Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и кибербезопасности
Высшая школа компьютерных технологий и информационных систем

Отчёт расчетной работе № 1

Дисциплина: Системный анализ и принятие решений.

Выполнил студент гр. 5130901/10101 ________________ Д.Л. Симоновский
(подпись)

Руководитель ________________ А.Г. Сиднев
(подпись)

“20” февраля 2024 г.

Санкт-Петербург
2024

Оглавление

1. Условие:

1.1. Вариант:

Вариант №10.
Граф №19.

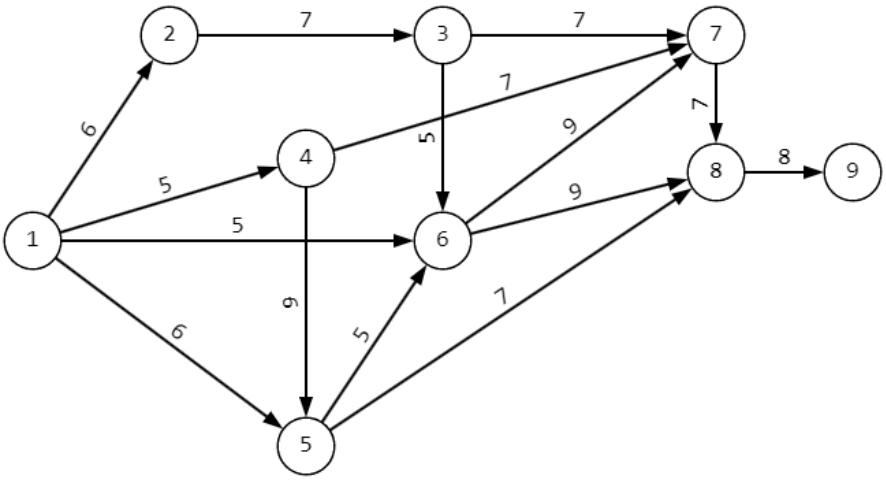

Рис. 1.1. Граф №19.

Число исполнителей 2.
Решающее правило: Короткие работы вперед.

1.2. Условие задания:

Выполнить следующие разделы:

1. Определить наиболее ранние моменты начала работ с использованием метода математического программирования.
2. Определить наиболее ранние моменты начала работ и их интенсивности, если длительность равна интенсивности выполнения работ, а суммарная интенсивность не превышает 75% от общего числа выполняемых работ.
3. Самостоятельно распределить работы между заданным числом исполнителей и сформулировать задачу математического программирования с бинарными индикаторными переменными. Определить число бинарных переменных и дополнительных ограничений в этой задаче и дать содержательную формулировку части ограничений с бинарными переменными.
   3.1. Изменить формулировку задачи так, чтобы число бинарных переменных не превышало 10. Решить полученную задачу с использованием команды **intlinprog**. Определить мощность множества бинарных переменных задачи и дать содержательную интерпретацию полученному решению.
4. Найти характеристики t_i^* , t_i^{**} и r_{ij} расписания выполнения комплекса работ с использованием метода динамического программирования. Привести соответствующие уравнения Беллмана. Определить критические пути на графе.
5. Найти те же характеристики t_i^* , t_i^{**} и r_{ij} расписания выполнения комплекса работ с использованием математического программирования.
6. Определить помимо полных резервов времени $F_n = r_{ij}$ работ ij резервы времени, относящиеся к событиям j сетевого графа, а именно $F_{\text{нз1}}, F_{\text{с}}, F_{\text{нз2}}$.
7. Рассмотреть вероятностную постановку задачи анализа расписания.
   Считать СКО времен выполнения работ равными 5% от их длительностей. Предполагая неизменным критический путь (оценить справедливость этого предположения) найти вероятность того, что время выполнения комплекса работ не превысит найденного для детерминированной задачи в п.1 на 10%.

8. Представить пошаговую процедуру имитационного моделирования расписания по схеме событий с учетом числа исполнителей и решающего правила ранжирования работ из числа возможных. По результатам моделирования построить диаграмму Ганта.

2. Ход решения

2.1. Определить наиболее ранние моменты начала работ с использованием метода математического программирования:

Для графа, представленного на Рис. 1.1, составим систему неравенств для последующего решения с помощью методов линейного программирования. Обозначим за t_{ij} наиболее ранний момент начала работы ij, а за t_{end} – наиболее ранний момент окончания всех работ.

$$t_{23} \geq t_{12} + 6 \qquad t_{58} \geq t_{15} + 6 \qquad t_{68} \geq t_{56} + 5$$

$$t_{36} \geq t_{23} + 7 \qquad t_{58} \geq t_{45} + 9 \qquad t_{78} \geq t_{37} + 7$$

$$t_{37} \geq t_{23} + 7 \qquad t_{67} \geq t_{16} + 5 \qquad t_{78} \geq t_{47} + 7$$

$$t_{45} \geq t_{14} + 5 \qquad t_{67} \geq t_{36} + 5 \qquad t_{78} \geq t_{67} + 9$$

$$t_{47} \geq t_{14} + 5 \qquad t_{67} \geq t_{56} + 5 \qquad t_{89} \geq t_{78} + 7$$

$$t_{56} \geq t_{15} + 6 \qquad t_{68} \geq t_{16} + 5 \qquad t_{89} \geq t_{68} + 9$$

$$t_{56} \geq t_{45} + 9 \qquad t_{68} \geq t_{36} + 5 \qquad t_{89} \geq t_{58} + 7$$

