

1.

$$\begin{aligned} & \min[30x + 20y + 50z] \\ & \begin{cases} 50x + 20y + 180z = 2000, \\ 6x + 4y + 3z = 120, \\ 2x + y + z = 40, \\ x, y, z \geq 0. \end{cases} \end{aligned}$$

MatLab Code:

```
clear all

clc

% Коэффициенты целевой функции
f = [30 20 50];

% Матрица ограничений равенств
Aeq = [50 20 180; 6 4 3; 2 1 1];
beq = [2000; 120; 40];

% Нижние границы для переменных (x, y, z >= 0)
lb = [0 0 0];

% Верхние границы отсутствуют
ub = [];

% Вызов функции linprog для решения задачи
[x, fval] = linprog(f, [], [], Aeq, beq, lb, ub);

% Вывод результата
x

fval
```

The screenshot shows the MATLAB Command Window and Workspace. The Command Window displays the output of the `linprog` function, indicating that an optimal solution was found. The optimal solution vector `x` is displayed as a column vector with values 16.7742, 0, and 6.4516. The optimal objective function value `fval` is 825.8065. The Workspace window shows the variables defined in the script: `Aeq` (a 3x3 matrix), `beq` (a 3x1 vector), `f` (a 1x3 vector), `fval` (a scalar), `lb` (a 3x1 vector), `ub` (an empty vector), and `x` (a 3x1 vector).

Name	Value
Aeq	[50,20,180;6,4,3;2,1,1]
beq	[2000;120;40]
f	[30,20,50]
fval	825.8065
lb	[0,0,0]
ub	[]
x	[16.7742;0;6.4516]

=30*C2+20*D2+50*E2							
	B	C	D	E	F	G	
	переменные	16,77419355	0	6,451612903	ЦФ		
		x	y	z	825,8064516	правые части	
	ограничение	50	20	180	2000	2000	
	ограничение	6	4	3	120	120	
	ограничение	2	1	1	40	40	

2.

$$(x_1 - 2)^2 + (x_2 - 1)^2 \Rightarrow \max$$

$$0 \leq x_1 \leq 5,$$

$$0 \leq x_2 \leq 3.$$

$$x_1^2 - 4x_1 + 4 + x_2^2 - 2x_2 + 1 = x_1^2 + x_2^2 - 4x_1 - 2x_2 + 5 \rightarrow \max$$

MatLab Code:

```
clear all
```

```
clc
```

```
% Матрица квадратичных коэффициентов
```

```
H = [2 0; 0 2];
```

```
% Вектор линейных коэффициентов
```

```
f = [-4; -2];
```

```
% Границы для переменных x и y
```

```
lb = [0; 0]; % Нижние границы
```

```
ub = [5; 3]; % Верхние границы
```

```
% Вызов функции quadprog для минимизации
```

```
[x, fval] = quadprog(H, f, [], [], [], [], lb, ub);
```

```
% Вывод результата
```

```
x
```

```
fval = (fval + 5); % Возвращаем значение
```

```
fval
```



3.

$$\begin{aligned}
 2x_1 + x_2 &\Rightarrow \max \\
 3x_1 + 2x_2 &\leq 14, \\
 4x_1 - 5x_2 &\leq 5, \\
 -7x_1 + 2x_2 &\leq 4, \\
 x_1, x_2 &\geq 0, \quad x_1, x_2 \in \mathbb{Z}.
 \end{aligned}$$

clear all

clc

% Коэффициенты целевой функции

f = [-2; -1]; % Для минимизации (-2x - y)

% Матрица линейных ограничений

A = [3 2; 4 -5; -7 2];

% Вектор правых частей ограничений

b = [14; 5; 4];

% Целочисленные переменные: 1 2 означает, что x и y целые

intcon = [1 2];

% Нижние границы для переменных x и y

lb = [0; 0];

