

Tentamen i TDP004

Objektorienterad Programmering

Praktisk del

- Datum: 2010-08-27
- Tid: 14-18
- Plats: SU-salar i B-huset.
- Jour: Per-Magnus Olsson, tel 285607
Jourhavande kommer att besöka skrivsalarna ungefär varje timme under skrivtiden.
- Hjälpmedel: Teoretisk del: Inga.
Praktisk del: Den C++ information som finns i systemet.
- Betygsättning: Max antal poäng: 42 med 21 poäng vardera på teori och praktikdel.
- | Poäng | Betyg |
|-------|----------|
| 36-42 | 5 |
| 30-35 | 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.
Uppgifterna är inte ordnade i svårighetsgrad.
En kort hjälptext till Eclipse finns på sid 2-3.

Lycka till!

Kort Eclipsehjälp

Om du inte är bekant med Eclipse rekommenderas att du läser igenom följande guide.

Att komma igång

Skriv `eclipse &` i terminal. Efter viss väntetid kommer ett fönster att visas. Välj ”C++ Perspective” i övre högra hörnet av fönstret. ”C++ Perspective” finns under ”Other”.

Välj ”Managed Make C++ Project” som din typ av projekt. Gör ett projekt för varje uppgift, om inte annat sägs i uppgiften.

Du måste även skapa en fil som har funktionen `main` definierad. Det går bra att skapa en `.cpp`-fil och där definiera funktionen `main` enligt nedan, vilken blir det som operativsystemet anropar du när exekverar programmet.

```
int main()
{
    //Övriga anrop görs här.
    return 0;
}
```

Om du inte skapar en `main`-funktion så kommer kompilatorn att säga att funktionen `main` saknas och att ett allvarligt symbolreferensfel har inträffat.

Ovanstående är viktigt för att Eclipse ska bete sig korrekt med avseende på kompilering etc.

Inkludera eventuella givna filer

För att använda programfiler som finns i katalogen `given_files`, kopiera dem till ditt workspace. Högerklicka sedan på ”C/C++ Project” vilken normalt är ett vertikalt fönster finns längst till vänster i huvudfönstret. Välj sedan ”Refresh” i den meny som visas. De inkluderade visas nu tillsammans med dina egna filer. Programfilerna kan nu inkluderas som vanligt.

Datafiler behöver endast kopieras till ditt workspace.

Kompilera

I ”C/C++ Project”, högerklicka på projektet och välj ”Build” i menyn. Som default är Eclipse inställt på att automatiskt kompilera under tiden du programmerar.

Om kompileringen krånglar

Högerklicka på projektet och välj ”Properties”->C/C++ Build. Välj fliken ”Build Settings”. Klicka bort ”Use default command”. I rutan under, skriv ”`gmake -k`”. Klicka på OK i nedre högra hörnet.

Automatisk kompilering

Som default är Eclipse inställt på att kompilera alla filer så snart du har slutat skriva. Om du vill avaktivera den funktionen, gå till ”Project”-menyn och klicka på texten ”Build Automatically”.

Ta bort alla kompillerade filer

Om du vill ta bort alla kompillerade filer (källkodsfiler är opåverkade): gå till "Project"-menyn och välj "Clean".

Köra program

I "C/C++ Project", välj "Run as"->"Run as local C/C++ application".

Det går även att välja "Run"-knappen i menyn längst upp i fönstret.

Om fönster inte visas

Om konsollen med resultatet av programkörningar inte visas, så kan du aktivera visningen av fönstret genom att gå till "Project"-menyn, och välja "Show window" och sedan välja "Console". Om "Problems"-fönstret som visar eventuella kompileringsfel inte visas, gör samma sak men välj "Problems" istället.

Om Eclipse verkar ha låst sig

Prova att växla mellan olika fönster med Alt-Tab. Ibland hamnar pop-up-fönster som väntar på din input bakom huvudfönstret. Om det inte fungerar kanske du måste stänga Eclipse från konsollen och starta om.

TDP004 Objektorienterad Programmering

Praktisk del

OBS!

Sedan senaste tentamen har systemet för datortentamen bytts ut. Det nya systemet kan medföra vissa temporära problem. Ett problem som man kan drabbas av är att Eclipses defaultinställningar för att kompilera inte fungerar och man får ett felmeddelande liknande:

```
make -k all
```

```
make: Fatal error in reader: makefile, line 6: Unexpected  
end of line seen
```

Då bör man ändra så att gmake används istället för make, vilket man gör på följande sätt: högerklicka på ditt projekt och välj "Properties" och sedan "C/C++ Build". Välj fliken "Build Settings". Klicka bort "Use default command". I rutan under, skriv gmake -k. Klicka på OK i nedre högra hörnet.

OBS!

1. Ett lösenord i ett visst system måste uppfylla samtliga följande krav:

- Det måste vara minst 5 och max 15 tecken långt.
- Det måste innehålla både siffror och bokstäver.

Gör en funktion med nedanstående signatur

```
bool check_password(const std::string password)
```

som testar huruvida lösenordet uppfyller kraven. (5p)

2. a) Den seriöse programmeraren har naturligtvis med sig en dator var hon/han än befinner sig, även så här i slutet av grillsäsongen. En användning för datorer i det här fallet är en beräkning av hur lång tid en viss mängd mat tar att tillaga. Olika sorters mat att grilla tar olika lång tid beroende på mängden, och i det här fallet finns tre olika sorters mat: korn, fisk och det vegetariska alternativet quorn. Grilltiden beräknas enligt nedan. De olika maträtterna ska även separat kunna svara på frågan om lämplig tillagningstemperatur. På grund av att grillen har den svårt att hålla en exakt temperatur, används benämningarna låg, mellan och hög temperatur enligt nedan.

Korn: 5 min per 0,2 kg, på mellan-temperatur.

Fisk: 10 min per kg fisk, på låg temperatur.

Quorn: 2 min per 0,1 kg, dock max 5 min, på hög temperatur.

Implementera en arvshierarki för olika sorters mat där de olika sorternas mat kan beräkna grilltiden enligt ovan. (10p)

b) Kontrollera att din implementation från uppgift 2 a) fungerar som avsett genom att skriva ett program som efterfrågar tillagningstemperatur och tid för de olika sorternas mat. Din kontroll ska använda polymorfism. (6p)