Задача оптимизации – минимизация следующей функции:

$$min(\sum_{i,j} t_{i,j} + t_{end})$$

Решим эту задачу с помощью функции Matlab linprog. Для этого преобразуем полученные ранее ограничения в матрицы A и b:

```
%   t12 t14 t15 t16 t23 t36 t37 t45 t47 t56 t58 t67 t68 t78 t89
A = [1   0   0   0  -1   0   0   0   0   0   0   0   0   0   0; % 1
     0   0   0   0   1  -1   0   0   0   0   0   0   0   0   0; % 2
     0   0   0   0   1   0  -1   0   0   0   0   0   0   0   0; % 3
     0   1   0   0   0   0   0  -1   0   0   0   0   0   0   0; % 4
     0   1   0   0   0   0   0   0  -1   0   0   0   0   0   0; % 5
     0   0   1   0   0   0   0   0   0  -1   0   0   0   0   0; % 6
     0   0   0   0   0   0   0   1   0  -1   0   0   0   0   0; % 7
     0   0   1   0   0   0   0   0   0   0  -1   0   0   0   0; % 8
     0   0   0   0   0   0   0   1   0   0  -1   0   0   0   0; % 9
     0   0   0   1   0   0   0   0   0   0   0  -1   0   0   0; % 10
     0   0   0   0   0   1   0   0   0   0   0  -1   0   0   0; % 11
     0   0   0   0   0   0   0   0   0   1   0  -1   0   0   0; % 12
     0   0   0   1   0   0   0   0   0   0   0   0  -1   0   0; % 13
     0   0   0   0   0   1   0   0   0   0   0   0  -1   0   0; % 14
     0   0   0   0   0   0   0   0   0   1   0   0  -1   0   0; % 15
     0   0   0   0   0   0   1   0   0   0   0   0   0  -1   0; % 16
     0   0   0   0   0   0   0   0   1   0   0   0   0  -1   0; % 17
     0   0   0   0   0   0   0   0   0   0   0   1   0  -1   0; % 18
     0   0   0   0   0   0   0   0   0   0   0   0   0   1  -1; % 19
     0   0   0   0   0   0   0   0   0   0   0   0   1   0  -1; % 20
     0   0   0   0   0   0   0   0   0   0   1   0   0   0  -1; % 21
     ];

b = -[6; 7; 7; 5; 5; 6; 9; 6; 9; 5; 5; 5; 5; 5; 5; 7; 7; 9; 7; 9; 7];
```

После чего вызовем linprog:

```
f = ones(15, 1);
lb = zeros(15, 1);

linprog(f, A, b, [], [], lb, []);
```

Полученный результат выглядит следующим образом:

t_{12}	t_{14}	t_{15}	t_{16}	t_{23}	t_{36}	t_{37}	t_{45}	t_{47}	t_{56}	t_{58}	t_{67}	t_{68}	t_{78}	t_{89}
0	0	0	0	6	13	13	5	5	14	14	19	19	28	35

Табл. 2.1. Время начала всех работ.

Теперь мы знаем минимальное время начала каждой работы. Для получения информации о времени выполнения всех работ необходимо к времени начала работы t_{89} прибавить время её выполнения т.е. 8.

Итого время выполнения всех работ равно 43.

2.2. Определить наиболее ранние моменты начала работ и их интенсивности, если длительность равна интенсивности выполнения работ, а суммарная интенсивность не превышает 75% от общего числа выполняемых работ.

Мы сможем увеличить время выполнения всех работ за счет добавления интенсивностей работ, отличных от 1 – некоторые работы ускорим (интенсивность > 1), а некоторые замедлим (интенсивность < 1), если это потребуется.

Изменим исходную систему неравенств согласно правилу:

$$\min\left\{\sum_{i,j} t_{ij}\right\}$$

$$\begin{cases} t_{ij} \geq \tau_{li} + \dfrac{Q_{li}}{m_{li}}, i = \overline{1, M-1}; l \in G^{-}(i) \\ \displaystyle\sum_{i,j} m_{ij} \leq 0{,}75 * 15, l \in G^{-}(M) \\ t_{ij} \geq 0 \end{cases}$$

где m_{ij} – интенсивность ij работы.

Такими условиями мы пытаемся минимизировать время начала всех работ, при интенсивности, не превышающей 75% от числа выполняемых работ т.е. 15.

Создадим набор всех «работ» т.е. ребер графа и массив троек, где закодируем систему неравенств, созданную ранее:

```
works = [12 14 15 16 23 36 37 45 47 56 58 67 68 78 89 99];
conds = [
    23 12 6
    36 23 7
    37 23 7
    45 14 5
    47 14 5
    56 15 6
    56 45 9
    58 15 6
    58 45 9
    67 16 5
    67 36 5
    67 56 5
    68 16 5
    68 36 5
    68 56 5
    78 37 7
    78 47 7
    78 67 9
    89 78 7
    89 68 9
    89 58 7
    99 89 8
    ];
```

Стоит заметить, что появилась работа-фальшивка. Это необходимо, чтоб MATLAB оптимизировал также и путь из 8 в 9 вершину и выводил нам результат этой оптимизации. Создадим необходимые параметры для fmincon, а также функцию, которая распарсит заданные нам тройки в требуемые для fmincon значения и выведем результат выполнения на экран:

```
x0 = ones(length(works) * 2 - 1, 1);
lb = zeros(length(works) * 2 - 1, 1);

fun = @(x) sum(x(1:length(works)));

res = fmincon(fun, x0, [], [], [], [], lb, [], @funs);

function [c, ceq] = funs(x)
    c = [];
    for i = 1:length(conds)
        t1 = find(works == conds(i, 1));
        t2 = find(works == conds(i, 2));
        q = conds(i, 3);
        m = length(works) + t2;
        c(end + 1) = -x(t1(1)) + x(t2(1)) + q / x(m);
    end
    c(end + 1) = sum(x(length(works) + 1:end)) - 0.75 * (length(works) - 1);
    ceq = 0;
end

t_res = res(1:length(works));
m_res = res(length(works) + 1:end);
sum_m_res = sum(m_res);

disp(t_res);
disp(m_res);
disp(sum_m_res);
```

В результате выполнения получим следующие значения:

Моменты начала работ	
t_{12}	0.00000
t_{14}	0.00000
t_{15}	0.00000
t_{16}	0.00000
t_{23}	4.56700
t_{36}	10.0636
t_{37}	10.0636
t_{45}	3.97630
t_{47}	3.97630
t_{56}	10.8390
t_{58}	10.8390
t_{67}	17.3038
t_{68}	17.3038
t_{78}	27.0136
t_{89}	36.5455
t_{end}	48.5008

Интенсивности	
m_{12}	1.3138
m_{14}	1.2574
m_{15}	0.5536
m_{16}	0.2890
m_{23}	1.2735
m_{36}	0.6906
m_{37}	0.4130
m_{45}	1.3114
m_{47}	0.3039
m_{56}	0.7734
m_{58}	0.2723
m_{67}	0.9269
m_{68}	0.4677
m_{78}	0.7344
m_{89}	0.6692

Табл. 2.2. Результат выполнения программы.