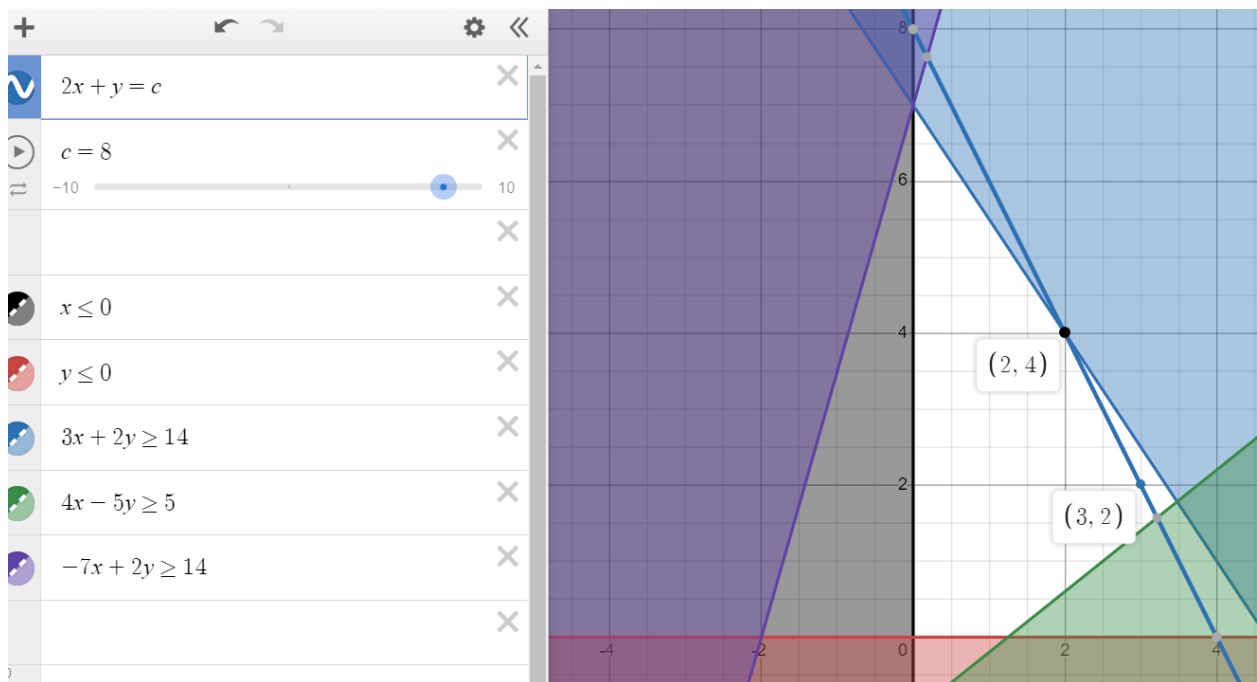
% Вызов функции intlinprog для решения задачи

[x, fval] = intlinprog(f, intcon, A, b, [], [], lb, []);

% Вывод результата

x

fval = -fval; % Так как мы минимизировали отрицательную функцию



4.

Рабочие	Виды работ			
	№1	№2	№3	№4
Иванов	8	4	6	5
Петров	6	5	8	5
Сидоров	8	2	4	7
Егоров	5	7	1	5

