



PROGRAMMING MED C#

Examination - Inlämningsuppgift

Hamnen



Uppgiften

- Hamn-administration
 - Vi ska bygga ett hamnadmin för en hamn med 64 hamnplatser.
 - Hamnen besöks av fyra olika sorters båtar
 - Roddbåt
 - Motorbåt
 - Segelbåt
 - Lastfartyg

Båttyperna - Roddbåt



- Roddbåt
 - Tar upp en halv hamnplats (två roddbåtar ryms på en plats)
 - Egenskaper
 - Identitetsnummer (slumpmässigt tre bokstäver med prefix "**R-**", t ex R-ABC)
 - Vikt (100-300 kg)
 - Maxhastighet (0-3 knop)
 - Unik egenskap:
 - Max antal passagerare (1-6 personer)
- Roddbåtar stannar i hamnen 1 dag, innan de åter ger sig ut på havet.

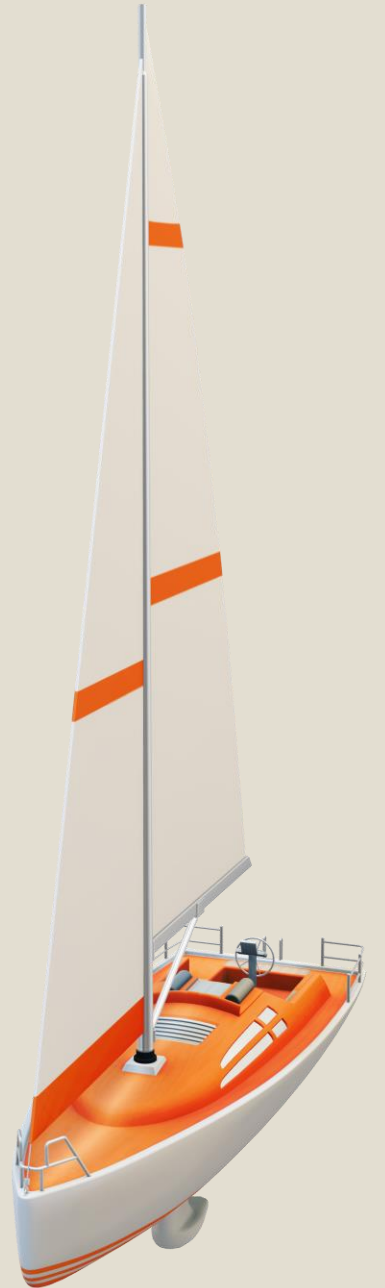
Båttyperna - Motorbåt



- Motorbåt
 - Tar upp en hel hamnplats (en motorbåt rymmer på en plats)
 - Egenskaper
 - Identitetsnummer (slumpmässigt tre bokstäver med prefix "**M-**", t ex M-ABC)
 - Vikt (200-3000 kg)
 - Maxhastighet (0-60 knop)
 - Unik egenskap:
 - Antal hästkrafter (10-1000 hk)
- Motorbåtar stannar i hamnen 3 dagar, innan de åter ger sig ut på havet.

Båttyperna - Segelbåt

- Segelbåt
 - Tar upp två hela hamnplatser
 - Egenskaper
 - Identitetsnummer (slumpmässigt tre bokstäver med prefix "**S-**", t ex S-ABC)
 - Vikt (800-6000 kg)
 - Maxhastighet (0-12 knop)
 - Unik egenskap:
 - Båtlängd (10-60 fot)
- Segelbåtar stannar i hamnen 4 dagar, innan de åter ger sig ut på havet.



Båttyperna - Lastfartyg

- Lastfartyg
 - Tar upp fyra hela hamnplatser
 - Egenskaper
 - Identitetsnummer (slumpmässigt tre bokstäver med prefix "**L-**", t ex L-ABC)
 - Vikt (3000-20000 kg)
 - Maxhastighet (0-20 knop)
 - Unik egenskap:
 - Last, antal containers på fartyget just nu: 0-500
- Lastfartyg stannar i hamnen 6 dagar, innan de åter ger sig ut på havet.