Сумма интенсивностей равна 11.25, что составляет ровно 75% от числа исполняемых работ, как и требовалось в задании. Стоит отметить, что общее время работы возрастало до 48.5 с 43 т.е. на 5.5 секунды (или в 1,13 раза). Это связано с тем, что при уменьшении интенсивности, некоторые работы стали работать дольше.

2.3. Самостоятельно распределить работы между заданным числом исполнителей и сформулировать задачу математического программирования с бинарными индикаторными переменными. Определить число бинарных переменных и дополнительных ограничений в этой задаче и дать содержательную формулировку части ограничений с бинарными переменными.

Распределим 15 работ по двум исполнителям следующим образом:

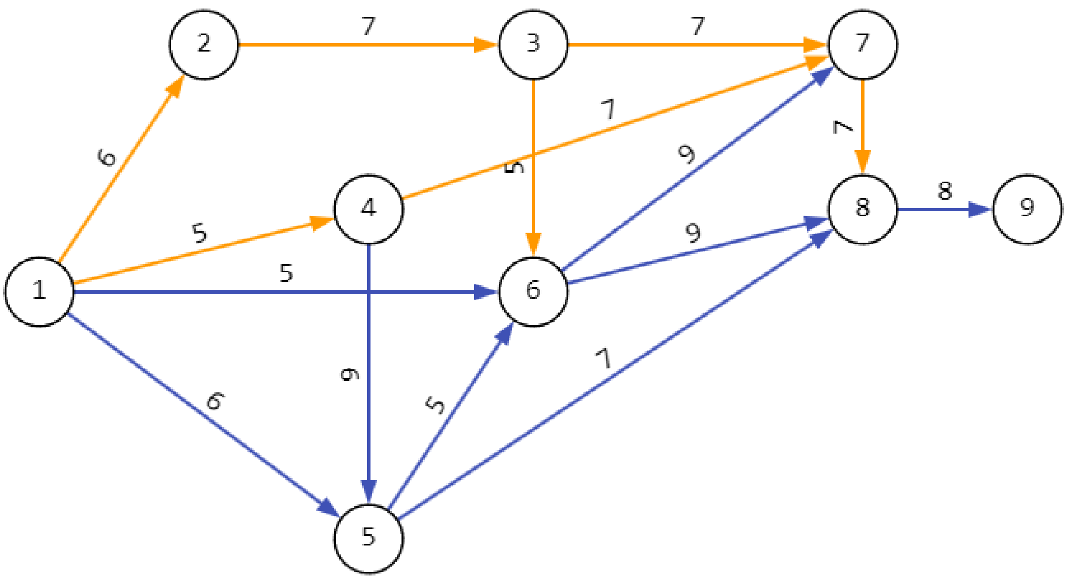

Рис. 2.1. Граф с задачами, распределенными по исполнителям.

Таким образом желтый цвет – первый исполнитель (7 задач), а синий – второй (8 задач).

Составим следующую систему для каждой пары работ $\{ij, lm\}$:

$$\begin{cases} (M+\tau_{lm})Y_{ij,lm,k} + (t_{ij} - t_{lm}) \geq \tau_{lm}, \\ (M+\tau_{ij})Y_{lm,ij,k} + (t_{lm} - t_{ij}) \geq \tau_{ij}, \\ Y_{ij,lm,k} + Y_{lm,ij,k} = 1 \end{cases}$$

где $|M| \gg \sum_{\{ij\}} \tau_{ij}$, тогда число дополнительных ограничений задачи с бинарными переменными будет равно $3(C_7^2 + C_8^2) = 3(21 + 28) = 147$, а число бинарных переменных $2(C_7^2 + C_8^2) = 2(21 + 28) = 98$.

2.3.1. Изменить формулировку задачи так, чтобы число бинарных переменных не превышало 10. Решить полученную задачу с использованием команды intlinprog. Определить мощность множества бинарных переменных задачи и дать содержательную интерпретацию полученному решению.

Упростим поставленную задачу, пусть только некоторые задачи выполняются определенным исполнителем, а над остальными может работать неограниченное число исполнителей:

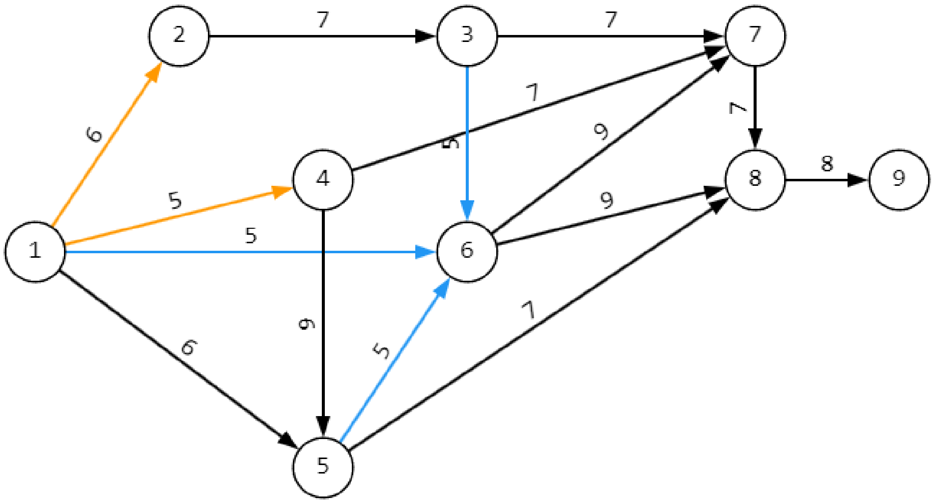

Рис. 2.2. Граф с задачами, часть которых распределена по исполнителям.

