

# Examinationsbeskrivning

## Individuell inlämningsuppgift

Kurs: Programmering C# .NET

Termin: Höstterminen 2022

Ansvariga lärare: Paul Tannenbergl

E-post: [pata01@gapp.uddevalla.se](mailto:pata01@gapp.uddevalla.se)

Tel: 0737-003138

- Beskrivning

Denna uppgift testar alla i kursen förekommande kunskaper, färdigheter och förmågor. Det finns möjlighet att visa VG nivå.

Ni skall skapa en program för att hantera en parkeringsplats vid slottet i prag. Appen kallas för Prague Parking

Parkeringsplatsen har 20 platser. En parkeringruta kan hantera en bil eller en mc eller tom. För VG så skall en parkeringruta även kunna hantera 2 mc.

Det skall vara en enkel **objektbaserad lösning**, arv är inte nödvändigt för att lösa uppgiften, där de parkerade fordonen kan hanteras i en fordonsklass som har följande minst egenskaper.

RegNr

Fordonstyp - Bil eller MC

Ankomsttid -

Obs Ett fordon känner INTE till var det är parkerat.

Parkeringsrutorna hanteras förmodligen enklast som en array 20 eller 20, 2 i storlek men om ni föredrar en annan lösning så är det tillåtet.

Det finns inga krav på att hantera användare eller fordonsägare i denna lösning.

Programmet skall hantera minst nedanstående funktioner...lämpligen genom en meny och metoder

- Inlämning av fordon - ta in regnr samt fordonstyp och försök parkera det i mån av plats, glöm inte att lagra ankomsttiden. Programmet väljer själv plats.
- Utlämning av fordon - Hämta ut fordonet med ett visst regnr, mc kostar 25kr per påbörjad timme, bilar kostar 40kr per påbörjad timme.

- Flyttning av fordon till en viss plats (använd fordonets regnr samt den önskade platsen)
- Sökning efter fordon skall tala om var fordonet med angivet regnr står (samt fordonstyp)
- Utskrift av innehållet i alla parkeringsrutor

Se till att dina metoder tar parametrar och gärna ger returvärden.  
Om ett fordon inte kan hittas....regnumret finns inte så tala om det.

**Felhantering** skall finnas så att programmet inte kraschar i samband med felaktiga inmatningar i menyn.

## 1. Bedömning

### 1.1 Mål som testas av denna examination

*Efter genomförd kurs ska den studerande ha kunskaper i/om:*

#### Kunskaper

*Efter genomgången kurs ska den studerande kunna*

- om IDE:n Visual Studio
- C# -syntax och semantik
- sekventiell programmering
- värdetyper och referenstyper

#### Färdigheter

*Efter genomgången kurs ska den studerande ha färdigheter i att*

- Felsöka och debugga
- Arbeta i Integrerade utvecklingsmiljöer (IDE) – Visual Studio

#### Kompetenser

*Efter genomgången kurs ska den studerande ha kompetens att*

- självständigt kunna utveckla enklare system i .NET och C#.
- skapa kod av god kvalité som är enkel att underhålla och vidareutveckla
- använda relevanta metoder för att felsöka kod

### 1.2 Betygskriterier

För att nå upp till en **G-nivå** ska den studerande ha nått samtliga mål som testas i denna examination gällande kunskaper, färdigheter och kompetenser.

På denna uppgift så visar då på VG-nivå när du

- Den studerande har nått samtliga mål för kursen enligt ovanstående G-nivå samt att den studerande visar förmågan att lösa programmeringsrelaterade uppgifter på ett

genomtänkt sätt som påvisar djupare förståelse för kodens uppbyggnad. Du kommer även att visa din djupare förståelse i din redovisning

## 2. Instruktioner

Ni ska bygga ett objektbaserad version av Prague Parking

I er applikation skall följande funktionalitet finnas med:

- Funktionalitet Meny : Byggs med hjälp av en evighetsloop samt break
- Funktionalitet Parkeringsrutor : Hanteras enklast med en array.
- Funktionalitet Funktioner för att lämna in, hämta ut, flytta och söka efter fordon samt
- Funktionalitet utskrift av innehåll på parkeringsplatsen...
- Funktionalitet: Initieringsrutin, initiera pplatsen med 3 bilar och 3 mc.

För att kunna uppnå VG-nivån på denna examination behöver ni som framgår av betygskriterierna

Exempel på hur ni uppnår detta är genom att:

- Gör en grafiskt snygg utskrift av parkeringsplatsen innehåll, använd färger och specialtecken för att göra den grafiskt trevlig.
- Logga alla uthämtade fordon i en textfil med all data, inklusive avhämtningstid, tid på parkeringen samt priset
- Motorcyklar är luriga, även om man har en snygg inmatningsrutin som optimerar så att man försöker hitta en plats för en mc där det redan så en mc så kan hamna i ett läge där många mc har blivit uthämtade och att flera p-rutor har en ensam mc i sig. Skriv en rutin som optimerar p-rutor så att alla ensamma mc flyttas ihop och att det bara finns max en ensam mc kvar.

Självklart kan ni, om ni siktar på en VG-nivå, även bolla med mig för att veta hur och vad ni skall göra för att nå dit.

### 2.1 Arbetsgång

Nedan finner du ett förslag på arbetsgång för att på bästa sätt ta dig an denna uppgift. Du måste inte följa denna arbetsgång, men det kommer att underlätta för dig när du bygger ditt Prague Parking

1. Börja med att rita upp hur du vill designa din applikation utifrån de funktionalitetskrav som du finner ovan.
2. Din initieringsrutin
3. Och sedan menyn
4. Därefter bygger du inlämningsmetoden
5. Sedan sökmetoden...då får du mycket lättare att avlusa och testa din kod
6. Därefter så bygger du metoderna i valfri ordning.
7. Nu behöver du välja vilken extra funktionalitet du vill bygga in för att nå VG-nivån

## **2.2 Inlämning och genomförande**

Inlämning av denna examination sker genom inlämning via classroom den 25/10 23:59.