

Лабораторная работа №6
Кобака Ф.А.
Вариант 2

Код:

```
% объявляю переменные которые представляют собой факторы производства
syms x1;
syms x2;

% формирую производственную функцию
disp('getted function');
f = (x1^(0.3))*(x2^(0.7))

disp('task 1++++++++');

% проверяю условие чтобы в нуле любого из ресурсов, производство равнялось нулю
if(subs(f , {'x1'} , 0) ~= 0)
    disp('its not production function');
    return;
end
% для второго фактора также
if(subs(f , {'x2'} , 0) ~= 0)
    disp('its not production function');
    return;
end

disp('first conditions completed');

% второе и третье условие я проверял графически, тут отрисовка графиков,
% первых частных производных (как по графику видно они положительны при
%положительных ресурсах)
dx1 = diff(f , 'x1');
dx2 = diff(f , 'x2');
subplot(2, 2, 1);
ezsurf(dx1, [0 3])
subplot(2, 2, 2);
ezsurf(dx2, [0 3])

% тут тоже самое для вторых, они же отрицательны при положительных ресурсах
d2x1 = diff(dx1, 'x1');
d2x2 = diff(dx2, 'x2');
subplot(2 , 2 , 3);
ezsurf(d2x1, [0 3]);
subplot(2, 2, 4);
ezsurf(d2x2, [0 3]);

% последнее условие не ясно как формализовать, но для этой функции оно точно
% выполняется, т.к. функция однородна
%  $(x1*k)^{0.3} * (x2*k)^{0.7} = k*(x1^{0.3} * x2^{0.7})$ ;

disp('task 1++++++++');

% далее рисование изоклин/изоквант
disp('task 2++++++++');
% массив на котором все будет рисоваться
tx1 = 0.02:0.001:3;
```

```

% массивы точек на изокостах, для различных уровней выпуска
izo1 = (2./(tx1.^0.3)).^(10/7);
izo2 = (3./(tx1.^0.3)).^(10/7);
izo3 = (5./(tx1.^0.3)).^(10/7);

% тут создаю главное окошко в которое буду рисовать в заданиях 2 , 3 , 6
mainfigure = figure

% в одно подокошко рисую самодельные изокванты
subplot(1 ,2 ,1);
plot(tx1 , izo1 , tx1 , izo2 , tx1 , izo3);
axis square;
title('my isoquants');

% тут формирую массивы значений для рисования изоквант как линий уровня
subplot(1 ,2 ,2);
% сетка на которой буду рисовать
[X1 , X2] = meshgrid(0:0.1:10);
% массив значений производственной функции на этой сетке
Z = subs(f , {x1 x2} , {X1 X2});

% по этой сетке рисуем линии уровня с заданными объемами выпуска
[C,H] = contour(X1 , X2 , Z , [2 3 5]);
axis square;
% подписываю линии уровня цифорками,
clabel(C,H);
title('isoquants by contour');

disp('task 2++++++++');

% пришло время изоклиналей
disp('taks 3++++++++');

% формирую массивы точек для изоклиналей с заданными нормами замещения
izoclinal1 = ((1-(0.3))/0.3)*tx1;
izoclinal2 = 2.*izoclinal1;
izoclinal3 = 3.*izoclinal1;

hold on;
% рисую изоклинали
plot(tx1 , izoclinal1 , tx1 , izoclinal2 , tx1 , izoclinal3);

disp('taks 3++++++++');

% задание 4 предполагалась с использованием графического редактора

disp('task 5++++++++');
% сгенеренный m-file и поправки в нем представлен ниже
disp('calling a m-file generated for plot builded in task 4, whith some changes from me');
% он называется createfigure и ждет что внего передадут данные для построения
% всего, что я рисовал выше, я просто по его коду до этого додумлся (типо никакой
% документации нет)
createfigure(tx1 , [izo1 ; izo2 ; izo3] , X1 , X2 , Z , [izoclinal1 ; izoclinal2 ; izoclinal3]);
disp('task 5++++++++');

disp('task6 ++++++');
% переключимся обратно на старое окошко

