Getting Start with a Game

- Now we use a Game as a starter
- Suppose you joint such a game:
  1. There are 40 persons in the game, 39 counterparties and you
  2. In each time, two person (20 pairs) will trade with each other
  3. You and your counterparties both have two options:
     - Trade, or says, trust.
     - Cheat, or says, betray.

     Once both persons choose his/her option, calculate the points he/she get as the table

		A	
		trust	betray
B	trust	A: +10; B: +10	A: +2X; B: -X
	betray	A: -X; B: +2X	A: -Y; B: -Y

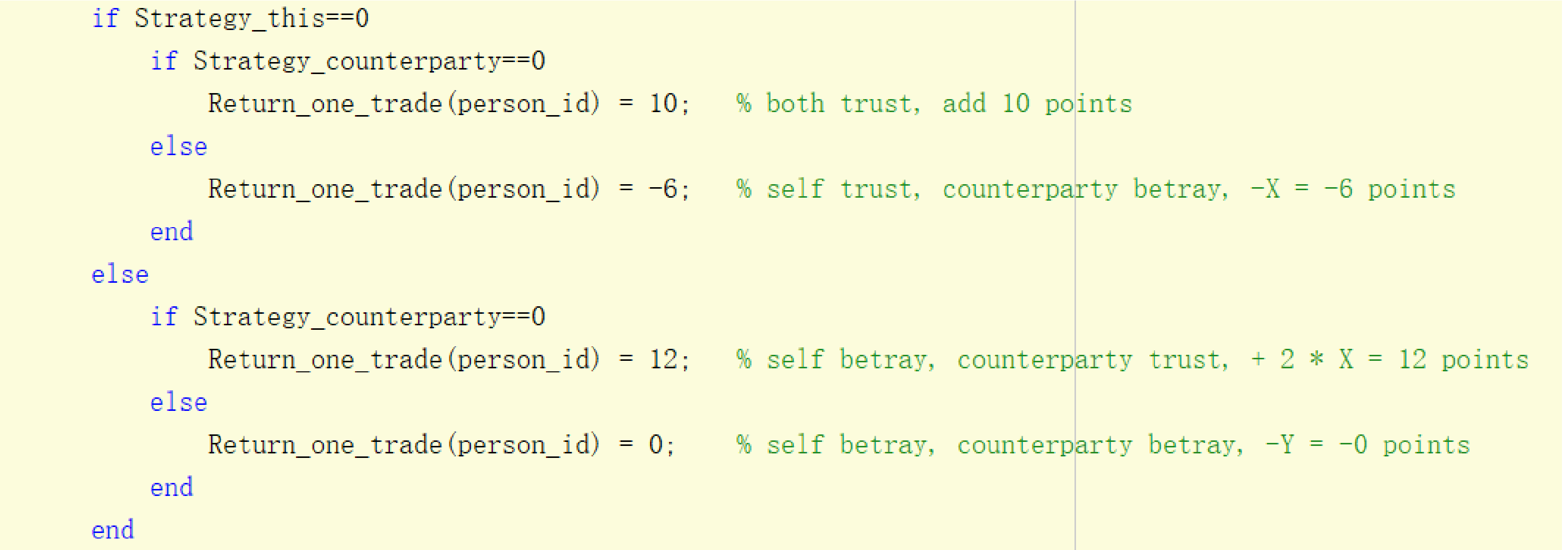
Getting Start with a Game

- Encourage trust and trade: win-win

- The social reality is: if one get betrayed, he will loss something, i.e. money; the person who betray him will gain something. (-X, +2X)

- In some cases, if both person betray each other, they will loss something, i.e., time, reputation, … (both -Y)

