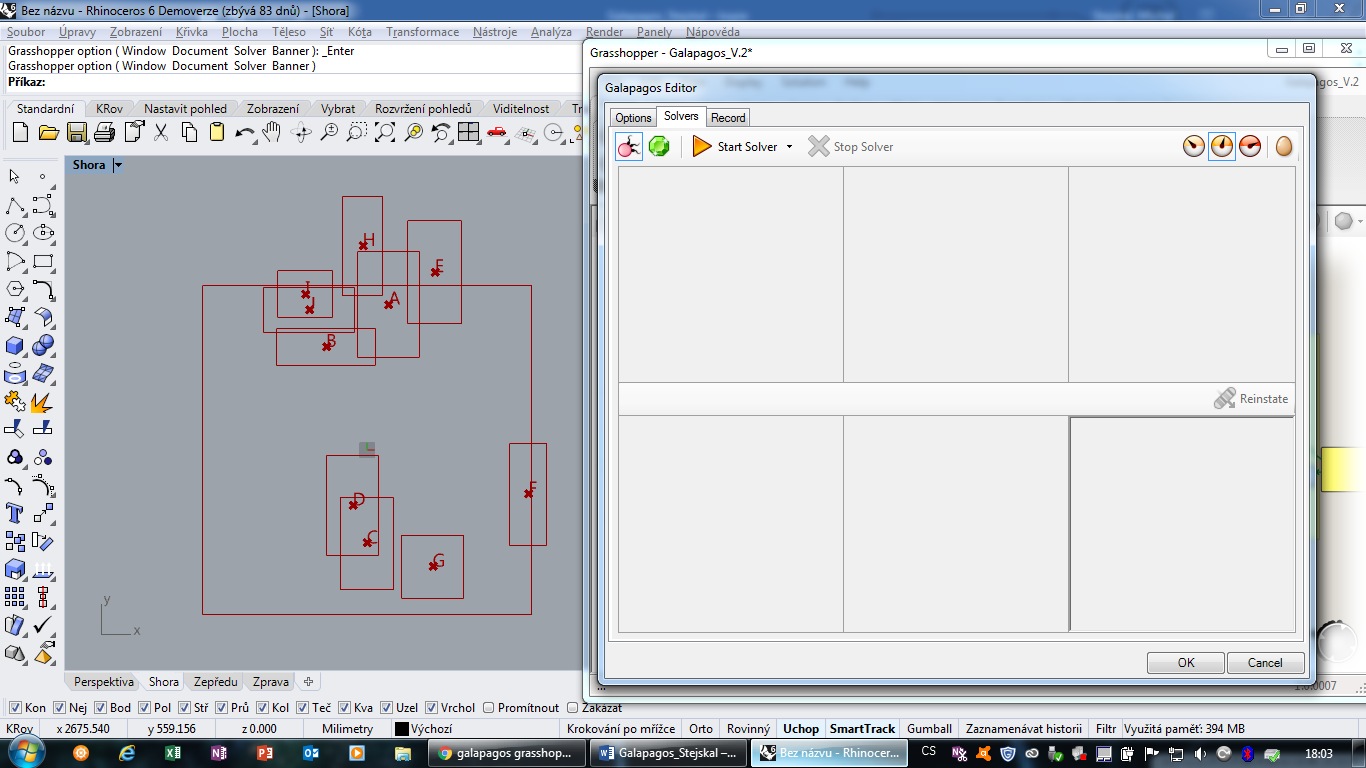
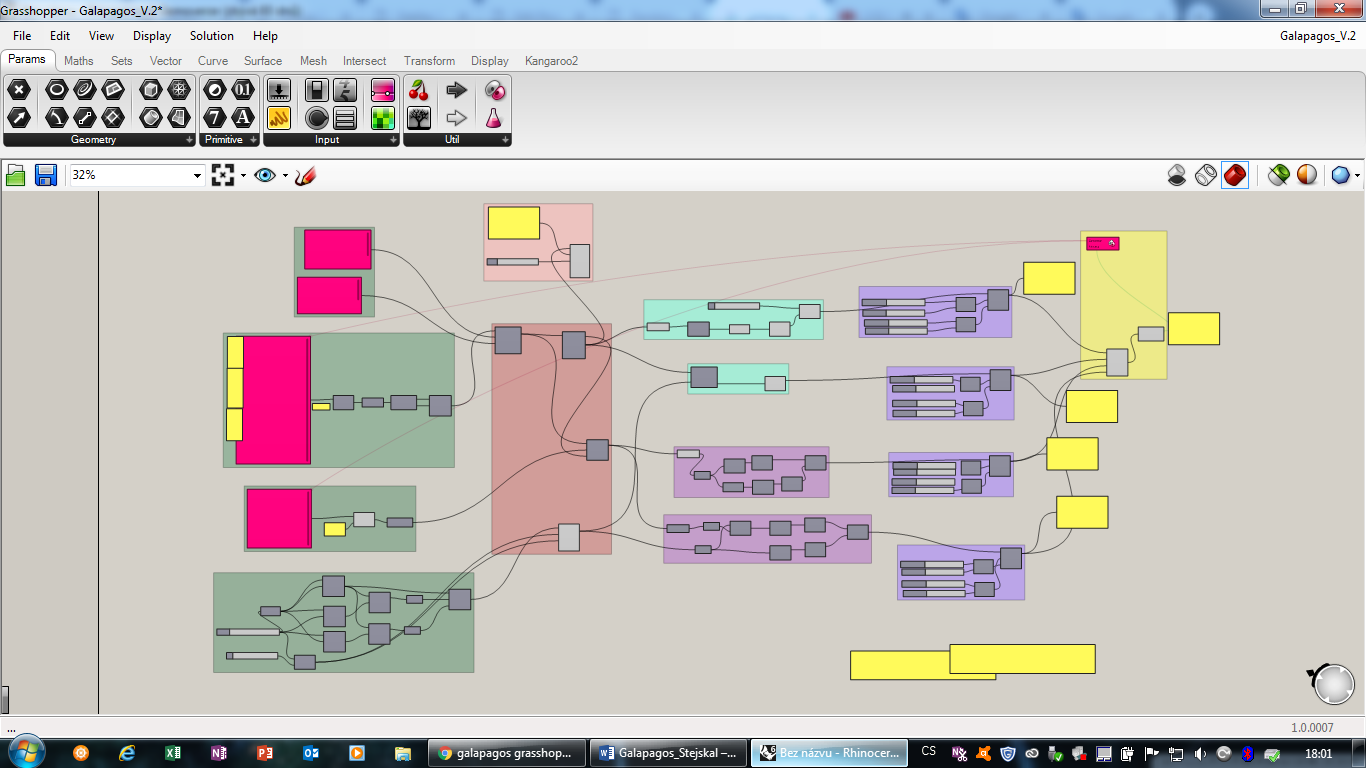
This semester I have worked with Galapagos. Galapagos is evolutionary solver which solving problems with sliders. Basically, you put data in it and set up Maximize or Minimize to get solution which contains the lowest or the biggest value. Galapagos set your sliders to the suited position for selected solution. All work is done in grasshopper so there is no .3dm file attached.