```

```

figure(mainfigure);

% сформируем массив symbolic выражений для каждой изокванты
izokvantArray = [sym('(2/(x1^0.3))^(10/7)'), sym('(3/(x1^0.3))^(10/7)'),
sym('(5/(x1^0.3))^(10/7)')];
% для каждой изокосты
izoklinArray = [sym('((1-(0.3))/0.3)*x1'), sym('((1-(0.3))/0.3)*x1^2'), sym('((1-
(0.3))/0.3)*x1^3')];

% открываем файл куда, будем писать все это
fileID = fopen('/home/dranik/KFA/programming/MatLab/CompModEconSystems/lab
6/pointsFile', 'w');
% делаем 'шапку'
fprintf(fileID, '%s \n', 'x    y')
% тут перебираю изокосты и изокванты и между каждой парой ищу координаты пересечения
for (izokv = izokvantArray)
    for (izokl = izoklinArray)
        % x пересечения
        pX = fzero(matlabFunction(izokv - izokl), 3);
        % y пересечения
        pY = subs(izokv, pX);
        % ставим там точку
        plot(pX, pY, '.r', 'MarkerSize', 20);
        % пишем в файл
        fprintf(fileID, '%s \n', [num2str(pX), ' ', num2str(pY)]);
    end
end
% закрываем файл
fclose(fileID);
disp('task6 ++++++')

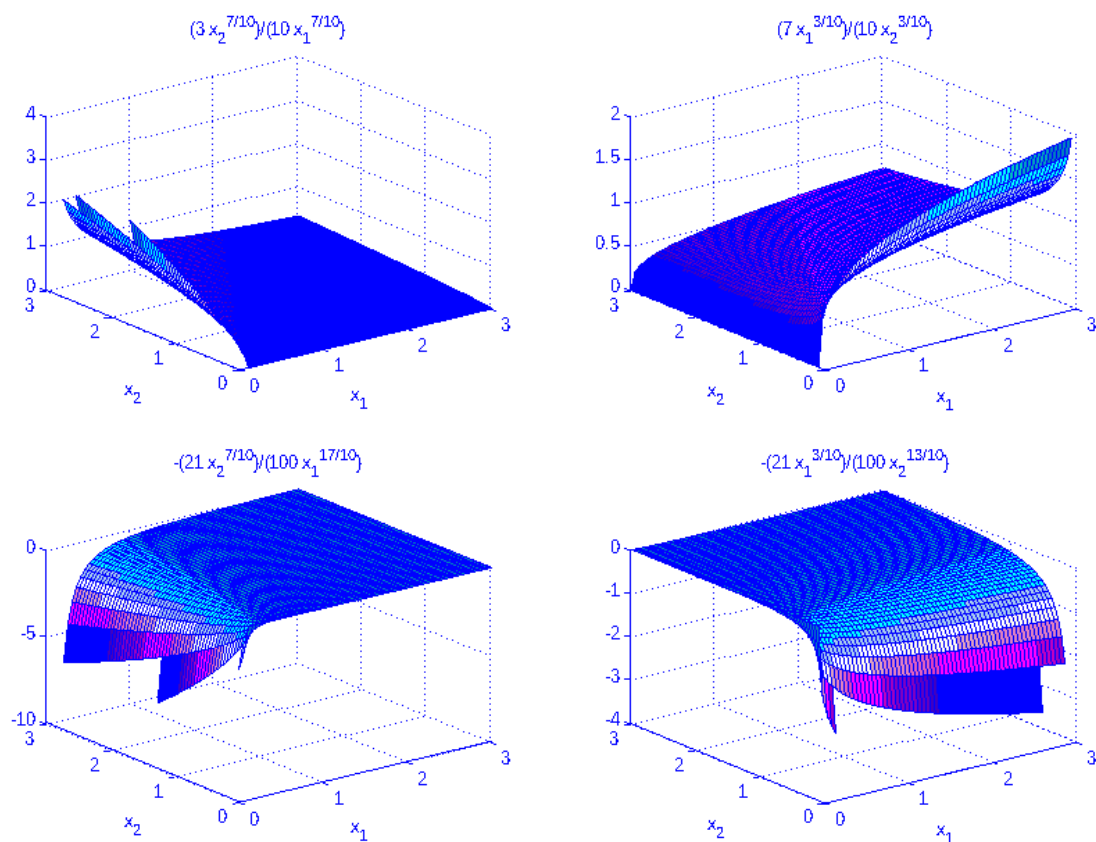
```