		A	
		trust	betray
B	trust	A: +10; B: +10	A: +2X; B: -X
	betray	A: -X; B: +2X	A: -Y; B: -Y



```
if Strategy_this==0
    if Strategy_counterparty==0
        Return_one_trade(person_id) = 10;   % both trust, add 10 points
    else
        Return_one_trade(person_id) = -6;   % self trust, counterparty betray, -X = -6 points
    end
else
    if Strategy_counterparty==0
        Return_one_trade(person_id) = 12;   % self betray, counterparty trust, + 2 * X = 12 points
    else
        Return_one_trade(person_id) = 0;    % self betray, counterparty betray, -Y = -0 points
    end
end
```

Getting Start with a Game

		A	
		trust	betray
B	trust	A: +10; B: +10	A: +2X; B: -X
	betray	A: -X; B: +2X	A: -Y; B: -Y

- Now we use a Game as a starter
- Suppose you joint such a game:

1. There are 40 persons in the game, 39 counterparties and you
2. In each time, two person (20 pairs) will trade with each other
3. You and your counterparties both have two options:
    - Trade, or says, trust.
    - Cheat, or says, betray.

    Once both persons choose his/her option, calculate the points he/she get as the table
4. Repeat 2-3 N times, here we use N = 100 first. In each time, the pairs are randomly settled.

Now, let’s go to the program

Getting Start with a Game

- Pre-set Strategies:
  - 13 always trust (ID1)
  - 13 always betray (ID11)
  - 13 trust one time and then betray one time and go on (ID21)
    - Therefore, this Strategy should remember how many times he/she had already traded, or, what action he/she had used in the last time.
    - Here we use ‘remember how many times he/she had already traded’ as an example
  - 1 Society Revenger (ID31)
    - If he/she get trusted last time, trust the person this time
    - If he/she get betrayed last time, betray the person this time
    - Therefore, the information of what action one person faced in last time will be provided via an .mat file.

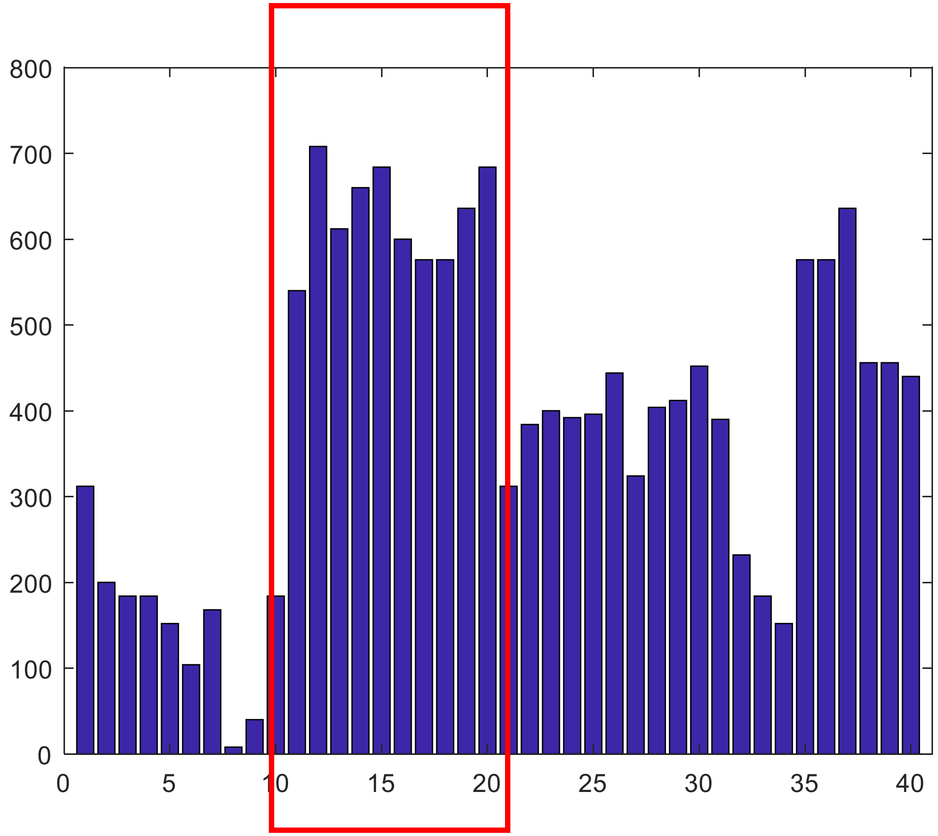
Getting Start with a Game: engineering basic