Таким образом желтый цвет – первый исполнитель (2 задачи), а синий – второй (3 задачи), что дает нам $2(C_2^2 + C_3^2) = 2(1 + 3) = 8$ дополнительных бинарных переменных и $3(C_2^2 + C_3^2) = 3(1 + 3) = 12$ дополнительных ограничений:

$$\begin{cases} (M+\tau_{12})Y_{14,12,1} + (t_{14} - t_{12}) \geq \tau_{12}, \\ (M+\tau_{14})Y_{12,14,1} + (t_{12} - t_{14}) \geq \tau_{14}, \\ Y_{14,12,1} + Y_{12,14,1} = 1 \end{cases}$$

$$\begin{cases} (M+\tau_{36})Y_{16,36,2} + (t_{16} - t_{36}) \geq \tau_{36}, \\ (M+\tau_{16})Y_{36,16,2} + (t_{36} - t_{16}) \geq \tau_{16}, \\ Y_{16,36,2} + Y_{36,16,2} = 1 \end{cases}$$

$$\begin{cases} (M+\tau_{36})Y_{56,36,3} + (t_{56} - t_{36}) \geq \tau_{36}, \\ (M+\tau_{56})Y_{36,56,3} + (t_{36} - t_{56}) \geq \tau_{56}, \\ Y_{56,36,3} + Y_{36,56,3} = 1 \end{cases}$$

$$\begin{cases} (M+\tau_{16})Y_{56,16,4} + (t_{56} - t_{16}) \geq \tau_{16}, \\ (M+\tau_{56})Y_{16,56,4} + (t_{16} - t_{56}) \geq \tau_{56}, \\ Y_{56,16,4} + Y_{16,56,4} = 1 \end{cases}$$

После чего решим задачу Matlab с использованием функции intlinprog. Для начала зададим наш граф, как делали это ранее, также добавим пары для новых ограничений и длины путей на каждой из задач:

```
works = [12 14 15 16 23 36 37 45 47 56 58 67 68 78 89 99];
conds = [
    23 12 6
    36 23 7
    37 23 7
    45 14 5
    47 14 5
    56 15 6
    56 45 9
    58 15 6
    58 45 9
    67 16 5
    67 36 5
    67 56 5
    68 16 5
    68 36 5
    68 56 5
    78 37 7
    78 47 7
    78 67 9
    89 78 7
    89 68 9
    89 58 7
    99 89 8
    ];

pairs = [
    12 14 6 5
    16 36 5 5
    56 36 5 5
    56 16 5 5
];
```

Далее создадим массив, как в пункте 2.1, но используя наши объявления:

```
cond_len = length(conds) + 2 * length(pairs);
x_len = length(works) + 2 * length(pairs);
f = ones(x_len, 1);
f(length(works) + 1:end) = 0;
A = zeros(cond_len, x_len);
b = zeros(1, cond_len);
for i = 1:length(conds)
    t1 = find(works == conds(i, 1));
    t2 = find(works == conds(i, 2));
    A(i, t1) = -1;
    A(i, t2) = 1;
    b(1, i) = -conds(i, 3);
end
```

Теперь необходимо создадим уравнения, которые добавились дополнительными ограничениями:

```matlab
Aeq = zeros(length(pairs), x_len);
beq = ones(1, length(pairs));

for i = 1:length(pairs)
    t1 = find(works == pairs(i, 1));
    t2 = find(works == pairs(i, 2));
    tau1 = pairs(i, 3);
    tau2 = pairs(i, 4);
    Y1 = length(works) + 2 * i - 1;
    Y2 = length(works) + 2 * i;
    M = 1000;

    idx = length(conds) + 2 * i - 1;
    A(idx, Y1) = -(M + tau2);
    A(idx, t1) = -1;
    A(idx, t2) = 1;
    b(1, idx) = -tau2;

    idx = idx + 1;
    A(idx, Y2) = -(M + tau1);
    A(idx, t1) = 1;
    A(idx, t2) = -1;
    b(1, idx) = -tau1;

    Aeq (i, Y1) = 1;
    Aeq (i, Y2) = 1;
end
```

Все необходимые переменные были созданы, теперь запустим intlinprog и посмотрим на результат:

```matlab
lb = zeros(x_len, 1);

x = intlinprog(f, (length(works) + 1):x_len, A, b, Aeq, beq, lb);

disp(x)
```

В результате запуска получим следующие значения:

Моменты начала работ	
t_{12}	5
t_{14}	0
t_{15}	0
t_{16}	0
t_{23}	11
t_{36}	19
t_{37}	18
t_{45}	5
t_{47}	5
t_{56}	14
t_{58}	14
t_{67}	24
t_{68}	24
t_{78}	33
t_{89}	40
t_{end}	48

Бинарные переменные	
$Y_{12,14,1}$	0
$Y_{14,12,1}$	1
$Y_{16,36,2}$	1
$Y_{36,16,2}$	0
$Y_{56,36,3}$	1
$Y_{36,56,3}$	0
$Y_{56,16,4}$	0
$Y_{16,56,4}$	1

Табл. 2.3. Результат работы программы.

Как мы видим по значениям бинарных переменных и времени начала работы наши условия выполняются.
Первый исполнитель выполняет сначала работу 14, а потом 12 (т. к. значение t_{12}=5, а t_{14}=0)
Второй исполнитель начинает с работы 16 (т. к. это единственная доступная со старта, что видно по нулевому значению t_{16}), далее решается работа 56 (т. к. до неё первой доходит очередь), после чего решается работа 36.
Остальные работы считаются параллельно, что видно, например по равным временам начала выполнения у работ 56 и 58 или 67 и 68.