Результаты выполнения:

Задание 1:

по прошествии первого условия получил
first conditions completed

далее получил графики

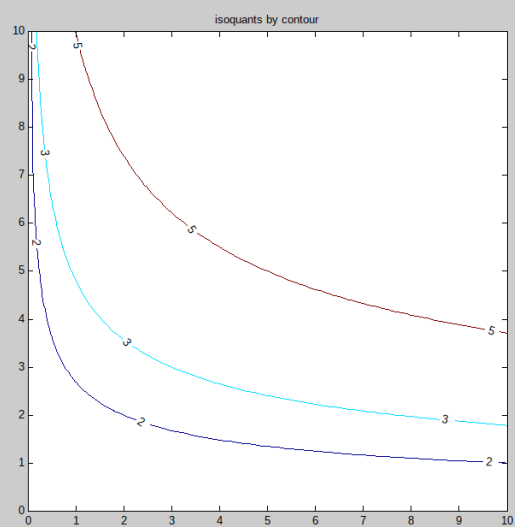
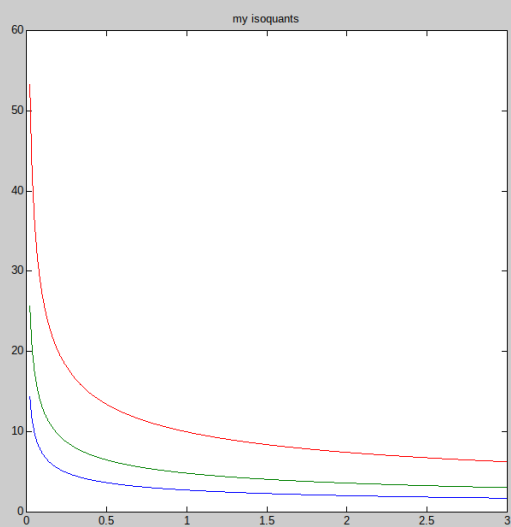


Даже не догадываюсь, что произошло с цветом картинки при сохранении, но даже на этой картинке, сверху первые производные, снизу вторые, слева по x_1 справа по x_2 , и первые производные выше нуля \rightarrow условие повышения продукта при повышении любого из факторов выполнено.

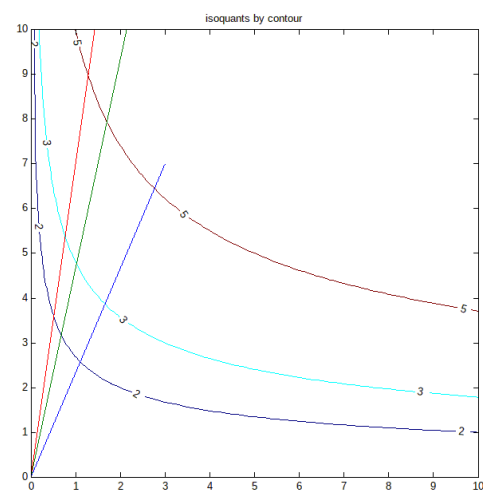
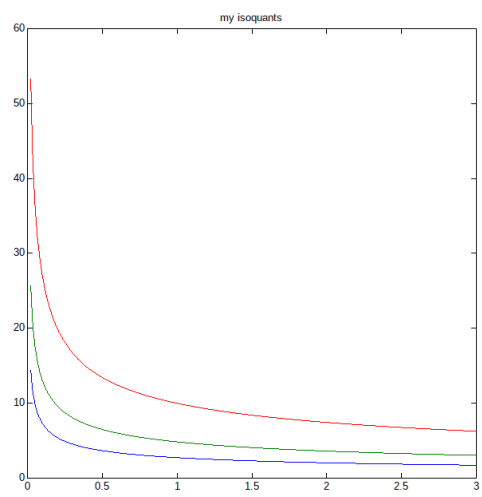
Даже по верхним картинкам уже видно, что второе условие так же выполнено, ведь при росте той переменной по которой дифференцировали, график „прижимается“ к нулю. Но на всякий случай я построил и вторые, где отчетливо видно, что они ниже 0 \rightarrow условие убывания полезности ресурса выполнено.