- Some parts of the program are very stupid, for example:
  - Every one has the right to storage and read files
  - …
- However, the basic prototype of this game can be said as ‘well developed’
- And also, this mini prototype shows the way to make lots of persons to cooperate with each other:
  - One write the whole framework, and design what and how others are written by who
  - Other persons finish his part following the specifications (规范)
- For a research person, ‘well developed’ prototype is very important: if it is valuable, you can improve it latter given time and money

Some interesting testing

- Pre-set Strategies:
  - 13 always trust (ID1)
  - 13 always betray (ID11)
  - 13 trust one time and then be
    - Therefore, this Strategy shoulc what action he/she had used in
    - Here we use 'remember how r
  - 1 Society Revenger (ID31)
    - If he/she get trusted last time, '
    - If he/she get betrayed last time
    - Therefore, the information of ' via an .mat file.

Betrayers win

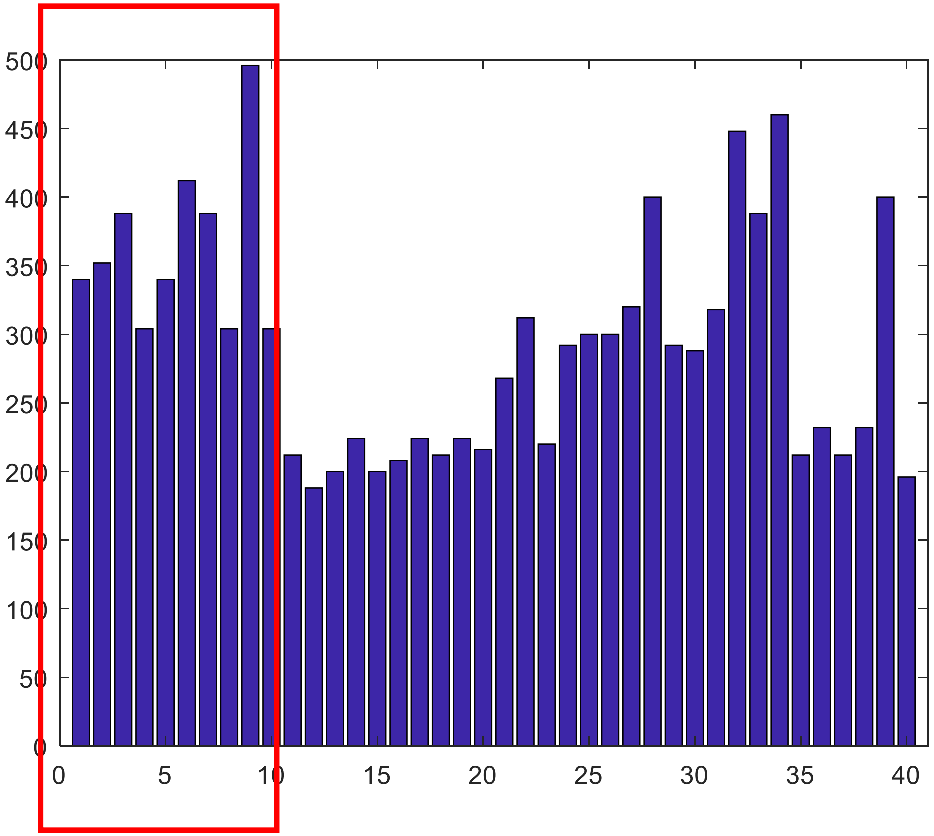


Under X=6, Y=0

		A	
		trust	betray
B	trust	A: +10; B: +10	A: +2X; B: -X
	betray	A: -X; B: +2X	A: -Y; B: -Y

Some interesting testing

- Pre-set Strategies:
  - 13 always trust (ID1)
  - 13 always betray (ID11)
  - 13 trust one time and then be
    - Therefore, this Strategy shoulc what action he/she had used ir
    - Here we use ‘remember how r
  - 1 Society Revenger (ID31)
    - If he/she get trusted last time,
    - If he/she get betrayed last time
    - Therefore, the information of via an .mat file.



