DIA SUMMARY

We have:

* 2 items to sell (infinite quantities available): FIRST and SECOND
* 4 classes of costumers: C\_i, i=1,..4  
  For each class the daily number of costumers that enter the shop, N\_i, is determined by a (truncated) Gaussian probability distribution
* C\_i\_FIRST: Given the price of FIRST, we have a conversion rate for each class i (Probability that a customer buys the first item). It depends on seasonality and on the price of FIRST
* C\_i\_SECOND: If the customer buys the first item, he can decide whether to buy the second or not. We have the according conversion rate, which depends on seasonality as well as on the price of SECOND and on the PROMO offered
* 4 different possible promos for SECOND, which corresponds to 4 different levels of discount (the professor suggested to use additive promos, not percentages. So if the first item costs 40 $, possible discounts may be: -0$, -5$, -10$, -15$)
* What we earn from each sale is a MARGIN, so the price at which the object was sold minus production costs

We have no data to work with, instead we have to come up with some (realistic ?) data ourselves.

Step 1.

* Pick 4 different prices for the FIRST item
* Pick 4 different prices for the SECOND item
* For each combination of (p\_FIRST, p\_SECOND), so 16 combinations, solve a bipartite matching problem to assign promos to costumers

Step 2.

Only theoretical, no code needed. We have to identify the random variables and propose a suitable distribution, for example:

* Daily number of costumers for each class: Gaussian distribution
* Conversion rates: Bernoulli distributions Be(p), where p is a function of the class, seasonality, price of the item

Step 3.

Among the 4 possible prices of the FIRST item we picked in step 1 we have to find the optimal one

This basically boils down to two for cycles:

for (each day)

for (each customer)

simulate feedback:

* Be(p\_FIRST) -> 1 buys first / 0 does not buy

If he bought the first item

* Be(p\_SECOND) -> 1 buys second / 0 does not buy

Using the generated feedbacks run a UCB1 or a TS