

Experiment 8: Random Number Generation

14.12.2018

Res. Asst. Abdullah Cihan Ak
akab@itu.edu.tr

Eğer bu parayı havaya atarsam bunun yazı ya da tura gelmesi şansa bağlı değil mi?

Nava başını salladı.

İşte burada yanılıyorsun. Eğer bir parayı fırlattığımda bunu etkileyen tüm fiziksel faktörleri hesaplayabilseydik, örneğin elimin açısı, yerden yüksekliği, parayı fırlatmak için ne kadar güç kullandığım, rüzgâr veya hava akımı, paranın almasını falan gibi, o zaman yazı mı tura mı geleceğini yüzde yüz bilebilirdin. Çünkü bu para da, diğer her şey gibi, Newton'un mutlak olan fizik kurallarından etkileniyor.

Nava bu sözleri düşünürken bir sigara daha yakmak için durdu. Belki tüm bunlar beni aşırıydı, ama David tüm bunları doğru hesaplamak olanaksız değil mi?

İnsanlar için öyle, dedi Caine. Ama sırf biz faktörleri hesaplayamıyoruz diye bu parayı attığımda ne geleceğinin şansa bağlı olduğunu söyleyemeyiz. Bunun anlamı şu: Biz insanlar evrenin belli gerçeklerini ölçebilecek becerilere sahip değiliz. Yani, olaylar her ne kadar rastgele görünse de, tamamen fiziksel gerçeklerle koşullandırılmışlardır ve böyle belirlenirler.

Böyle düşünenlerin akımına Determinizm denir. Deterministler hiçbir şeyin belirsiz olmadığına inanırlar; her şey önceki bir sebebin sonucu olarak ortaya çıkar ama biz bu sebebin ne olduğunu bilemeyiz.

Olasılıksız (2004), Adam Fawer

1 Introduction

In the following experiment, you are going to implement a random number generator. Note that this is the last experiment and we are expecting a better coding such as using subroutines with operands transferred using stack. You will not get much of help for the syntax in this experiment.

Experiment 8: Random Number Generation

2 Part 1

In this part, you should implement a random number generator using "Middle Square Weyl Sequence" approach. Please check the following *Middle Square Weyl Sequence* algorithm and implement it as an interrupt subroutine. Use P2.5 as button to interrupt the microcomputer to generate a new random number between 0-128 and show it on the 7-segment display.

```
1 x = 0, w = 0, s = any seed
  x = square(x);
3 x = x + (w = w + s);
  r = (x>>4) | (x<<4);
```

3 Part 2

In this part, you are going to use the Random Number Generator in Part1. Use P2.5 as button to interrupt the microcomputer to generate 128 random number between 0-8(8 is not included) and write it into preallocated memory locations. Check that how much the distribution of the random numbers generated from your random number generator close uniform. You can use counting to check the uniformity. Show your work on the memory using preallocated memory locations. Show your work with different amount of random numbers and with different intervals of random numbers.

4 Part 3 (optional)

For extra credits implement MSE (Mean Squared Error) to analyze the uniformity of the distribution.