Четвертое условие не удалось формализовать, не смотря на то, что оно очевидно $(x_1 \cdot k)^{0.3} \cdot (x_2 \cdot k)^{0.7} = k \cdot (x_1^{0.3} \cdot x_2^{0.7}) \rightarrow$ функция однородна.

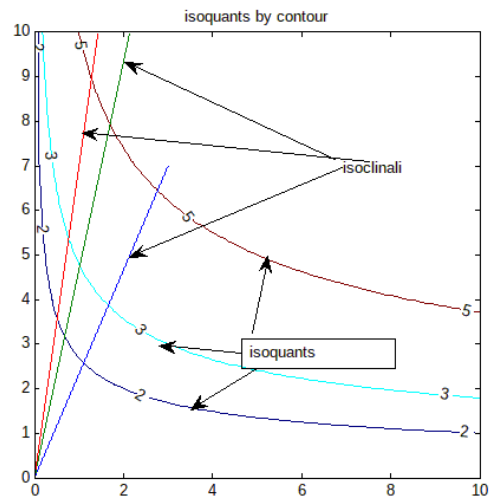
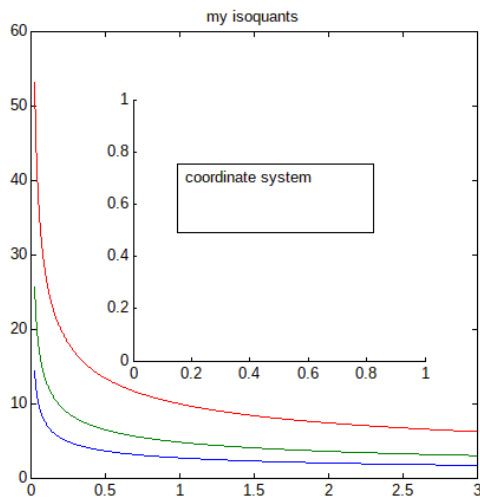
Задание 2:



Задание 3:



Задание 4 (оно я так понял не кодом делается):



Задание 5.