2.4. Найти характеристики t_i^*, t_i^{**} и r_{ij} расписания выполнения комплекса работ с использованием метода динамического программирования. Привести соответствующие уравнения Беллмана. Определить критические пути на графе.

Каждому узлу на графе можно сопоставить два момента: минимальное время, когда событие будет осуществлено t_i^* и наиболее поздний момент t_i^{**}.
Воспользуемся методом динамического программирования и определим наиболее ранние моменты t_i^* для каждого узла графа:

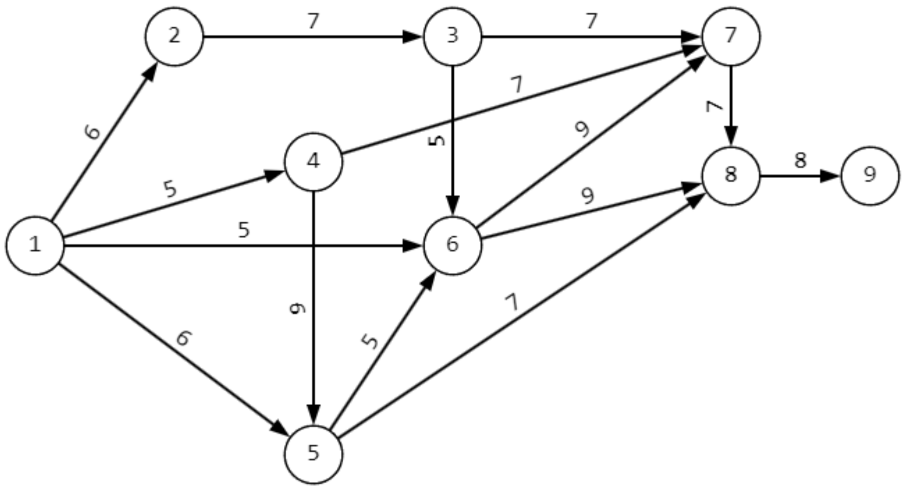

Рис. 2.3. Граф.

Самое ранее время начала выполнения работы – время, когда выполнятся все работы, предшествующие заданной:
$t_1^* = 0$
$t_2^* = t_1^* + \tau_{12} = 6$
$t_3^* = t_2^* + \tau_{23} = 13$
$t_4^* = t_1^* + \tau_{14} = 5$
$t_5^* = \max\{t_1^* + \tau_{15};\ t_4^* + \tau_{45}\} = \max\{6;\ 14\} = 14$
$t_6^* = \max\{t_5^* + \tau_{56};\ t_1^* + \tau_{16};\ t_3^* + \tau_{36}\} = \max\{19;\ 5; 18\} = 19$
$t_7^* = \max\{t_3^* + \tau_{37};\ t_4^* + \tau_{47};\ t_6^* + \tau_{67}\} = \max\{20; 12; 28\} = 28$
$t_8^* = \max\{t_7^* + \tau_{78};\ t_6^* + \tau_{68};\ t_5^* + \tau_{58}\} = \max\{34; 28; 24\} = 35$
$t_9^* = t_8^* + \tau_{89} = 43$
Полученные значение совпадают с полученными в предыдущих пунктах, что свидетельствует о корректности проделанной работы.
Теперь используя полученные значения определим наиболее поздние моменты времени t_i^{**}:
$t_9^{**} = 43$
$t_8^{**} = t_9^{**} - \tau_{89} = 43 - 8 = 35$
$t_7^{**} = t_8^{**} - \tau_{78} = 35 - 7 = 28$
$t_6^{**} = \min\{t_7^{**} - \tau_{67};\ t_8^{**} - \tau_{68}\} = \min\{19; 26\} = 19$
$t_5^{**} = \min\{t_6^{**} - \tau_{56};\ t_8^{**} - \tau_{58}\} = \min\{14; 28\} = 14$
$t_4^{**} = \min\{t_7^{**} - \tau_{47};\ t_5^{**} - \tau_{57}\} = \min\{21; 5\} = 5$
$t_3^{**} = \min\{t_7^{**} - \tau_{37};\ t_6^{**} - \tau_{36}\} = \min\{21; 14\} = 14$

$t_2^{**} = t_3^{**} - \tau_{23} = 14 - 7 = 7$
$t_1^{**} = \min\{t_2^{**} - \tau_{12};\ t_4^{**} - \tau_{14};\ t_6^{**} - \tau_{16};\ t_5^{**} - \tau_{15}\} = \min\{1; 0; 14; 8\} = 0$
По формуле $r_{ij} = t_j^{**} - (t_i^{*} + \tau_{ij})$ определим резервы времени выполнения всех работ:
$r_{12} = t_2^{**} - (t_1^{*} + \tau_{12}) = 7 - 6 = 1$
$\boldsymbol{r_{14} = t_4^{**} - (t_1^{*} + \tau_{14}) = 5 - 5 = 0}$
$r_{15} = t_5^{**} - (t_1^{*} + \tau_{15}) = 14 - 6 = 8$
$r_{16} = t_6^{**} - (t_1^{*} + \tau_{16}) = 19 - 5 = 14$
$r_{23} = t_3^{**} - (t_2^{*} + \tau_{23}) = 14 - 13 = 1$
$r_{36} = t_6^{**} - (t_3^{*} + \tau_{36}) = 19 - 18 = 1$
$r_{37} = t_7^{**} - (t_3^{*} + \tau_{37}) = 28 - 20 = 8$
$r_{47} = t_7^{**} - (t_4^{*} + \tau_{47}) = 28 - 12 = 16$
$\boldsymbol{r_{45} = t_5^{**} - (t_4^{*} + \tau_{45}) = 14 - 14 = 0}$
$\boldsymbol{r_{56} = t_6^{**} - (t_5^{*} + \tau_{56}) = 19 - 19 = 0}$
$r_{58} = t_8^{**} - (t_5^{*} + \tau_{58}) = 35 - 21 = 14$
$\boldsymbol{r_{67} = t_7^{**} - (t_6^{*} + \tau_{67}) = 28 - 28 = 0}$
$r_{68} = t_8^{**} - (t_6^{*} + \tau_{68}) = 35 - 28 = 7$
$\boldsymbol{r_{78} = t_8^{**} - (t_7^{*} + \tau_{78}) = 35 - 35 = 0}$
$\boldsymbol{r_{89} = t_9^{**} - (t_8^{*} + \tau_{89}) = 43 - 43 = 0}$
Работы, у которых резерв равен 0 – критический путь. Их длительность напрямую влияет на продолжительность выполнения всех работ.