I put random objects into bigger one. Then I tried to move them with program to be near the outside line of bigger rectangle plus to be how far from each other how even is possible. I created two conditions. Object should not overlap others and objects must stay in specified rectangle.

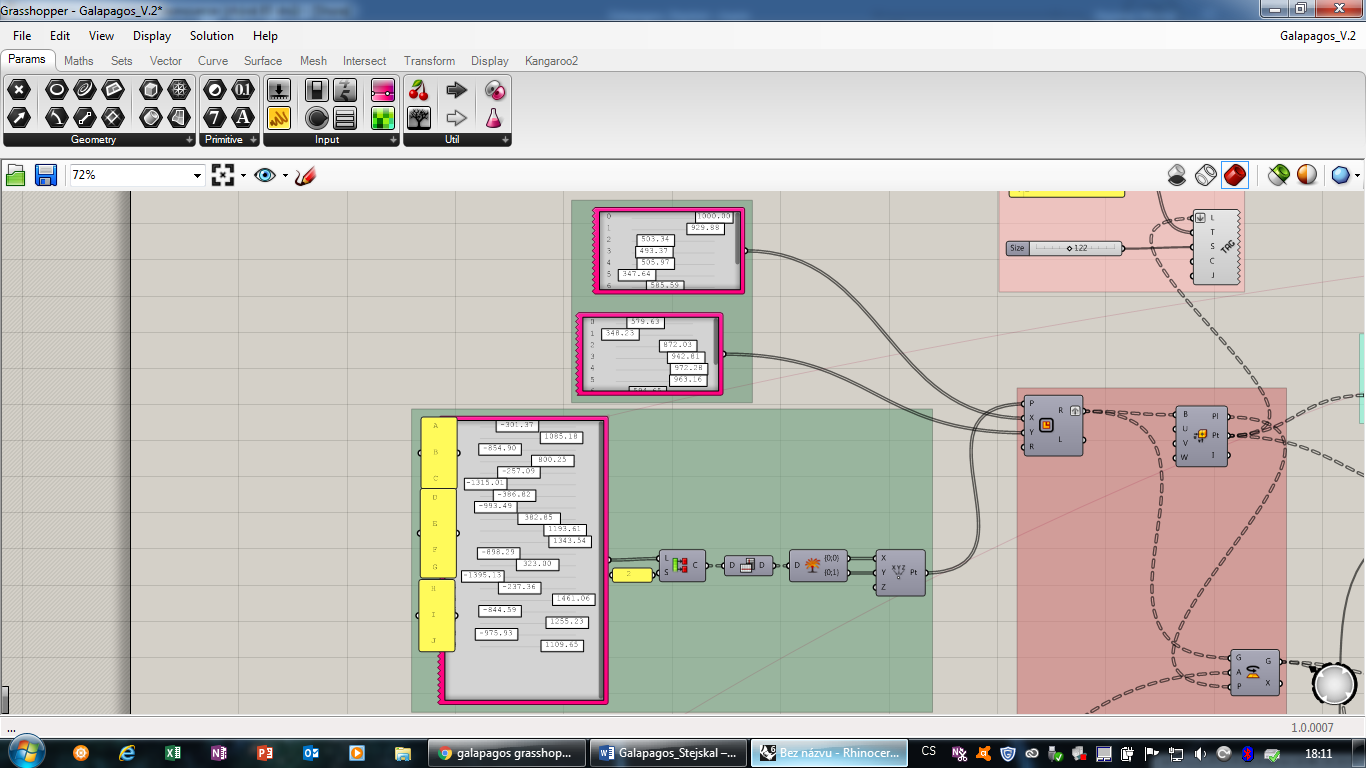


For tutorial I have divided my script by colours. Green for part for input data. Red for collected data. Turquoise for problems which should be solved. Purple for conditions. Blue for remapping numbers. Yellow for Galapagos. Pink is used for part which create designation for created rectangles but is not important for function of program.



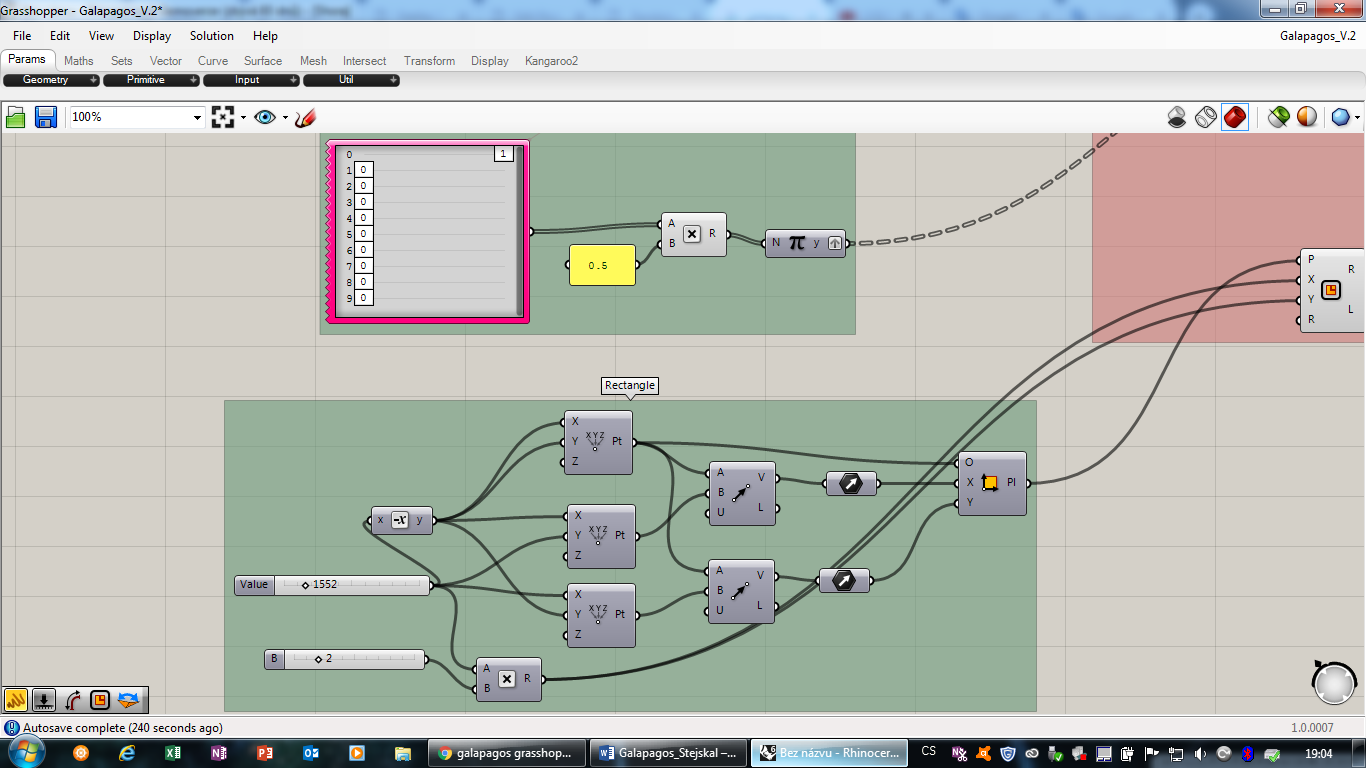
1. I created sizes of rectangles, points to be able to move with them. Part which is used for rotation of rectangles and dimensions of main rectangle. (Green part)
2. Here I collected previous data and created all rectangles. Put point in middle of them and allowed them to rotate. (Red part)
3. I name objects (Pink part)
4. I created polyline from middle points and set that the longer it is the lowest number will be output. Then I created point on main rectangle which are nearest point to the each middle point of smaller rectangles. (Turquoise part)
5. I made two conditions. One for overlap smaller rectangles and second to keep them in main rectangle (Purple)
6. I remapped all outputs number to set priorities. (Blue part)
7. Finally I summed output data and connected them to Galapagos together with Gene pools for rectangles coordinates and their rotation.