Förutsättningar

- Varje dag inkommer 5 slumpmässiga båtar. (Gör det möjligt att enkelt ändra det värdet)
- Båtarna ska placeras i hamnen så effektivt som möjligt, så det är så få luckor som möjligt
- Roddbåtarna ska placeras två och två, på samma hamnplats.
- Ingen båt får flyttas efter att de placerats
- I takt med att båtarna lämnar hamnen uppstår luckor, som inkommande båtar i första hand ska fylla
- Hamnens båtbestånd ska kontinuerligt sparas som en fil, så applikationen kan startas om, utan att båtarna försvinner.
- En "dag" är antingen t ex 5 sekunder, eller så växlas dagarna med tangenttryckning eller knapp-klick.
- Om en båt inte kan placeras i hamnen, så måste båten avvisas. Håll reda på hur många de är.

Information som ska visas

- Först och främst ska de **64** platsernas beläggning redovisas, antingen som en lista, eller grafiskt.
- Dessutom ska båtarnas egenskaper redovisas i kolumnform, omräknade till metriska enheter. Dessa är identitetsnummer, vikt, maxhastighet (omräkning från knop till km/tim).
- I sista kolumnen visas båtens unika egenskap, t ex antal passagerare för roddbåt.
- Dessutom ska en summering av båthamnens bestånd visas ovanför eller nedanför listan:
 - Antal rodd- segel- motorbåtar samt lastfartyg i hamnen
 - Total vikt i hamnen dvs summan av alla båtars vikt
 - Medeltal av alla båtars maxhastighet
 - Antal lediga platser
 - Antalet avvisade båtar (de båtar som inte fått plats)

Utseende

- I den enklaste formen byggs detta som Console-projekt och på skärmen ser man tex följande:

Plats	Båttyp	Nr	Vikt	Maxhast	Övrigt
1-2	Segelbåt	S-SDF	4535	18 km/h	Båtlängd: 11 m
3	Tomt				
4	Roddbåt	R-ETR	420	2 km/h	Max Antal pers: 3
4	Roddbåt	R-WLM	632	1 km/h	Max Antal pers: 4
5	Tomt				
6-9	Lastfartyg	L-CEK	12500	14 km/h	Containers: 129
...osv till plats 64					

Betyg

- Godkänt:
 - Uppgiften löses med den kunskap vi hittills lärt oss, såsom klasser med arv, kollektioner, loopar och arrayer.
 - Använd Console eller WPF
- Väl godkänt
 - Uppgiften löses med den kunskap vi hittills lärt oss, såsom klasser med arv, kollektioner, loopar och arrayer.
 - Dessutom ska delegater, lambdayttryck och LINQ användas.
 - Dessutom ska följande "försvårande" omständigheter hanteras:
 - Hamnen är uppdelad på två kajer, med 32 platser per kaj.
 - En ytterligare båttyp
 - Katamaran, som upptar 3 hamnplatser, **och som ligger kvar i hamnen i tre dagar**,
 - Identitetsnummer (slumpmässigt tre bokstäver med prefix "**K-**", t ex K-ABC)
 - Vikt (1200-8000 kg)
 - Maxhastighet (0-12 knop)
 - Unik egenskap: Antal bäddplatser (1-4)
 - Grafiskt utseende med WPF

Tips

- Börja med att skissa hamnen och båtar på papper
- Fundera över hur man enklast ser till att hamnen är sparsamt utnyttjad.
- Oavsett slutgiltig lösning (WPF eller Console, G eller VG-nivå), börja med att göra den enklaste lösningen, och se till att det fungerar. Bygg därefter vidare på uppgiften.
- Just roddbåtarna kan vara en utmaning, då det ryms två båtar per plats, och finns det en roddbåt på en plats kan den inte användas till någon annan båttyp. Fundera över hur det hanteras bäst.

Frågor

- Som vanligt finns lärarna tillgängliga för frågor.
- Men tänk på att uppgiften också är en examination, för att vi ska se vad ni lärt er.

Redovisning

- Muntlig redovisning sker torsdag den 5:e November, inför gruppen.
 - Dels visar ni applikationen, dess funktion och eventuella finesser
 - Dels visar ni lite av koden, främst hur ni lagt upp klasser och metoder.
 - Dels visar ni eventuella smarta lösningar som ni gjort i koden.
- Inlämning sker direkt efter redovisningen
- Formen för redovisningen, med tanke på eventuella elever som är på distans återkommer vi till.
- Det är att föredra att vara på plats i skolan just den här dagen.