

Tentamen, 2.5 högskolepoäng

Januari 9, 2023, 14:15 - 18.15

Du ska inte använda någon bok, bilder eller online-resurs utom ESP32 Technical Reference Manual. Du kan använda en fickräknare. Du får inte samarbeta med andra studenter via mobiltelefon eller chatt eller att prata. Misstänkta fall av samarbete och plagiering kommer att rapporteras och tentan anses ogiltig.

Du kan svara på svenska eller engelska, men engelska är att föredra. Oroa dig inte om din engelska inte är perfekt, den kommer inte att utvärderas, men svaren måste vara förståeliga.

*Tentamen har 10 frågor med en sammanlagd poäng på 25. Krav för betyg 3 (godkänt) är **15** poäng, krav för betyg 4 är **19** poäng och krav för betyg 5 är **23** poäng.*

Du borde svara med papper och penna. Se till att skriva numret på den fråga du svarar på. Svar som inte kan tolkas på grund av dålig handskrift eller dålig grammatik ger inga poäng.

För frågor, du kan ringa Dario till 0723756073.

Tips: läs igenom HELA tentamen först och lägg upp en strategi för i vilken ordning ni löser uppgifterna – tex de ni tycker är enkla först och de svåra i mån av tid.

Fråga 1 (2pt)

EN

Can you rewrite the following .h file with its "include guards"?

SE

Kan du skriva om följande .h-fil med dess "include guards"?

Filename: accel.h

Content:

```
#include <esp_types.h>

#define MPU6050_ADDR 0x68
#define MPU6050_PWR_MGMT_1 0x6B
#define MPU6050_SMPLRT_DIV 0x19

struct Accel
```

```
{  
    int16_t x;  
    int16_t y;  
    int16_t z;  
};  
  
void initAccel(int sda_pin, int scl_pin);  
  
struct Accel getAcceleration();
```

Answer:

Add

```
#ifndef ACCEL_H_  
#define ACCEL_H_
```

At the beginning and

```
#endif
```

At the end.

Fråga 2 (2pt):

EN

How can you define a struct with 3 unsigned integers fields, where the first two fields, named A and B, take 1 bit each and the third, C, takes 6 bits?

SE

Hur kan du definiera en struktur med 3 heltalsfält utan tecken, där de två första fälten, benämnda A och B, tar 1 bit var och det tredje, C, tar 6 bitar?

Answer:

```
struct  
{  
    unsigned int A : 1;  
    unsigned int B : 1;  
    unsigned int C : 6;  
} mystruct;
```

Fråga 3 (2pt):

EN

Describe a software solution to avoid the “bouncing” effect in an ISR.

SE

Beskriv en mjukvarulösning för att undvika den "bouncing" effekten i en ISR.

Answer:

Using a code such as:

```
PUSH_TIME = reasonable amount of time needed to push
    counter = 0;
lastPush = -PUSH_TIME;

ISR_handler()
{
    now = get_time();
    if (now - lastPush > PUSH_TIME_US)
    {
        lastPush = now;
        counter++;
    }
}
```

Fråga 4 (3pt):

EN

Your company develops a chip for 5G communication. Internally, it uses a microcontroller with a single CPU running at 128MHz, programmed with a real time operating system. Your main customer, a major mobile phone manufacturer, is receiving complaints from customers that tell that the connection to the network sometimes fails and that they need to reset the phone. You check the code running in the chip and you identify 4 periodic tasks using Rate-monotonic Scheduling:

Task1:

lines of code: 220 kloc (thousands of lines of code)

period: 100 ms

Task2:

lines of code: 10 kloc

period: 5 ms

Task3:

lines of code: 120 kloc

period: 50 ms

Task4:

lines of code: 850 kloc

period: 250 ms

Can you identify why the communication fails?

Hint: pay attention to the unit of measurement

SE

Ditt företag utvecklar ett chip för 5G-kommunikation. Internt använder den en mikrokontroller med en enda CPU som körs på 128MHz, programmerad med ett realtidsoperativsystem. Din huvudkund, en stor mobiltelefonföretagare, får klagomål från kunder som berättar att anslutningen till nätverket ibland misslyckas och att de behöver återställa telefonen. Du kontrollerar koden som körs i chippet och du identifierar 4 periodiska uppgifter (task) med hjälp av Rate-monotonic Scheduling:

Task 1:

kodradar: 220 kloc (thousands of lines of code)

period: 100 ms

Task 2:

kodradar: 10 kloc

period: 5 ms

Task 3:

kodradar: 120 kloc

period: 50 ms

Task 4:

kodradar: 850 kloc

period: 250 ms

Kan du identifiera varför kommunikationen misslyckas?

