



## Försättsblad Prov Original

Kurskod	Provkod	Tentamensdatum
D T 0 4 7 G	T 1 0 3	2 0 1 9 - 0 1 - 1 8
Kursnamn	Datateknik GR (B), Programmeringsmetodik med projekt	
Provnamn	Skriftligt teoriprov	
Ort	Sundsvall	
Termin		
Ämne		



Tentamen, moment T104

dt047g, dt079g Programmeringsmetodik

Martin Kjellqvist\*

2019-01-18

## Instruktioner

Skriv svaren på tentan. Skriv tydligt. Om svaret är oläsligt får du 0 poäng - även om svaret är korrekt. Frågorna är *inte* ordnade efter svårighetsgrad.

*Ett svar är noll, ett eller flera val av de givna alternativen.*

Tid 5 timmar.

Hjälpmedel Inga.

Antal frågor 10

## Preliminära gränser

Godkänd:  $\geq 70\%$  korrekta svar.

## Frågor

1. Vilket alternativ ger iteratorkategorierna i sin hierarkiska ordning.
  - A. Random, Input, Reverse, Bidirectional, Pointer
  - B. Pointer, Forward, Input, Output
  - C. Pointer, Forward, Bidirectional, Random
  - D. Input, Output, Forward, Bidirectional, Random
  - E. Input, Output, Forward, Reverse, Random

1. \_\_\_\_\_

2. ODR? (1 alternativ är rätt)
  - A. Optimal definition resource
  - B. Often defined resource

---

\*martin.kjellqvist@miun.se

- C. Länkaren kan få bekymmer här.
- D. För minnesoptimerad rekursion.

2. \_\_\_\_\_

3. RAII? (1 alternativ är rätt)

- A. resource acquisition is initialisation
- B. runtime asset initialisation
- C. reduced access instantiation inference
- D. runtime asset instantiation

3. \_\_\_\_\_

4. catch-klausulen som väljs då ett undantag kastas ... (1 alternativ är rätt)

- A. Analyseras och bestäms av kompilatorn under kompileringssteget.
- B. Analyseras och bestäms av länkaren då hela programmet länkas.
- C. Analyseras och bestäms av runtime under runtime.
- D. Väljs deterministiskt av satsblocket där throw-satsen finns.

4. \_\_\_\_\_

5. Vilka av följande påståenden är rimliga anledningar att använda nyckelordet auto?

- A. För att representera en lång känd typ.
- B. För att representera typparametern för en template.
- C. För att slippa onödiga kopior.
- D. Namn för funktioner/lambdas konstruerade med lambdasyntax.

5. \_\_\_\_\_

6. Vilka av följande påståenden är sanna för följande sats?

```
auto s = std::find( v.begin(), v.end(), 0f);
```

- A. Uttrycket `v.begin()` uppfyller kraven för en input-iterator.
- B. Det är ett exempel på en dynamisk typ.
- C. Typen för `s` är `float`.
- D. Typen för `s` är samma som uttrycket `v.begin()`.

6. \_\_\_\_\_

7. Vilka av följande allmänna påståenden är sanna

- A. En funktions-template instansieras under runtime.
- B. `mersenne_twister_engine` finns definierad i `<random>`
- C. Det är vanligt att inte skriva någon programkod i cpp-filer då man skapar templates.
- D. Att använda bra namn på klasser, funktioner, medlemmar och variabler är en central del i att skapa välskrivna programkod.

7. \_\_\_\_\_

8. Vad är sant om typen `std::unique_ptr`?

- A. Den kan vara `nullptr`.
- B. Den kan kopieras.
- C. Den kan flyttas.
- D. Den uttrycker ägande av ett objekt som en pekare.

8. \_\_\_\_\_

9. Uttrycker följande korta snuttar komposition eller klassifikation?

Markera svaren med Komp eller Klass i ordning. Om inget svar är tillämpligt, ange detta med ett streck.

S är en struct, C är en klass.

- A. `struct S: C { };`
- B. `struct S { C c; };`
- C. `class C: S { };`
- D. `class C { S s; };`

9. \_\_\_\_\_

10. Lämpligt funktionsnamn?

Listing 1: Finurlig funktion

```
#include <iostream>

using is = std::istream&&;
using os = std::ostream&&;

// input is formatted numeric.
void F(is A, is B, os C){
    double a, b;
    A >> a;
    B >> b;
    while(A || B){
        if (a < b && A || !B){
            C << a << " ";
            A >> a;
        } else {
            C << b << " ";
            B >> b;
        }
    }
}
```

10. \_\_\_\_\_

Lycka till,  
Martin.