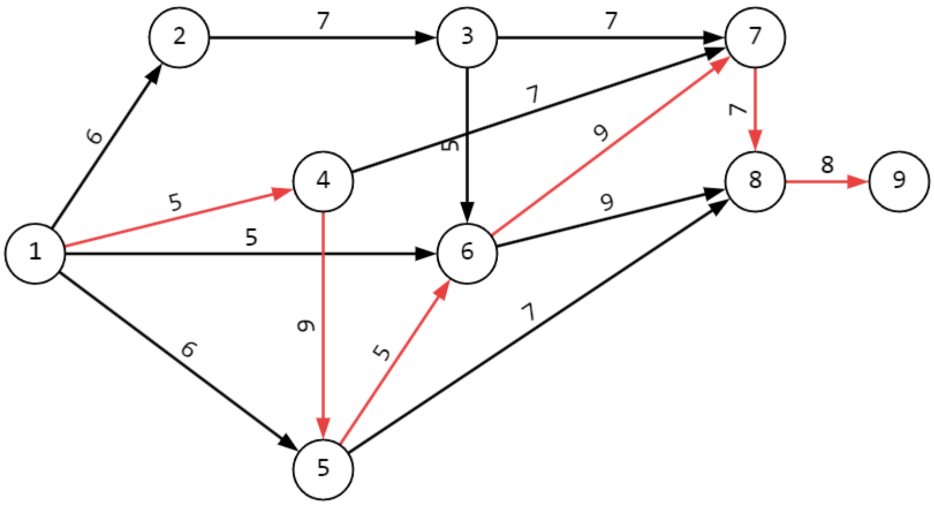

Рис. 2.4. Критический путь в графе.

2.5. Найти те же характеристики t_i^{*}, t_i^{**} и r_{ij} расписания выполнения комплекса работ с использованием математического программирования.

Оптимизационная задача для поиска наиболее ранних моментов t_i^{*} может быть сформулирована следующим образом:

$$\min\left\{\sum_{j=1} t_j^{*}\right\}$$

$$t_i^{*} - t_j^{*} \le -\tau_{ij}, j = \overline{1, n}$$

Для исходного графа получим следующую оптимизационную задачу:

$$\min\left\{\sum_{j=1} t_j^{*}\right\}$$

$t_1^{*} - t_2^{*} \le -6$	$t_3^{*} - t_6^{*} \le -5$	$t_5^{*} - t_8^{*} \le -7$
$t_1^{*} - t_4^{*} \le -5$	$t_3^{*} - t_7^{*} \le -7$	$t_6^{*} - t_7^{*} \le -9$

$$t_1^* - t_6^* \le -5$$
$$t_1^* - t_5^* \le -6$$
$$t_2^* - t_3^* \le -7$$
$$t_4^* - t_5^* \le -9$$
$$t_4^* - t_7^* \le -7$$
$$t_5^* - t_6^* \le -5$$
$$t_6^* - t_8^* \le -9$$
$$t_7^* - t_8^* \le -7$$
$$t_8^* - t_9^* \le -8$$
$$t_j \ge 0, j = \overline{1{,}9}$$

Запишем эти выражения в массивы:

```
times = [1 2 3 4 5 6 7 8 9];
conds = [
    1 2 -6
    1 4 -5
    1 6 -5
    1 5 -6
    2 3 -7
    3 6 -5
    3 7 -7
    4 5 -9
    4 7 -7
    5 6 -5
    5 8 -7
    6 7 -9
    6 8 -9
    7 8 -7
    8 9 -8
];
```

Запишем эти выражения в виде, подходящем для linprog и вычислим:

```
f = ones(length(times), 1);
A = zeros(length(conds), length(times));
b = zeros(1, length(conds));
for i = 1:length(conds)
    t1 = find(times == conds(i, 1));
    t2 = find(times == conds(i, 2));
    A(i, t1) = 1;
    A(i, t2) = -1;
    b(1, i) = conds(i, 3);
end

lb = zeros(length(times), 1);

t = linprog(f, A, b, [], [], lb);
disp(t)
```

Полученный результат выглядит следующим образом:

t_1^*	t_2^*	t_3^*	t_4^*	t_5^*	t_6^*	t_7^*	t_8^*	t_9^*
0	6	13	5	14	19	28	35	43

Табл. 2.4. Результат решения программой.

Как можно заметить, они идентичны полученным ранее другим способом.

Оптимизационная задача для поиска наиболее поздних моментов t_i^{**} может быть сформулирована следующим образом:

$$max\left\{\sum_{j=1} t_j^{**}\right\}$$

$$t_i^{**} - t_j^{**} \le -\tau_{ij}, j = \overline{1,n}$$

$$-t_1^{**} + t_9^{**} = 43$$

$$t_1^{**} = 0$$

Для решения поставленной задачи необходимо чуть-чуть изменить вызов функции, объявления условий останется аналогичным:

```
f = ones(length(times), 1);
A = zeros(length(conds), length(times));
b = zeros(1, length(conds));
for i = 1:length(conds)
    t1 = find(times == conds(i, 1));
    t2 = find(times == conds(i, 2));
    A(i, t1) = 1;
    A(i, t2) = -1;
    b(1, i) = conds(i, 3);
end

lb = zeros(length(times), 1);
Aeq = zeros(2, length(times));
Aeq(1, 1) = 1;
Aeq(2, 1) = -1;
Aeq(2, length(times)) = 1;
beq = [0; 43];

t = linprog(-f, A, b, Aeq, beq, lb);
disp(t)
```

Результатом выполнения будет следующим:

t_1^{**}	t_2^{**}	t_3^{**}	t_4^{**}	t_5^{**}	t_6^{**}	t_7^{**}	t_8^{**}	t_9^{**}
0	7	14	5	14	19	28	35	43

Табл. 2.5. Результат выполнения программы.