Tips: var uppmärksam på måttenheten

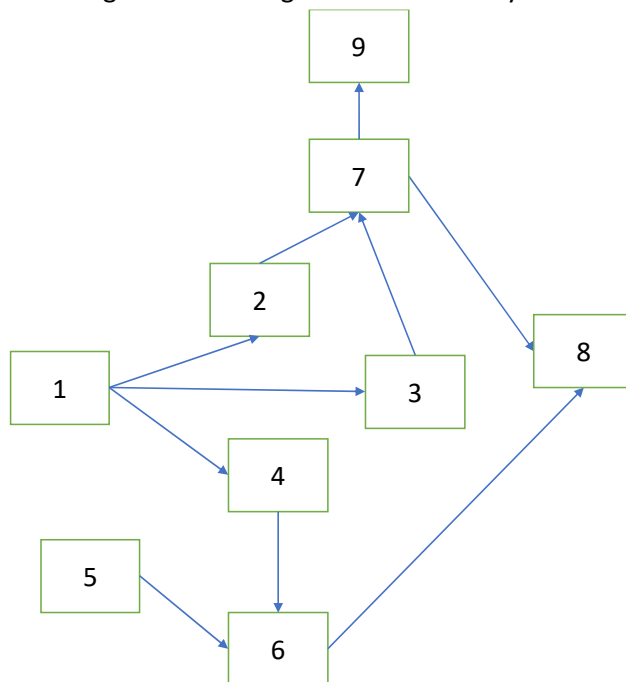
Answer:

The utilization factor is 0.78, but, if RMS is used, according to the Utilization bound test, it should not be more than 0.76.

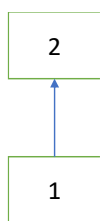
Fråga 5 (3pt):

EN

Rearrange the following modules into a layered architecture.



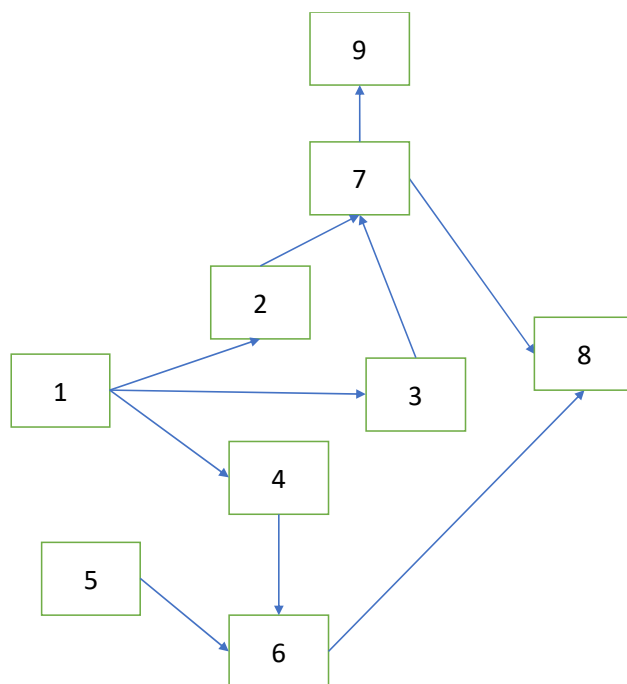
Consider that the arrow means “offers services to”, for example in:



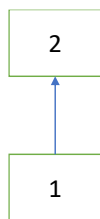
The block 2 depends on block 1.

SE

Ordna om följande moduler till en skiktad arkitektur.



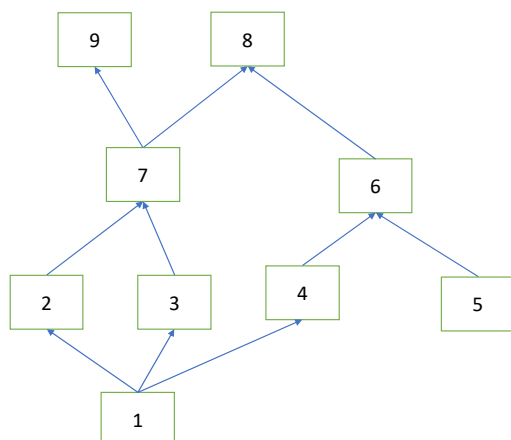
Tänk på att pilen betyder "erbjuder tjänster till", till exempel i:



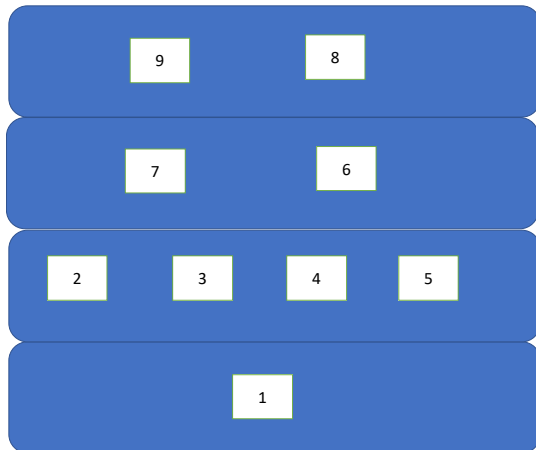
Block 2 beror på block 1.

