

Laboration 3 – Tärningsspel

Laboranter:

Namn : Anna Selstam
Datorid : am3963

Namn : Malin Ramkull
Datorid : am5914

Datum då laborationen genomfördes: 2021-12-02



(<https://ak.picdn.net/shutterstock/videos/739930/thumb/1.jpg>)

Genom att skicka in labbrapporten intygar du/ni att följande regler har följts:

1. Laborationsuppgifter skall lösas självständigt av varje laborationsgrupp. Det är tillåtet att diskutera lösningar, men INTE att kopiera lösningar! Det är alltså INTE tillåtet att ge laborationsresultat eller färdiga lösningar till en annan grupp.
2. Bägge gruppmedlemmarna förväntas ta aktiv del i genomförandet av laborationen och skrivandet av rapporten. Detta inkluderar att bygga, programmera, dokumentera, testa och felsöka. Bägge gruppmedlemmarna skall kunna svara på frågor om hur laborationen genomförts och vilka resultat som erhållits.
3. Examination baseras alltid på individuella resultat

Resultat

Uppgift 4.2.2 (redovisas i rapport)

Förklara vad den rad vi lade till i koden i uppgift 4.2.1 innebär och varför innehållet mellan "" är organiserat som det är ovan.

Svar: DB står för "define bytes". Detta kommando tillåter oss att definiera strängar av bytes. I vårt fall står informationen inom citationstecken "147*2580369#" för respektive tangenter på tangentbordet som används.

Uppgift 4.2.4 (redovisas i rapport)

Förklara vad kod-raderna i uppgift 4.2.3 gör. INTE vad de olika instruktionerna betyder, utan **varför** de behövs!

Svar: När vi initierar en pekare där vi vill lagra våra objekt, kan det ske att ZL kan gå över byte gränsen dvs att den går över 255. Därför laddar vi RVAL med 0 och adderar det till ZH mha operationen ADC som då kommer ta med sig en carry från föregående operation - om det finns en sån, och på så sätt fångar vi upp vår carry som annars inte hade kommit med. Efter att både ZL och ZH är satta till sitt rätta värde kan LPM sen hämta datan.

Uppgift 5.2.4 (redovisas i rapport)

Förklara vad de rader vi lade till i koden i uppgift 5.2.1 och 5.2.2 gör. Vad innebär "+" efter Z?

Svar: .DB definierar en sträng "Welcome!" av bytes, antal chars för den definierade strängen sätts av .EQU på raden under till 8. "+"- tecknet innebär att vi flyttar Z-pekaren till nästa lagrade värde i flashminnet (fungerar som instruktionen INC Z).

Uppgift 5.2.5 (redovisas i rapport)

I exemplet använder vi en extra konstant som talar om hur många tecken som ingår i strängen. Det finns ett annat sätt att hantera strängens slut. Googla på tex: "strings in C" och beskriv kortfattat (kod behövs inte) hur man skulle kunna göra istället. Även kod och exempel i föreläsningsbilder kan vara till hjälp.

Svar: I C representeras strängar av arrays med chars. Man kan själv definiera antal tecken för sin sträng genom .EQU och ange antal chars. Ett annat alternativ för att ange storleken på sin definierade sträng är att ange en speciell char, 0, efter sin sträng, som markerar slutet.

Uppgift 8.1.2 (redovisas i rapport)

Antal kast:	100 (0x64)	Procent %
Antal '1'	11 (0x0B)	11%
Antal '2'	19 (0x13)	19%
Antal '3'	19 (0x13)	19%
Antal '4'	17 (0x11)	17%
Antal '5'	18 (0x12)	18%
Antal '6'	16 (0x10)	16%

Uppgift 8.1.3 (redovisas i rapport)

Redogör för era erfarenheter från denna laboration. Vad har ni lärt er? Gick allting bra eller stötte ni på problem? Om allting gick bra, vad var i så fall anledningen detta? Om ni stötte på problem, hur löste ni i så fall dem? Skriv gärna var sin reflektion.

Anna: Denna laboration var svår, lång och utmanande. Vi återkom i 2 långa tillfällen för att bli klara med den, och tillslut gick det. Förvånansvärt var, hur det i mitten av tiden vi jobbade på denna, plötsligt verkade klicka för båda av oss. Tidigare tillfällen hade upplevts väldigt förvirrande och överväldigande men nu var det som att vi kunde strukturera våra tankar och idéer och implementera dem på ett logiskt sätt. Det var otroligt kul att inse detta, trots att det såklart fortfarande var en väldigt stor utmaning att klara av labben. Något annat som var roligt var att föreläsningarna upplevdes knytas samman i takt med förståelsen och lärdomarna från laborationerna.

Malin: Detta var, utan tvekan, den mest utmanande laborationen hittills. Vi förberedde oss, främst teoretiskt, inför labbtillfället denna gången.

Väl på plats flöt det på bra fram till moment 7. Här blev kod-implementationen klurigare. Vi testade många olika lösningar på de problem vi stötte på. Efter mycket om och men lyckades vi lösa en fungerande "store_stats" rutin, som var kärnan i problemen. Efter detta löste sig även resterande moment i laborationen.

Trots att laborationen var utmanande och att vi stötte på stora problem så kände man hur inlärningskurvan har rört sig betydligt uppåt, om man ser tillbaka på lab 1, vilket är kul.