

# Kodprov "testomgång" #1

## Instruktioner

Checka ut ditt repo: `git clone http://USERNAME:PASSWORD@wrigstad.com/git/USERNAME.git`

Nu får du en katalog `USERNAME` som beroende på vilken tenta du skriver kan innehålla filer som du skall bygga vidare på. Vissa tentor har inga uppgifter med utdelad kod i katalogen `utdelad-kod`. Du skall *inte* lägga dina filer under denna katalog!

Du lämnar in tentan via git. Du kan skriva `git status` för att få se vilka filer som inte är incheckade, använda `git add` för att lägga till filer, `git commit` för att checka in tillagda filer, etc. **OBS!** Om du inte skriver `git push origin master` efter din sista incheckning finns tentan bara på din lokala hårddisk och har då *inte* lämnats in!

Skicka också hela din tenta i ett zip-arkiv till `tobias.wrigstad@gmail.com`.

## Uppgifter

1. Filerna i paketet `graph` implementerar en graf med noder sammanlänkade med kanter. Det skall vara möjligt att kopiera en hel graf så att det skapas en helt separat kopia som inte delar några objekt med sitt original. Skriv färdigt metoden `deepClone` i berörda klasser så att följande program producerar en identisk kopia av grafen `g1` i variabeln `g2` som inte delar några objekt med grafen i `g1`.

*Notera: grafen kan mycket väl innehålla cykler.*

```
1   Graph g1 = new Graph();
2   ... // code that builds up g1 with 100+ nodes with cycles
3   Graph g2 = g1.deepClone();
4   // g2 can be used wherever g1 can be used, but modifications
5   // to g2 aren't visible in g1 and vice versa
```

Ledning: om grafen inte hade cykler är uppgiften väldigt enkel. För att hantera cykler, använd `DeepCloneHelper` som är en avbildning från objekt som redan kopierats till deras kopior.

### Att lämna in:

- (a) De modifierade filerna `*.java` (plus eventuella tillagda)
  - (b) Filerna skall ligga i katalogen `uppgift1`
2. Programmet `list` implementerar en länkad lista. Skriv klart programmet genom att implementera stöd för att avallokera listan och stöd för iteratorer i enlighet med `list.h` sådan att följande kod skulle kompilera (och fungera korrekt):

```
1   list *l = mkList();
2   append(l, (void*) 1);
3   append(l, (void*) 4);
4   append(l, (void*) 91);
5   iterator *i = mkIterator(l);
6   int sum = 0;
7   while (hasMore(i)) {
8       sum += (int) next(i);
9   }
10  printf("Summa: %d\n", sum);
11  rmIterator(i);
12  rmList(l);
```

De funktioner som finns i headerfilen men som saknas i `list.c` skall alla implementeras.

### Att lämna in:

- (a) De modifierade filerna `*.h` och `*.c` (plus eventuella tillagda)
- (b) Filerna skall ligga i katalogen `uppgift2`