полученный m-file

```
function createfigure(X1, YMatrix1, xdata1, ydata1, zdata1, YMatrix2)
```

```
%CREATEFIGURE(X1,YMATRIX1,XDATA1,YDATA1,ZDATA1,YMATRIX2)
```

```
% X1: vector of x data
```

```
% YMATRIX1: matrix of y data
```

```
% XDATA1: contour x
```

```
% YDATA1: contour y
```

```
% ZDATA1: contour z
```

```
% YMATRIX2: matrix of y data
```

```
% Auto-generated by MATLAB on 14-Apr-2020 17:21:01
```

```
% Create figure
```

```
figure1 = figure;
```

```
% Create subplot
```

```
subplot1 = subplot(1,2,1,'Parent',figure1,'PlotBoxAspectRatio',[1 1 1]);
```

```
box(subplot1,'on');
```

```

hold(subplot1, 'all');

% Create multiple lines using matrix input to plot
plot(X1,YMatrix1, 'Parent', subplot1);

% Create title
title('my isoquants');

% Create subplot
subplot2 = subplot(1,2,2, 'Parent', figure1, 'PlotBoxAspectRatio', [1 1 1], ...
    'Layer', 'top');
% Uncomment the following line to preserve the X-limits of the axes
% xlim(subplot2, [0 10]);
% Uncomment the following line to preserve the Y-limits of the axes
% ylim(subplot2, [0 10]);
box(subplot2, 'on');
hold(subplot2, 'all');

% Create contour
contour(xdata1,ydata1,zdata1, 'ShowText', 'on', 'LevelList', [2 3 5], ...
    'Parent', subplot2);

% Create title
title('isoquants by contour');

% Create multiple lines using matrix input to plot
plot(X1,YMatrix2, 'Parent', subplot2);

% Create axes
axes('Parent', figure1, ...
    'Position', [0.206743566992014 0.420882669537137 0.219165927240461
0.23789020452099]);

% Create textarrow
annotation(figure1, 'textarrow', [0.802129547471162 0.642413487133984], ...
    [0.596416576964478 0.515608180839612], 'TextEdgeColor', 'none', ...
    'String', {'isoclinali'});

% Create arrow
annotation(figure1, 'arrow', [0.796805678793256 0.638864241348713], ...
    [0.603951560818084 0.693218514531755]);

% Create arrow
annotation(figure1, 'arrow', [0.822537710736469 0.60603371783496], ...
    [0.601798708288482 0.628632938643703]);

% Create textbox
annotation(figure1, 'textbox', ...
    [0.23968677905945 0.53821313240043 0.147181011535049 0.0624327233584501], ...
    'String', {'coordinate system'}, ...
    'FitBoxToText', 'off');

% Create textbox
annotation(figure1, 'textbox', ...
    [0.725933451641526 0.414424111948331 0.115237799467613 0.0269106566200216], ...
    'String', {'isoquants'}, ...
    'FitBoxToText', 'off');

% Create arrow

```



```

annotation(figure1,'arrow',[0.733806566104703 0.74622892635315],...
    [0.445716899892357 0.516684607104413]);

% Create arrow
annotation(figure1,'arrow',[0.725820763087844 0.664596273291925],...
    [0.427417653390743 0.434876210979548]);

% Create arrow
annotation(figure1,'arrow',[0.738243123336291 0.687666370896185],...
    [0.413424111948332 0.376749192680301]);

