

1 Assignment 1

1.1 Question 1

Svaret borde ges genom att beräkna $0x5b - 0x30 = 0x2B = 43_{10}$

1.2 Question 2

Man klarar sig med att ändra på 3 lines. Jag ändrade från:

```
addi      $s0,$s0,1

li        $t0,0x5b
bne       $t0,$0,loop

till

addi      $s0,$s0,3

li        $t0,0x5b
slt       $t1,$s0,$t0
bne       $t1,$0,loop
```

2 Assignment 2

2.1 Question 1

This is my code:

```
hexasc:
    andi    $v0, $a0, 15
    slti    $t0, $v0, 10
    beq     $t0, $0, bokst
    addi    $v0, $v0, 0x30
    jr      $ra
bokst:
    #sub     $v0, $v0, 10
    addi    $v0, $v0, 0x37
    jr      $ra
```

2.2 Question 2

Kan visa om det behövs.

2.3 Question 3

The answer must be when the input is in the form of $16 * n + 10 \in N$ Det finns många värden som är möjliga. T.ex. resulterar värdena 10 och 26 i rätt resultat.

3 Assignment 3

3.1 Fråga 1

time2string hade en adress där den skulle lagra informationen, istället för ett register. Detta underlättade då man behövde spara värden vid flera tillfällen.

3.2 Fråga 2

Jag använde PUSH och POP för att försäkra mig om att värdena på \$s0 och \$s1 blev återställda till deras ursprungliga värde efter funktionen hade kört klart.

3.3 Fråga 3

Jag sparade värdet på första \$ra i ett annat, vanligt register, \$s2, och sedan återskapade jag det när det behövde det. Man hade också kunna använt PUSH och POP för att spara det i början för att sedan hämta det i slutet.

3.4 Fråga 4

Jag tror att det kommer förstöra koden då *words* sparar saker med 4 register i taget. Kanske tillåter kompilatorn inte det på än gång.

4 Assignment 4

4.1 Fråga 1

Är värdet noll kommer koden bryta när den gör checken i whileloopen. Den kommer aldrig komma in i for loopen och väldigt kort tid går.

4.2 Fråga 2

Alla delar utav koden skulle ske. For loopen skulle loopas till sitt maxantal var 2 gånger innan indexet för whileloopen skulle vara lika med noll och loopen skulle lägga ner.

4.3 Fråga 3

Om inputten är ett negativ nummer kommer delay lägga av efter under 10 klockcyklar vilket tar 10^{-8} sekunder på en 1 GHz processor.

5 Assignment 6

Det är stora skillnader mellan att simulera koden på en dator som inte är byggd för den, och att faktiskt köra koden på en processor som använder sig av den här arkitekturen. Skillnaden är att de helt enkelt inte kör en klockcykel på samma tid, och därför måste *delay* ändras för att kompensera.