Good persons win

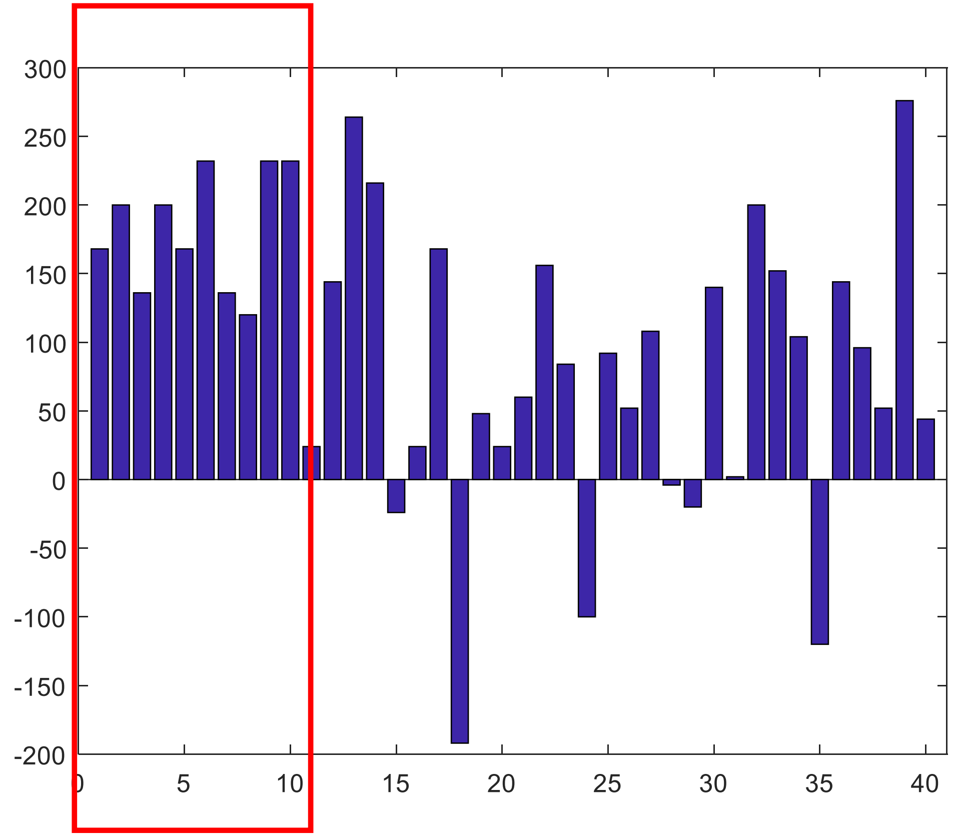
Under X=2, Y=0

		A	
		trust	betray
B	trust	A: +10; B: +10	A: +2X; B: -X
	betray	A: -X; B: +2X	A: -Y; B: -Y

Some interesting testing

- Pre-set Strategies:
  - 13 always trust (ID1)
  - 13 always betray (ID11)
  - 13 trust one time and then be
    - Therefore, this Strategy shoul what action he/she had used i
    - Here we use ‘remember how
  - 1 Society Revenger (ID31)
    - If he/she get trusted last time,
    - If he/she get betrayed last time
    - Therefore, the information of via an .mat file.

Good persons win
in the sense of probability



Under X=6, Y=12

		A	
		trust	betray
B	trust	A: +10; B: +10	A: +2X; B: -X
	betray	A: -X; B: +2X	A: -Y; B: -Y