

Inlämningsuppgift 2

Projektets namn

Förnamn1 Efternamn1 (anvnamn1)

Förnamn2 Efternamn2 (anvnamn2)

Objektorienterad programmering

Höstterminen 2012

Kursansvarig: Henrik Bergström

Innehåll

1 Inledning

Inledningens syfte är att leda in läsaren i arbetet. Här ska ni alltså ge en kort beskrivning av systemet och dess funktionalitet. Det finns inga speciella krav på omfattning, men 1/2–1 sida bör vara ganska lagom.

2 Design

Här ska ni beskriva designen av systemet. Ett lämpligt format är ett klassdiagram kompletterat med en kort textuell beskrivning. Var noga med att redovisa vilka delar ni har gjort själva och vilka som är tagna från Java eller från andra bibliotek. För ett vanligt projekt bör 2 sidor räcka mer än väl för denna del. Har man en avancerad design

3 Kravuppfyllnad

Här ska ni visa att ni uppfyller de formella kraven på uppgiften. Vissa av dessa har redan täckts av de föregående delarna så håller till de krav som listas nedan. Normalt räcker det med ett eller ett par stycken, och kanske ett exempel, per krav som visar att ni har tänkt på det och uppfyller det.

3.1 Samverkande klasser

3.2 Arv och polymorfism

3.3 Namngivning

3.4 Korrekta skyddsnivåer

4 Reflektion

Den sista delen i rapporten är tänkt för egenreflektion. Vad har fungerat bra? Vad har fungerat dåligt? Var uppgiften för lätt? För svår? Är det något ni borde ha övat mer på innan ni satte igång? Är det något vi borde ha ägnat mer tid åt eftersom ni behövde det?

5 Mallanvändningen

Detta avsnitt tar upp några vanliga frågor/problem med mallandvändningen. Det ska naturligtvis plockas bort i den version ni lämnar in.

5.1 Bilder

Bilder ska tillföra något till texten, vara tydliga, ha en förklarande bildtext, och sitta på ett vettigt ställe i förhållande till texten. De bör också refereras till i texten. Tänk på att bilder i \LaTeX bör placeras i floats, och att dessa kan flyttas runt automatiskt. Du måste alltså anta att bilden kan flyttas runt om du gör några ändringar. Man bör alltså alltid referera till figurer med hjälp av labels: Figur

Money
+Money() +Money(in money : Money) +Money(in kronor : int, in öre : int) +getKronor() : int +getÖre() : int +add(in addend : Money) : Money +subtract(in subtrahend : Money) : Money +isPositive() : boolean +isNegative() : boolean +isZero() : boolean +toString() : String +toString(in characters : int) : String

Figur 5.1: Exempelbild

Rubriknivå 3

Rubriknivå tre och neråt bör man normalt undvika att använda. Ibland går det inte, men ofta är det ett tecken på att man delar upp sitt dokument i för små bitar.

5.2 Referenser

Referenshanteringssystemet till L^AT_EX heter BibTex, och det ska man naturligtvis använda. På OOP använder vi IEEE:s format med numrerade referenser inom hakparanteser. En referens till kursanvisningarna skulle alltså se ut så här: [?], eller så här om man vill ha med sidanvisningar: [?, s. 5].

5.3 Kod

Kod i dokumentet ska naturligtvis vara läsbar och helst formaterad som i en utvecklingsmiljö. Ett tips är att inte använda skärmdumpar. De blir nästan alltid oläsliga, och det går inte att få ut källkoden från dem om man skulle vilja.

Mallen använder sig av listingspaketet (<http://mirror.hmc.edu/ctan/macros/latex/contrib/listings/listings.pdf>) som kan formatera kod antingen direkt i latexkoden som i exempel

Inställningarna för programspråk, radnumrering, etc. finns i direkt anslutning till där paketet importeras i latexmall.tex.

Kodexempel 5.1: Kod som ligger i latexfilen

```
1 || class Exempel{
2 ||
3 ||     int number;
4 ||
5 || }
```

Kodexempel 5.2: Kod som hämtas från fil

```
1 || class Exempel{
2 ||
3 ||     int number;
4 ||
5 || }
```

Kodexempel 5.3: En rad från en fil

```
1 ||     int number;
```


5.4 Avsnitt i mallen

Den text som finns i mallen är naturligtvis avsedd bara för att illustrera användningen. Har ni, till exempel, inte några bilagor plockar ni naturligtvis bort dessa från dokumentet.