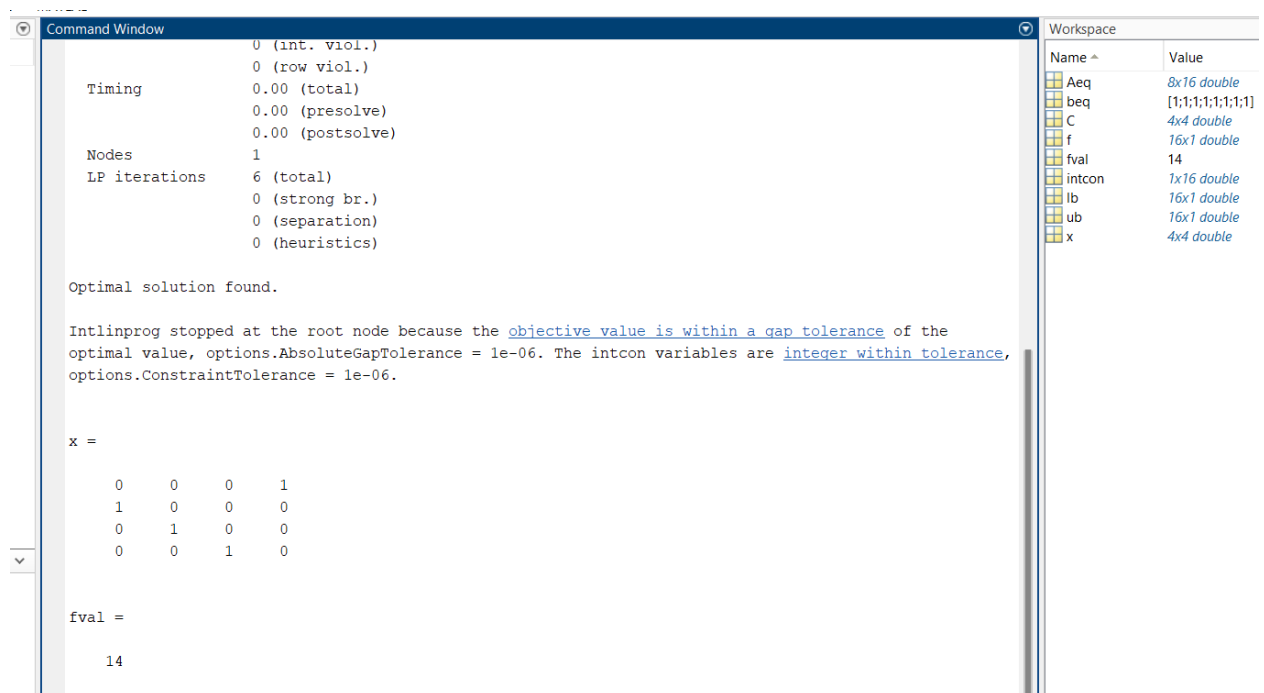
clear all

clc

```

% Матрица стоимости (эффективности)
C = [8 4 6 5; 6 5 8 5; 8 2 4 7; 5 7 1 5];
f = C(:); % Превращаем матрицу в вектор
% Ограничения: каждому рабочему назначается одна работа
Aeq = [kron(eye(4), ones(1,4)); kron(ones(1,4), eye(4))];
beq = ones(8,1);
% Указываем, что переменные целочисленные булевы (0 или 1)
intcon = 1:16;
% Нижние и верхние границы для переменных
lb = zeros(16,1); % 0
ub = ones(16,1); % 1
% Решение задачи с помощью функции intlinprog
[x, fval] = intlinprog(f, intcon, [], [], Aeq, beq, lb, ub);
% Переводим результат в матрицу 4x4
x = reshape(x, 4, 4)
% Оптимальное значение функции
fval

```



5.

<i>Пункты производства</i>	<i>Пункты распределения</i>				<i>Объемы производства</i>
	<i>№1</i>	<i>№2</i>	<i>№3</i>	<i>№4</i>	
<i>№1</i>	2	7	7	6	15
<i>№2</i>	1	1	1	2	17
<i>№3</i>	5	5	4	1	45
<i>№4</i>	2	8	3	4	20
<i>№5</i>	3	2	1	5	13
<i>Объемы потребления</i>	40	30	10	30	

```
clc
```

```
clear all
```

```
f = [ 2 7 7 6 1 1 1 2 5 5 4 1 2 8 3 4 3 2 1 5];
```

```
intcon = [1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20];
```

```
Aeq = [kron(eye(5), ones(1, 4)); kron(ones(1, 5), eye(4))];
```

```
beq = [15 17 45 20 13 40 30 10 30];
```

```
lb = zeros(length(f), 1);
```

```
[x, fval] = intlinprog(f, intcon, [], [], Aeq, beq, lb, []);
```

```
Xopt = reshape(x, 4, 5)';
```

```
Fopt = fval;
```

```
disp('result matrix:');
```

```
disp(Xopt);
```

```
disp('min');
```

```
disp(Fopt);
```

result matrix:

15	0	0	0
0	17	0	0
5	10	0	30
20	0	0	0
0	3	10	0

min

208

	40	30	10	30			
15	15	0	0	0			208
17	0	17	0	0			
45	5	10	0	30			
20	20	0	0	0			
13	0	3	10	0			