```

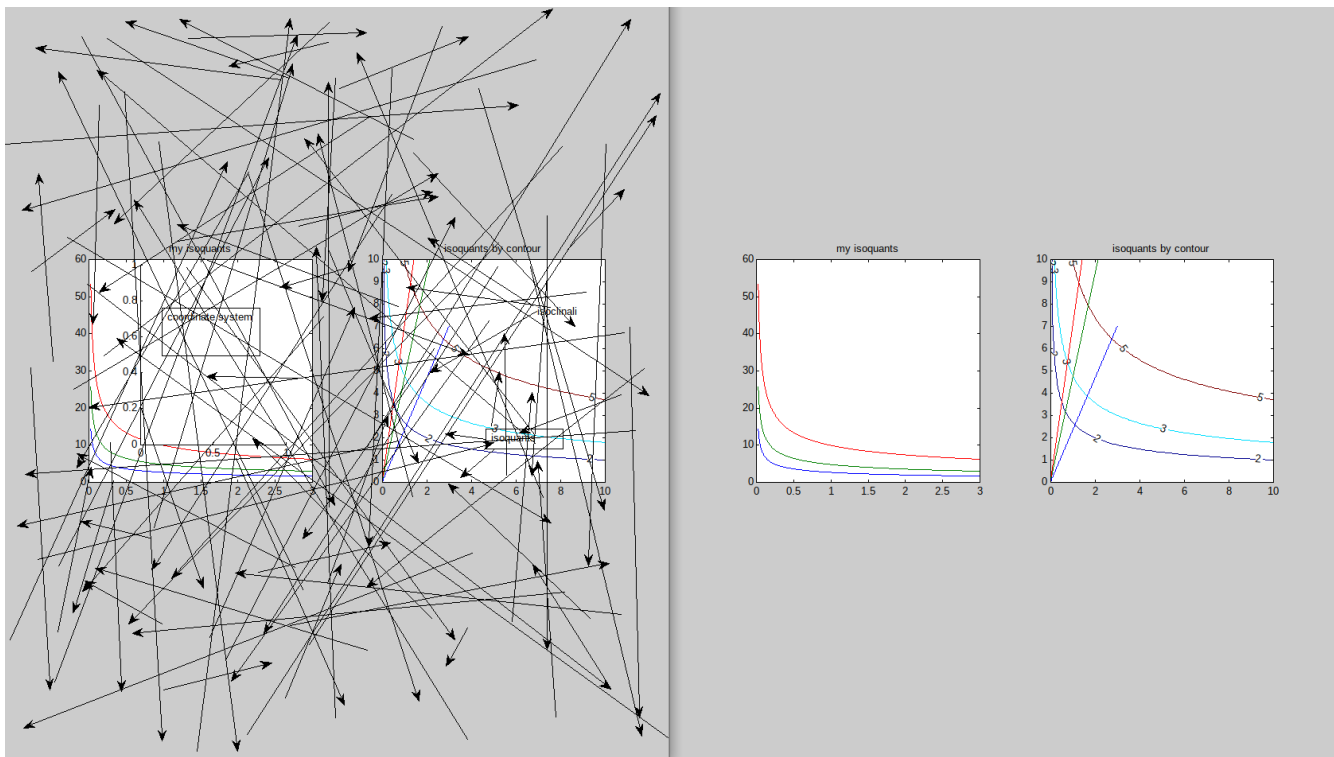
в конце я дописал свои изменения, нагенерировал 100 случайных стрелочек

```

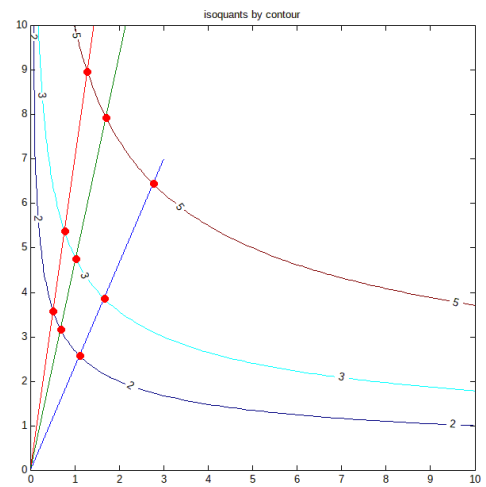
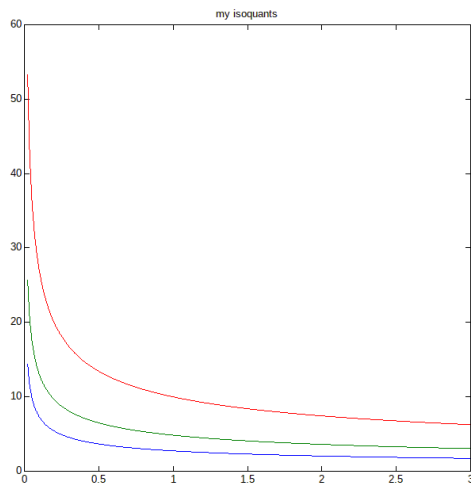
for(i = 1:100)
    annotation(figure1,'arrow',[rand() rand() ],...
        [rand() rand()]);
end

```

При вызове получил результат



Задание 6.



и в результате сохранения получены следующие точки

x y	
1.1052	2.5788
0.68034	3.1749
0.51223	3.5856
1.6578	3.8682
1.0205	4.7624
0.76834	5.3784
2.763	6.4471
1.7008	7.9373
1.2806	8.9639