Answer:

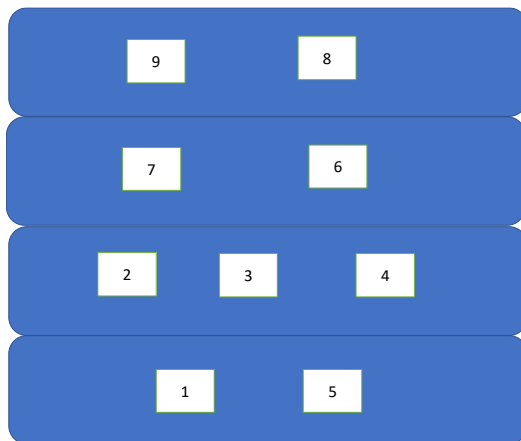
Making sure that blocks that depend on others stay on top, you can re-organize the blocks vertically like:



From which it is easy to derive the layers, like:



This could also be valid (although a bit stretched):



Fråga 6 (3pt)

Beakta följande funktioner som alla beror på n .

- $f(n) = n^2 + n$
- $g(n) = 3n^3 \cdot 2n^2$
- $h(n) = 4n^2 + 5$
- $i(n) = 12n + 22$

- Vilken (eller vilka) av funktionerna är $O(n^2)$? Motivera kortfattat ditt svar. (1pt)
- Vilken (eller vilka) av funktionerna är $\theta(n^2)$? Motivera kortfattat ditt svar. (1pt)
- Vilken (eller vilka) av funktionerna är $\Omega(n^2)$? Motivera kortfattat ditt svar. (1pt)

Svar:

a. $f(n), h(n), i(n)$. $g(n)$ växer fortare än något som är proportionellt mot n^2 .

- b. $f(n)$ och $h(n)$. $g(n)$ växer fortare och $i(n)$ långsammare än något som är proportionellt mot n^2
c. $f(n), g(n), h(n)$. $i(n)$ växer långsammare än något som är proportionellt mot n^2 .

Fråga 7 (3pt)

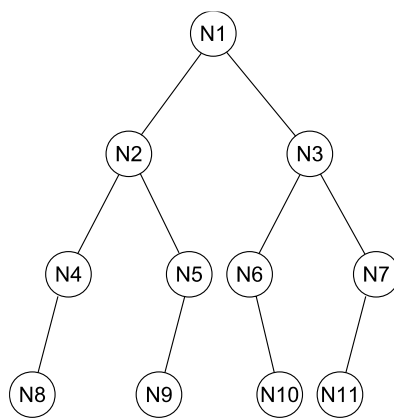
Merge är den grundläggande operationen i algoritmen Mergesort, och den används för att "sammanfoga" två sorterade listor till en sorterad lista. I värsta fall krävs det $2n - 1$ jämförelser för att slå samman två sorterade listor som innehåller n element vardera till en sorterad lista med $2n$ element. Förklara när detta "värstafall" inträffar och ge ditt eget exempel som illustrerar detta fall.

Svar: Värstafallet inträffar när det största elementet och det näst största elementet av de två listorna inte finns i samma lista.

Till exempel innehåller den första listan värdena 1, 2, 3, 7 och den andra listan innehåller värdena 4, 5, 6, 8.

Fråga 8 (2pt)

Ordna noderna i följande sökträd från lägst till högst.



Svar: $N8 < N4 < N2 < N9 < N5 < N1 < N6 < N10 < N3 < N11 < N7$

Fråga 9 (2p)

Hur visar man att ett problem P är \mathcal{NP} -fullständigt (Eng: \mathcal{NP} complete)? (2p)

Tips: Man behöver göra två saker.

Svar:

1. Visa att P är i \mathcal{NP}

2. Identifiera något annat problem P' som man vet är \mathcal{NP} -fullständigt, och visa att P' kan bli polynomiskt reducerat till P .

Fråga 10 (3p)

Visa med ett eget exempel hur en hierarkisk finit tillståndsmaskin (Eng: Hierarchical FSM) fungerar.

Answer: See lecture slides for an example.