



Försättsblad Prov Original

Kurskod	Provkod	Tentamensdatum
D T 0 9 6 G	T 1 0 1	2 0 1 9 - 0 3 - 2 2
Kursnamn	Datateknik GR (C), Programspråksteori	
Provnamn	Tentamen	
Ort	Sundsvall	
Termin		
Ämne		



Tentamen

dt096g Programspråksteori

Martin Kjellqvist*

2019-03-22

Instruktioner

Läs igenom frågorna noggrant innan du börja besvara dem. Du har begränsat med tid, planera hur du ska besvara frågorna. Besvara endast det som efterfrågas. Skriv inte om saker som inte berörs av frågan.

Skriv svaren på erhållna svarpapper, inte på tentan. Varje ny fråga besvaras på ett nytt svarpapper. Skriv bara på en sida på svarpapperet.

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

Tid 5 timmar.

Hjälpmedel Inga.

Max poäng 50

Antal frågor 11

Preliminära gränser

$E \geq 40\%$, $D \geq 50\%$, $C \geq 65\%$, $B \geq 80\%$, $A \geq 90\%$.

Frågor

- (4p) 1. En viktig egenskap då man utvärderar ett språks uttrycklighet är dess primitivers ortogonalitet. Vad innebär ortogonala primitiver?
- (3p) 2. Vilka designkriterier ligger i fokus för c++? Motivera kort.
- (4p) 3. Beskriv skillnaderna mellan heltalstyper, flyttalstyper och decimaltyper. Ange i vilka typiska situationer varje typ är lämplig.
- (5p) 4. Betrakta följande kodavsnitt

*martin.kjellqvist@miun.se

```

1 int j = 0;
2
3 void f(int i){
4     j = i + 1;
5     std::cout << j + i << "\n";
6     if(std::rand() > RAND_MAX / 5){
7         f(i + j);
8     }
9 }
10
11 int main(){
12     f(4);
13 }

```

Programkörningen använder activation records på det sätt vi beskrivit under kursen. Beskriv hur programkörningen använder activation records.

- (4p) 5. Vilka egenskaper måste ett språk ha för att kallas objektorienterat? Var utförlig, beskriv egenskaperna, gärna med exempel.
- (3p) 6. Vad innebär ett starkt typsystem?
- (3p) 7. Vad gör ett starkt typsystem önskvärt?
8. Betrakta följande kodavsnitt.

```

1     String line = null;
2     File f = null;
3     try{
4         FileReader f = new FileReader("input.txt");
5         line = readLine(f);
6     }catch(FileNotFoundException fne){
7         showMissingFileDialog(fne.getSource());
8     }catch(IOException e){
9         line = "Error in opening or reading file";
10    }finally{
11        closeSilent(f);
12    }

```

- (2p) (a) Vad gör programmet?
- (2p) (b) Vilket är språket? Ange de ledtrådar du ser.
- (3p) (c) Beskriv språket. Ange vilken paradigmen det tillhör och dess ursprung.
- (2p) (d) try/catch-mekanismen kan betraktas som styrsatser. Väljs exekveringsvägen för dettas kodavsnitt under kompilering eller runtime. Beskriv och motivera.
- (2p) 9. (a) Räkna upp några språk som var tidiga i den funktionella paradigmen.
- (4p) (b) Vilka egenskaper måste ett språk ha för att kallas funktionellt? Var utförlig, beskriv egenskaperna, gärna med exempel.
- (5p) 10. Det finns flera språk som inte tillåter funktionsöverlagring, dvs att det får inte finnas två funktioner med samma namn. Varför finns denna restriktion? Vad är svårigheterna med funktionsöverlagring? Ange något språk som tillåter funktionsöverlagring och något som inte tillåter funktionsöverlagring.

(4p) 11. För många programspråk har tillägg gjorts i syntaxen så att man kan uttrycka lambdas.

Vad är lambdas?

Räkna upp 3 programspråk som stödjer lambdas.

Lycka till,
Martin