Green part



Upper part creating sizes of rectangles. I used Gene pool because it is easier then create many sliders.

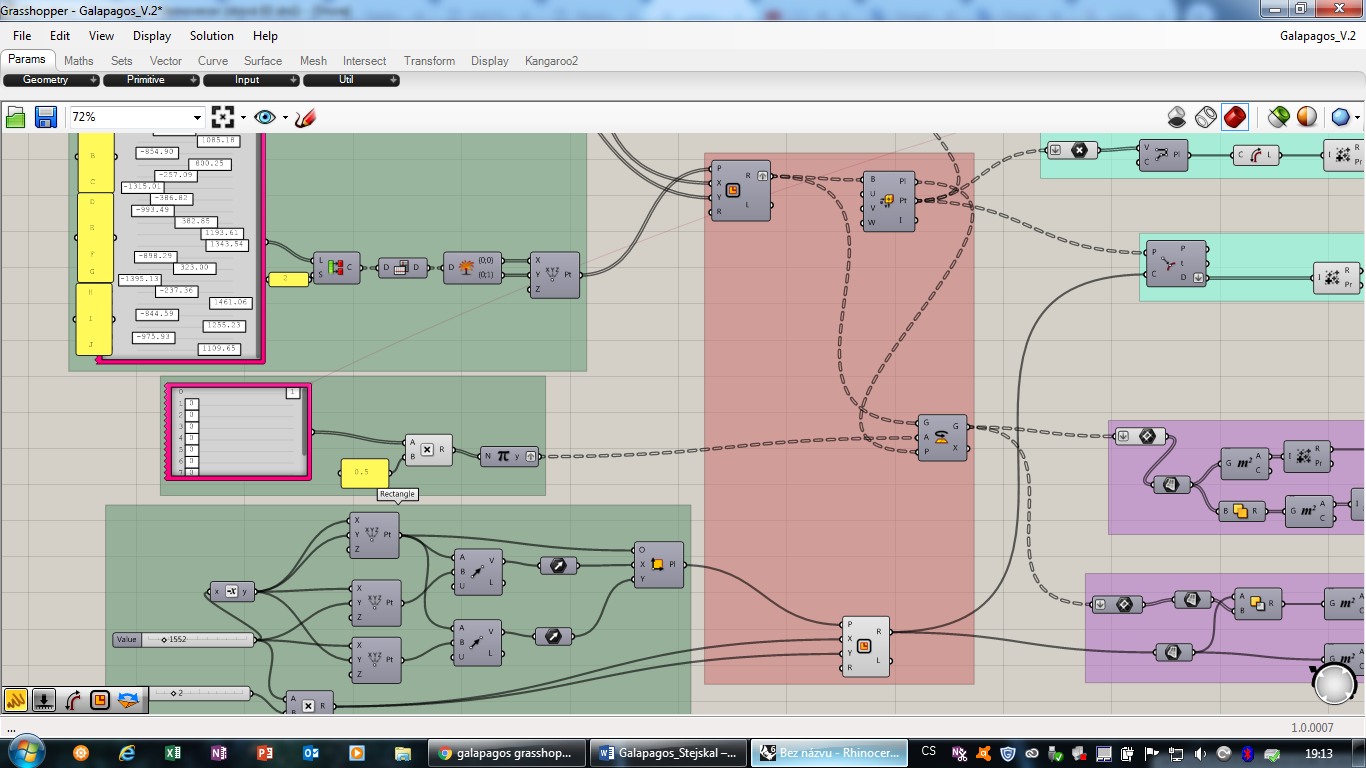
Second part create coordinates. I used only one Gene pool, so I have to divided it into two branches of data tree. Firstly, I used Partition list and input 2 because that is the number of coordinates I needed. Then by Flip matrix I divided them into two columns every one of them has different y coordinate so when I used Explode tree it made two outputs one for x a and one for y coordinate.



