
12. I vilken minnesenhet allokeras variabler som skapas med operatoren `new`? Blir det en statisk eller dynamisk allokering?
13. Vad heter den inversa operatoren till `new`? Vad gör den?
14. Om satsen `static float b;` står i en funktion kommer den lokala variabeln `b` att gälla i funktionen när den anropas, eller även leva vidare efter funktionsanropet?

15. Vilka processer körs i lagringsplatsen *Non-RAM* ?
16. Vad är nollpekaren? Hur skrivs den i kod? Vad kan den användas för?
17. Testa programmet **Dynamic** (sid 96) och förklara utgående från koden och med egna ord, vad som händer när det körs. Testa olika inmatningar, även med lite större textmängd.

Kap 5 Objektorienterad programmering, sid 133-197

18. Vad menas med paradigmskifte i programmeringens historia?
19. Mellan vilka två programmeringsspråk går historiskt skiljelinjen mellan procedural och objektorienterad programmering? När ungefär inträffade övergången?
20. Vad var anledningen till paradigmskiftet inom programutveckling?
21. Vilka för- och nackdelar har enligt din åsikt den procedurala synen på programmering?
22. Vilka för- och nackdelar har enligt din åsikt den objektorienterade synen på programmering?
23. Är det korrekt att pepparkakor är klasser och pepparkaksformen objekt?
24. Kan man via abstraktion komma från objekt till klass eller är det tvärtom?
25. Om pennor är objekt var kan man hitta klassen penna?
26. Av vilka två huvudingredienser består en klass i regel?
27. Anta att Tal är en klass. Är addition() en metod eller en datamedlem i klassen Tal?
28. Anta att Bil är en klass. Är Motor en metod eller en datamedlem i klassen Bil?
29. Vad är skillnaden mellan funktioner och metoder i C++?
30. Vilka är den objektorienterade programmeringens tre hörnstenar?
31. Vad innebär modularisering på klassnivå?
32. Är + en metod, en datamedlem eller en operator i klassen String? Vad heter den?
33. Med vilken fråga hittar man datamedlemmarna i en klass?
34. Med vilken fråga hittar man metoderna i en klass?
35. Är objekt och instans synonymer?
36. Vad betyder punktnotation?
37. Vad betyder allokering i datasammanhang?
38. Vilka är de tre hörnstenarna inom objektorienterad programmering?
39. Pekar i ett klassdiagram pilen från subklassen till superklassen eller tvärtom?
40. När en subklass ärver en superklass, vilka delar av superklassen ärver den?
41. Varför är polymorfism en speciell form av överlagring av metoder?
42. Vad skiljer polymorfism från vanlig överlagring av metoder?
43. Kan polymorfa metoder förekomma i en och samma klass?
44. Kan överlagrade metoder förekomma i en och samma klass?
45. Kan man komma åt de privata datamedlemmarna i en superklass från en subklass?