Как можно заметить, эти значения идентичны посчитанным ранее.
Очевидно, что значение r_{ij} будет аналогично равно посчитанному ранее.

2.6. Определить помимо полных резервов времени $F_n = r_{ij}$ работ ij резервы времени, относящиеся к событиям j сетевого графа, а именно $F_{\text{нз1}}, F_{\text{с}}, F_{\text{нз2}}$.

По формуле $F_{j\text{нз1}} = t_j^{**} - t_j^{*}$ определим независимые резервы 1-го порядка

j	1	2	3	4	5	6	7	8	9
t_j^{*}	0	6	13	5	14	19	28	35	43
t_j^{**}	0	7	14	5	14	19	28	35	43
$F_{j\text{нз1}}$	0	1	1	0	0	0	0	0	0

Табл. 2.6. Независимый резерв 1-го порядка.

По формуле $F_{ijC} = t_j^{*} - (t_i^{*} + \tau_{ij})$ определим свободные резервы времени:

$F_{12C} = t_2^{*} - (t_1^{*} + \tau_{12}) = 6 - 6 = 0$
$F_{14C} = t_4^{*} - (t_1^{*} + \tau_{14}) = 5 - 5 = 0$
$F_{15C} = t_5^{*} - (t_1^{*} + \tau_{15}) = 14 - 6 = 8$
$F_{16C} = t_6^{*} - (t_1^{*} + \tau_{16}) = 19 - 5 = 14$

$F_{23C} = t_3^* - (t_2^* + \tau_{23}) = 13 - 13 = 0$
$F_{36C} = t_6^* - (t_3^* + \tau_{36}) = 19 - 18 = 1$
$F_{37C} = t_7^* - (t_3^* + \tau_{37}) = 28 - 20 = 8$
$F_{47C} = t_7^* - (t_4^* + \tau_{47}) = 28 - 12 = 16$
$F_{45C} = t_5^* - (t_4^* + \tau_{45}) = 14 - 14 = 0$
$F_{56C} = t_6^* - (t_5^* + \tau_{56}) = 19 - 19 = 0$
$F_{58C} = t_8^* - (t_5^* + \tau_{58}) = 35 - 21 = 14$
$F_{67C} = t_7^* - (t_6^* + \tau_{67}) = 28 - 28 = 0$
$F_{68C} = t_8^* - (t_6^* + \tau_{68}) = 35 - 28 = 7$
$F_{78C} = t_8^* - (t_7^* + \tau_{78}) = 35 - 35 = 0$
$F_{89C} = t_9^* - (t_8^* + \tau_{89}) = 43 - 43 = 0$
По формуле $F_{ij\text{Н32}} = t_j^* - (t_i^{**} + \tau_{ij})$ определим независимые резервы 2-го порядка:
$F_{12\text{Н32}} = t_2^* - (t_1^{**} + \tau_{12}) = 6 - 6 = 0$
$F_{14\text{Н32}} = t_4^* - (t_1^{**} + \tau_{14}) = 5 - 5 = 0$
$F_{15\text{Н32}} = t_5^* - (t_1^{**} + \tau_{15}) = 14 - 6 = 8$
$F_{16\text{Н32}} = t_6^* - (t_1^{**} + \tau_{16}) = 19 - 5 = 14$
$F_{23\text{Н32}} = t_3^* - (t_2^{**} + \tau_{23}) = 13 - 14 = -1$
$F_{36\text{Н32}} = t_6^* - (t_3^{**} + \tau_{36}) = 19 - 19 = 0$
$F_{37\text{Н32}} = t_7^* - (t_3^{**} + \tau_{37}) = 28 - 21 = 7$
$F_{47\text{Н32}} = t_7^* - (t_4^{**} + \tau_{47}) = 28 - 12 = 16$
$F_{45\text{Н32}} = t_5^* - (t_4^{**} + \tau_{45}) = 14 - 14 = 0$
$F_{56\text{Н32}} = t_6^* - (t_5^{**} + \tau_{56}) = 19 - 19 = 0$
$F_{58\text{Н32}} = t_8^* - (t_5^{**} + \tau_{58}) = 35 - 21 = 14$
$F_{67\text{Н32}} = t_7^* - (t_6^{**} + \tau_{67}) = 28 - 28 = 0$
$F_{68\text{Н32}} = t_8^* - (t_6^{**} + \tau_{68}) = 35 - 28 = 7$
$F_{78\text{Н32}} = t_8^* - (t_7^{**} + \tau_{78}) = 35 - 35 = 0$
$F_{89\text{Н32}} = t_9^* - (t_8^{**} + \tau_{89}) = 43 - 43 = 0$

2.7. Рассмотреть вероятностную постановку задачи анализа расписания. Считать СКО времен выполнения работ равными 5% от их длительностей. Предполагая неизменным критический путь (оценить справедливость этого предположения) найти вероятность того, что время выполнения комплекса работ не превысит найденного для детерминированной задачи в п.1 на 10%.

Оценим справедливость неизменности критического пути. Среднее значение длительности работ в графе равно 6.4762 временных единиц. По условию СКО равно 5%, то есть $6.4762 * 0.05 = 0.32381$. Следовательно, значение длительности работы может отклониться более чем на 1 с очень маленькой вероятностью (по правилу трех сигм). Так как минимальные временной резерв у работы, не лежащей на критическом пути равен 1, то вероятность изменения критического пути очень мала.