Upper one is used to determinate which object and how will rotate. I needed to turn object around a quarter. Which mean I used half a pi.

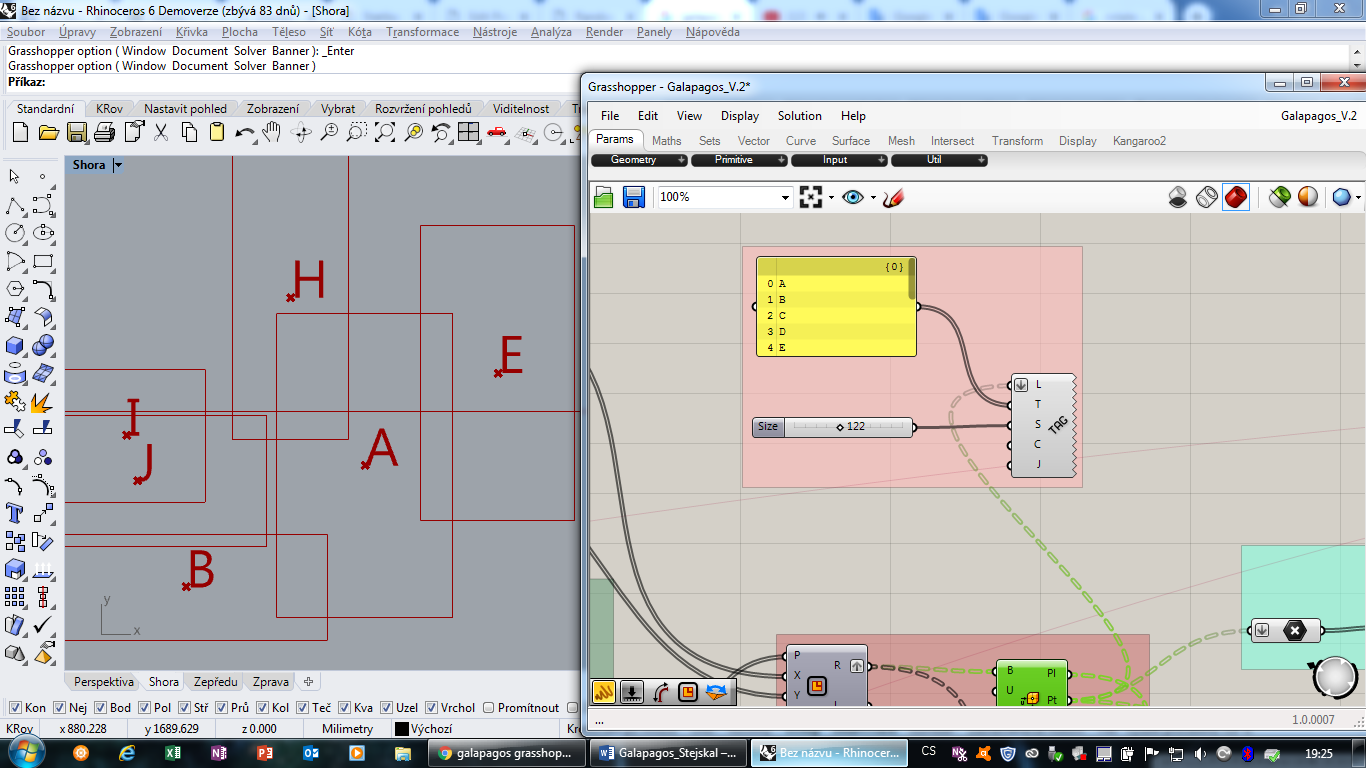
Second is creation of main rectangle. I used points to determinate vector and from one point and vectors I created rectangle.

