# Tentamen i TDP004
# Objektorienterad Programmering
# Praktisk del

| | |
|---|---|
| Datum: | 2008-08-14 |
| Tid: | 08-12 |
| Plats: | PC6-PC7 i E-huset. |
| Jour: | Per-Magnus Olsson, tel 285607<br>Jourhavande kommer att besöka skrivsalarna varje timme under skrivtiden. |
| Hjälpmedel: | Teoretisk del: Inga.<br>Praktisk del: En bok om C++. |
| Betygsättning: | Max antal poäng: 44 med 22 poäng vardera på teori och praktikdel. |

| Poäng | **Betyg** |
|---|---|
| 38-44 | **5** |
| 30-37 | **4** |
| 23-29 | **3** |
| 0-22 | **U** |

Anvisningar: Börja med den teoretiska delen. När du är klar med den lämnar du in den och får den praktiska delen. När du har lämnat in den teoretiska delen kan du inte återvända till den.

Skriv svaret på varje uppgift på ett separat blad.

Uppgifterna är inte ordnade i svårighetsgrad.

Lycka till!

# TDP004 Objektorienterad Programmering
# Praktisk del

1. Implementera klassen Lightbulb med två public-variabler: name av typen std::string och strength av typen int. Skriv sedan en medlemsfunktion i samma klass:
   void PrintSortedLightbulbs(std::vector<Lightbulb*> lightbulbs)
   vilken först sorterar i ordning enligt följande ordning: glödlamporna ska sorteras i alfabetisk ordning på sortens glödlampa, och om det finns flera glödlampor av samma sort så ska dessa sorteras så att en med lägre styrka kommer före en med högre styrka. Efter att vectorn har sorterats ska den skrivas ut, från första till sista elementet. Ovanför utskriften av typen och styrkan ska det stå ”Type strength”. Exempel på utskrift:

   ```
   Type strength
   -----------------
   Diode 5
   Diode 12
   LowEnergy 10
   LowEnergy 30
   Normal 25
   ```

   (7p)

2. Implementera en basklass Testcase med de båda subklasserna TestAddition, TestSubtraction. Klassen TestCase ska ha en funktion Run vilken ska ta två parametrar av typen const int, och utföra lämplig operation på dessa, och resultatet av operationen ska returneras. Subklasserna ska implementera funktionen Run, och vilken operation som ska utföras kan man se i klassnamnet. Låt kompilatorn verifiera att funktionen Run inte ändrar några medlemsvariabler.

   Efter att du har implementerat dessa klasser, skapa en instans av varje subklass och lägg dessa i en std::list. Iterera genom alla instanser i listan och anropa Run för vart och ett av elementen, med av dig valda inparametrar (10p).

3. Skriv en funktion int ConvertToBinary(const std::string number) som konverterar en binär representation av ett tal till en vanligt heltal. Du får förutsätta att number enbart innehåller ettor och nollor. Exempel: om ConvertToBinary anropas med en sträng innehållande 110111 så bör den returnera 55, om den anropas med 101 så ska den returnera 5 etc (5p).