

Tentamen i 2D1387 Programsystemkonstruktion med C++

Datum: 20 december 2000

Hjälpmedel: Valfri lärobok i C++

Tid: 5 timmar

Skriv tydligt och ge motiveringar till dina svar. För godkänt krävs 25 poäng, för betyg fyra krävs 34 poäng, för betyg fem krävs 40 poäng. Alla betygsgränser är inklusive bonuspoäng. Maxpoäng är 45 utan bonus. Maximalt antal bonuspoäng är nio.

1. Betrakta följande program:

```
struct A {
    virtual void foo() { cout << "A" << endl; }
};
struct B : public A {
    void foo() { cout << "B" << endl; }
};

int main()
{
    A *ap = new B;
    A &ar = *ap;
    B b;

    ap->foo();
    ar.foo();
    b.foo();

    delete ap;
    return 0;
}
```

- a) Vad skriver programmet ut när man kör det? 3p
- b) När bör man och när behöver man inte ha en virtuell destruktör? Motivera ditt svar. 4p
- c) Om virtuella funktioner inte innebar några nackdelar skulle man kunna låta alla funktioner vara virtuella. Förklara varför det är olämpligt att låta samtliga medlemsfunktioner vara virtuella. 5p

2. Mallfunktionen (*template function*) `sum_matrix` summerar alla element i en $n \times n$ -matris med element av typ T. För att kunna använda `sum_matrix` måste vi ställa vissa krav på datatypen i matrisen. I vårt fall kräver vi att T har en överlagrad `operator+=`. Observera att `sum_matrix` klarar mer än att bara summera heltalstyper, eftersom alla typer med överlagrad `operator+=` kan hanteras (som t.ex. `std::string`).

a) Hur deklarerar du funktionen `sum_matrix`? 3p

b) Implementera `sum_matrix` så att den utför det arbete som specificeras ovan. 6p

3. Följande program innehåller fyra fel och kan därför inte kompileras. Vilka är felen och hur kan man avhjälpa dem? Motivera. 10p

```
class A {
    A() {}
public:
    int j;
    static int i;
};
class B : public A {
    B() : j(2) {}
};

int main()
{
    B b;
    A::i = 7;
    return 0;
}
```

4. Antag att vi har en klass A med en medlem `e`. Konstruktorn i A kan implementeras på olika sätt:

```
A::A(int i) : e(i) {}           // (1)
A::A(int i) { e = i; }         // (2)
```

a) Det finns fall då implementation (2) av konstruktorn inte kan användas trots att (1) fungerar (beroende på hur `e` är deklarerad). Ange minst två av dessa fall och motivera varför kompilatorn ger fel i (2). 4p

b) Låt `e` vara av typen T, definierad nedan. Vilka funktioner anropas i T då man använder implementation (1) respektive (2) av `A::A(int)`? Motivera. 4p

```
struct T {
    T();
    T(int);
};
```

c) Vissa medlemsfunktioner ges av kompilatorn om man inte implementerar dem. Ge deklARATIONEN för fem sådana funktioner och förklara hur de fungerar. 6p