Red part



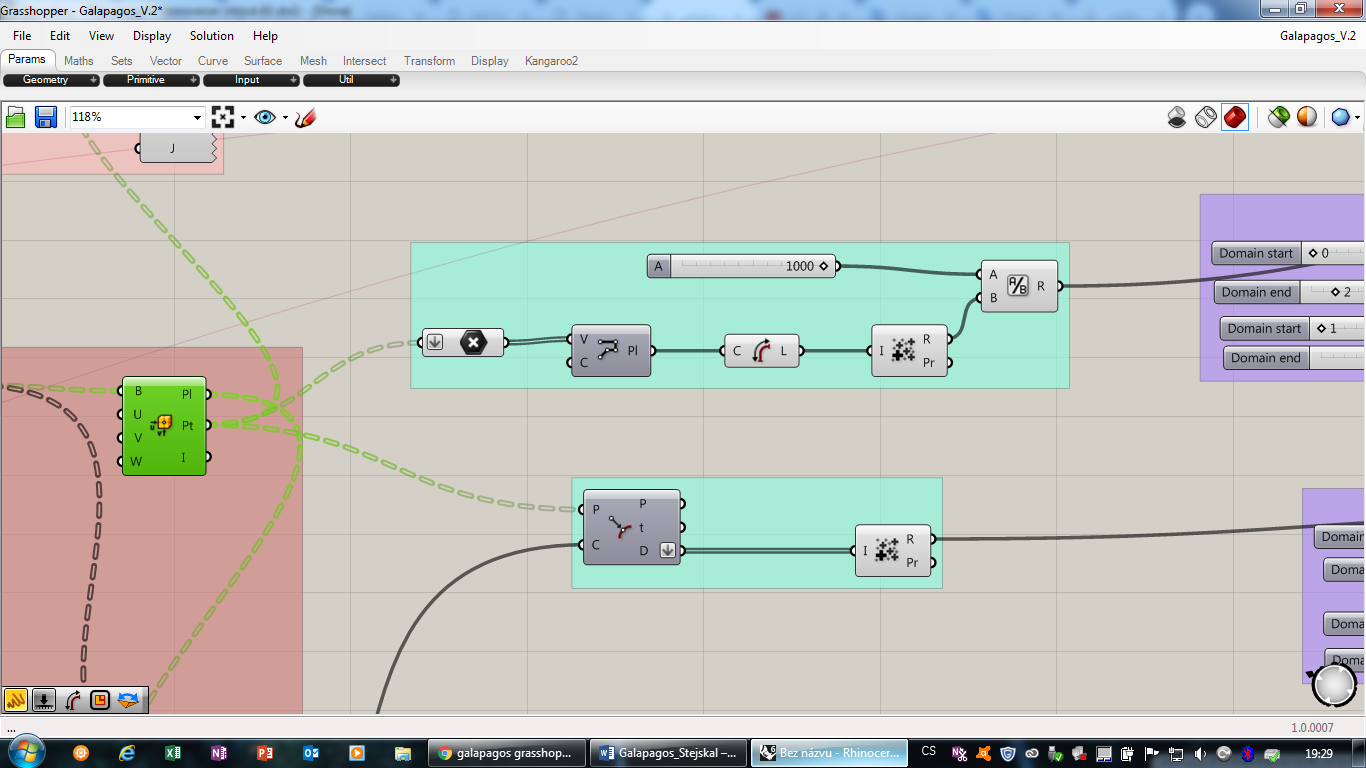
Here you can see creation of smaller rectangles from sizes determined by Gene pools. Then I created Evaluate box to get middle points of every small rectangle. After that I can rotate them. For this I used component Rotate which is determined by plan of rotation (plan from Evaluate box), Geometry (small rectangles) and Angle which mean how much it should rotate here it is half of pi.

Pink part



This part put names on the points, so I took middle points and set bigger size of text.

