



Försättsblad Prov Original

Kurskod	Provkod	Tentamensdatum
D T 0 2 8 G	T 1 0 4	2 0 1 9 - 0 4 - 0 2
Kursnamn	Datateknik GR (A), Introduktion till programmering	
Provnamn	Skriftligt teoriprov	
Ort	Sundsvall	
Termin		
Ämne		



Mittuniversitetet

MID SWEDEN UNIVERSITY

Institutionen för informationssystem och -teknologi

Nayeb Maleki

010-142 88 53

Introduktion till programmering (i C++) 6/7.5 hp

Tentamen	Datum: 2019-04-02	Tid: 5 tim.
Hjälpmedel:	Inga!	
Skrivningsanvisningar:	Denna tentamen är ett komplement till ett projektarbete som tillsammans ger ett slutbetyg. Skriv enkelt och <u>tydligt</u> för att undvika feltolkningar. Skriv endast på <u>en</u> sida, max <u>en</u> uppgift per papper. Ange namn och skrivkod på alla sidor. Du får ej använda röd penna! En uppgift med flera delar kan ge poäng utan att alla delar är lösta. Uppgifterna är <u>inte</u> ordnade efter svårighetsgrad.	
Max. poäng	10 (Prel. gränser: 50% and upp för G)	

Lycka till

1 (1p) Vad skrivs ut?

```
cout << ( 2 * 3 == 6 && !(1 > 2) ) << endl;  
cout << ( !(2 * 3 == 6 && 1 > 2) ) << endl;  
cout << ( !(2 * 3 == 6 && 1 != 2) ) << endl;
```

2 (1p) Vilket alternativ visar det korrekta sättet på hur innehållet i variabeln **letter** med deklARATIONEN, **char letter = 'S'**; lagras.

- a) 83
- b) S
- c) 0x53
- d) s

3 (1p) Se kod segmenten nedan (a. och b.) och välj rätt alternativ.

```
a. int one = 1;  
   int two = 2;  
   int six;  
   six = two * 3;  
  
b. int one = 1;  
   int two = 2;  
   int six;  
   two * 3 = six;
```

- a) a. är inte korrekt
- b) b. är inte korrekt
- c) a. och .b båda är inte korrekta
- d) a. och b. båda är korrekta

4 (1p) Vad skrivs ut?

```
#include <iostream>
using namespace std;

int main()
{
    int a = 5;
    int b = 10;
    a += ++b;
    cout << a + b << endl;
    return 0;
}
```

- a) 15
- b) 27
- c) 37
- d) 17

5 (1p) Ange radnummer där felet ligger.

```
1. #include <iostream>
2. using namespace std;
3. void firstFunction(int var);

4. int main()
5. {
6.     int number;
7.     cout << "Enter a number:";
8.     cin >> number;
9.     firstFunction(number);
10.    return 0;
11. }

12. void firstFunction(int var)
13. {
14.     cout << "You are inside of firstFunction." << endl;
15.     return var + 1;
16. }
```

6 (1p) Vad skrivs ut?

```
#include <iostream>
using namespace std;

void receiveVariable(int var);

int main()
{
    int var = 100;
    for (int i = 0; i<5; ++i)
    {
        receiveVariable(var);
    }
    cout << var << endl;
}

void receiveVariable(int var)
{
    var += 10;
}
```

7 (1p) Vilket alternativ deklarerar en referens till variabeln: **int x=256; ?**

- a) int y=x;
- b) char &y=x;
- c) int &y=&x;
- d) int &y=x;

8 (1p) Välj det korrekta alternativet som visar en funktionsprototyp till en funktion med en referensparameter.

- a) int function(int);
- b) int& function(int);
- c) int function(int&);

9 (1p) Given nedanstående funktionsdefinition

```
void calc (int a, int& b) {  
    int c;  
    c = a + 2;  
    a = a * 3;  
    b = c + a;  
}
```

Välj den korrekta utskriften när funktionen calc anropas enligt nedan?

```
int x = 1, y=2, z = 3;  
calc(x, y);  
cout << x << " " << y << " " << z << endl;
```

- a. 1 2 3
- b. 1 6 3
- c. 3 6 3
- d. 1 14 9
- e. Inget av ovanstående

10 (1p) Vad är skillnaden mellan Sequence containers resp. Associative containers?

- a) Båda lagrar elementen i den ordning dem sätts in.
- b) Sequence containers lagrar elementen i den ordning dem sätts in medan Associative containers representera icke-linjär form av datastrukturer och använder nycklar för att komma åt data.
- c) Associative containers lagrar elementen i den ordning dem sätts in medan Sequence containers representera icke-linjär form av datastrukturer och använder nycklar för att komma åt data.
- d) inget av ovanstående