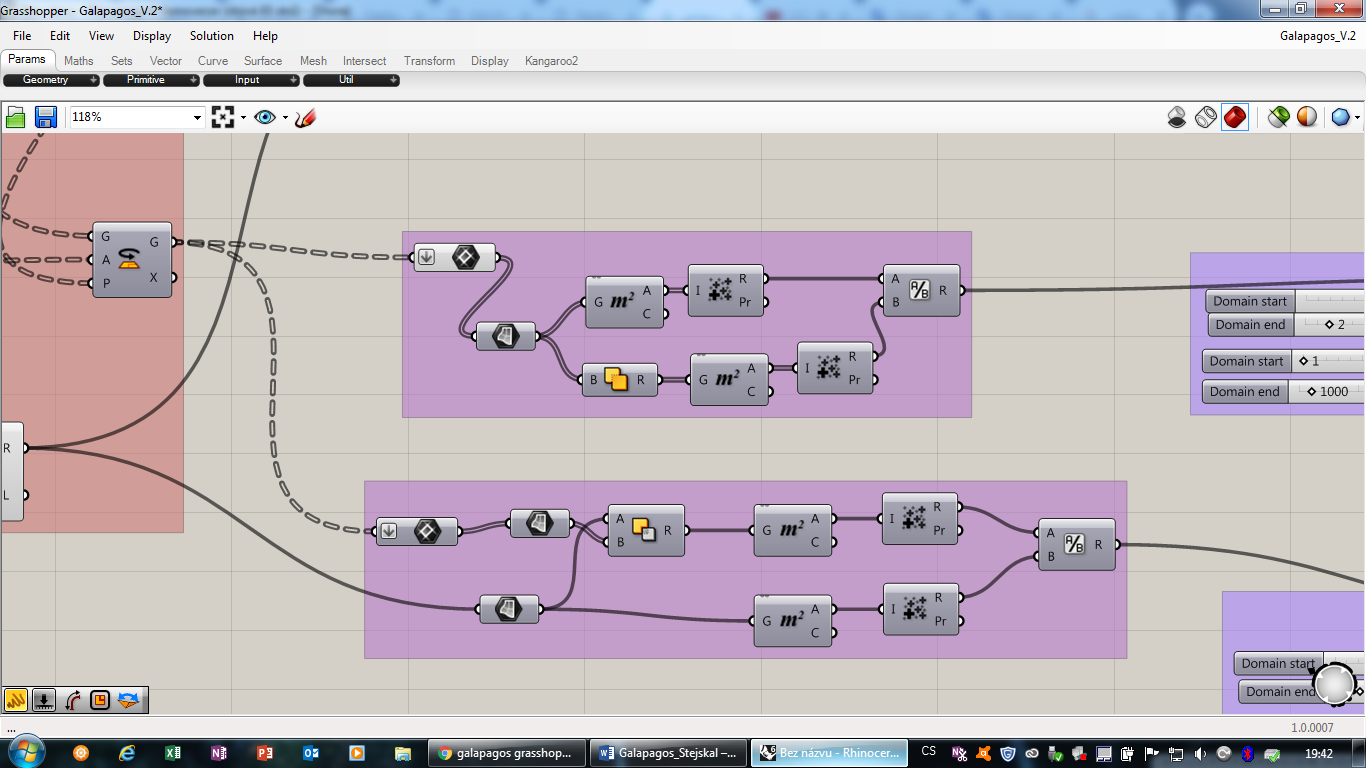
Turquoise part



Upper one is creation of polyline form middle points. Then I took its length. I had to used Mass addition to collect these data. Because I tried to find smallest number I used Division that means the higher number I will get from Mass addition the lowest number I will get from Division.

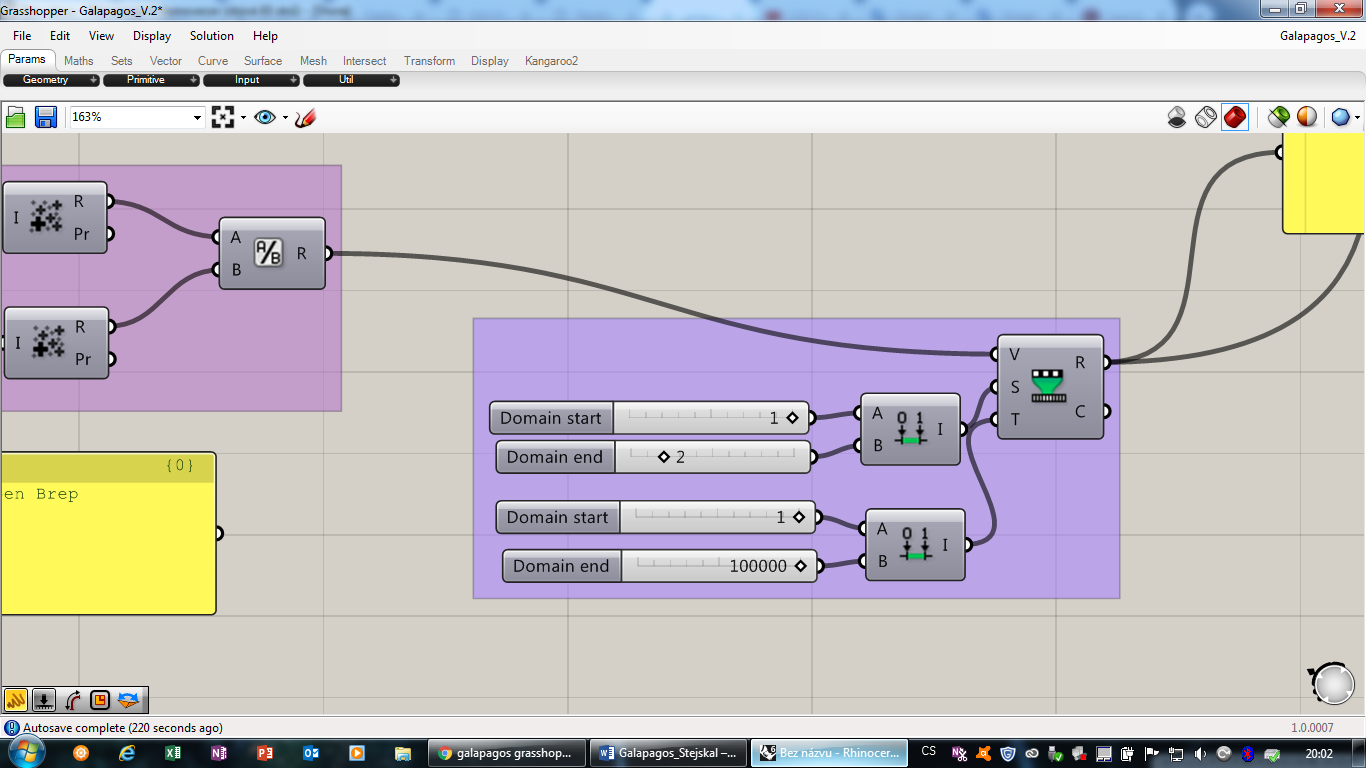
Second collect distances of middle points from main rectangle.

Purple part



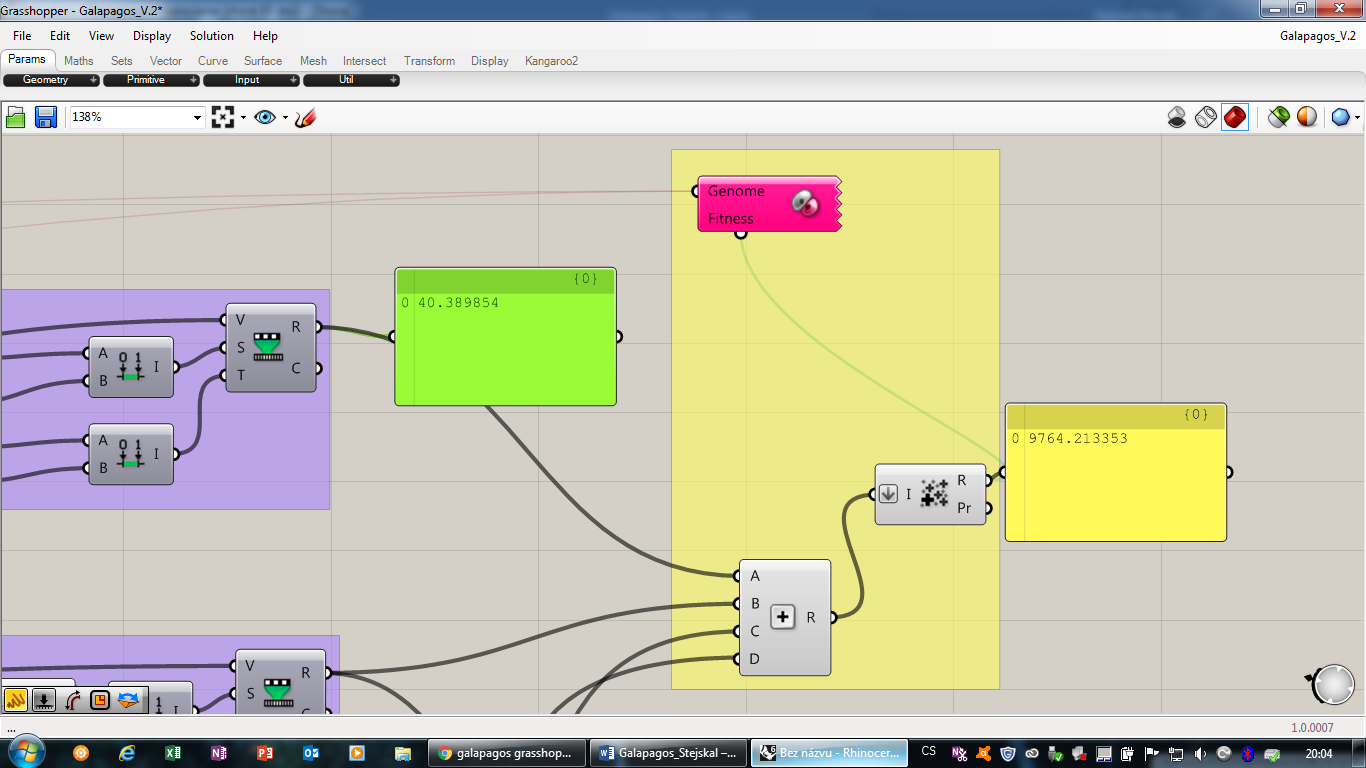
Here are two conditions. In both I took rectangles and created plane because it was easier for me to create conditions. First condition took area of all small rectangles when they are alone and area of Solid Union which is area filled by rectangles on surface. When they are same after division I get number 1 if not I get higher number which means some rectangles are overlap.

Second condition took area of main rectangle and Solid Difference of main rectangle and smaller rectangles when these two numbers are different it means part of some smaller rectangle is out of main rectangle. Division works same as in previous condition.

Blue pat

In this part you can see Remap number. Which I used for setting priority for each subject. This was created form two domain one for input data, which is range of data we get from our components and second domain which is range of numbers we want to get. In remaps for condition I set higher number of domain end, so I get higher numbers from this component which means Galapagos will prioritize them.

Yellow part

  
I used addition to sum all the important numbers then I used Mass addition to connect these data with Galapagos which is set to Minimize because I am looking for lowest number.