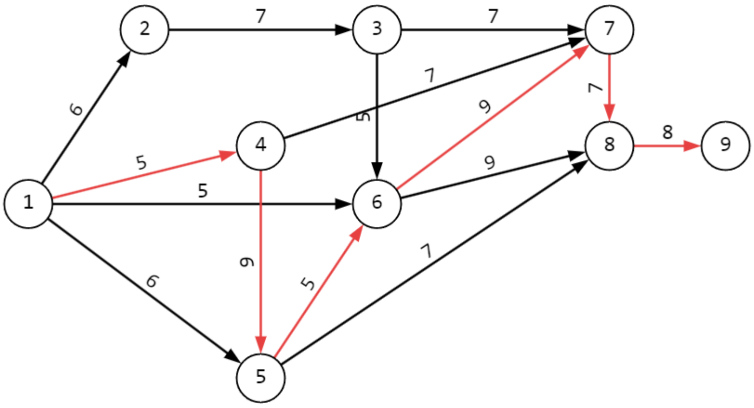

Рис. 2.5. Критический путь в графе.

Математическое ожидание суммы случайных величин равно сумме математических ожиданий:

$$\sum M_{ij} = \sum t_{ij} = 5 + 9 + 5 + 9 + 7 + 8 = 43$$

Дисперсия суммы равна сумме дисперсий:

$$\sum D_{ij} = 0.05^2 * (5^2 + 9^2 + 5^2 + 9^2 + 7^2 + 8^2) = 0.8125$$

Для суммы случайных величин длительностей работ имеем:

$P\big(T \geq 1.1M(T)\big) = 0.5 - \Phi\left(\frac{e}{\vartheta T}\right)$, где Φ – это функция Лапласа (табулированный интеграл вероятности):

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_0^x e^{-\frac{t^2}{2}} dt$$

По условию время выполнения комплекса работ не должно превышать детерминированное значение на 10%, то есть на $e = 43 * 0.1 = 4.3$.

$$P\big(T \geq 1.1M(T)\big) = 0.5 - \Phi\left(\frac{e}{\vartheta T}\right) = 0.5 - \Phi\left(\frac{4.3}{\sqrt{0.8125}}\right) = 0.5 - \Phi(4.7) = 0.5 - 0.499999 \approx 0$$

Результат показывает, что шанс отклониться от математического ожидания времени выполнения более чем на 10% крайне мал.

2.8. Представить пошаговую процедуру имитационного моделирования расписания по схеме событий с учетом числа исполнителей и решающего правила ранжирования работ из числа возможных. По результатам моделирования построить диаграмму Ганта.

Правило выбора работ:

- Решающее правило: Короткие работы — вперед
- Число исполнителей: 2

Параметры:

- T – системное время.
- $\Omega_{\text{Р}}(T)$ – ранжированный список возможных работ.
- $N(T)$ – список выполняемых на момент времени T работ: начатых, но не завершенных к этому моменту.
- $Z(T)$ – список времен освобождения ресурсов на момент времени T.
- $B(T)$ – список выполненных на момент времени T работ.
- $I(T)$ – список осуществленных событий.
- IJ – множество дуг-работ, исходящих из осуществленных событий.
- r_{sJOB} – список работ, выполняемых ресурсом s.
- r_{sSTART} – список моментов начала работ, выполняемых ресурсом s.
- $r_{sFINISH}$ – список моментов окончания работ, выполняемых ресурсом s.

T	$\Omega_{\text{Р}}(T)$	$N(T)$	$Z(T)$	$B(T)$	$I(T)$	IJ	r_{sJOB}	r_{sSTART}	$r_{sFINISH}$
	Что доступно	Выполняется сейчас	Время на выполнение	Список выполненных	Какие узлы закрыли	Все осущ работы+ доступ	Кто и что делает		
0	12, 14, 15, 16	∅	∅	∅	1	12, 14, 15, 16	∅	∅	∅
0	12, 15	14 16	5 5	∅	1	12, 14, 15, 16	1: 14 2: 16	0 0	5 5
5	12, 15, 47, 45	∅	∅	14, 16	1, 4	12, 14, 15, 16, 47, 45	∅	∅	∅
5	47, 45	12 15	6 6	14, 16	1, 4	12, 14, 15, 16, 47, 45	1: 12 2: 15	5 5	11 11
11	47, 45, 23	∅	∅	14, 16, 12, 15	1, 4, 2	12, 14, 15, 16, 47, 45, 23	∅	∅	∅
11	45	23 47	7 7	14, 16, 12, 15	1, 4, 2	12, 14, 15, 16, 47, 45, 23	1: 23 2: 47	11 11	18 18
18	45, 37, 36	∅	∅	14, 16, 12, 15, 23, 47	1, 4, 2, 3	12, 14, 15, 16, 47, 45, 23, 36, 37	∅	∅	∅
18	45	36 37	5 7	14, 16, 12, 15, 23, 47	1, 4, 2, 3	12, 14, 15, 16, 47, 45, 23, 36, 37	1: 36 2: 37	18 18	23 25
23	45	37	7	14, 16, 12, 15, 23, 47, 36	1, 4, 2, 3	12, 14, 15, 16, 47, 45, 23, 36, 37	2: 37	18	25
23	∅	45 37	9 7	14, 16, 12, 15, 23, 47, 36	1, 4, 2, 3	12, 14, 15, 16, 47, 45, 23, 36, 37	1: 45 2: 37	23 18	31 25
25	∅	45	9	14, 16, 12, 15, 23, 47, 36, 37	1, 4, 2, 3	12, 14, 15, 16, 47, 45, 23, 36, 37	1: 45	23	31
31	56, 58	∅	∅	14, 16, 12, 15, 23, 47, 36, 37, 45	1, 4, 2, 3, 5	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58	∅	∅	∅
31	∅	56 58	5 7	14, 16, 12, 15, 23, 47, 36, 37, 45	1, 4, 2, 3, 5	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58	1: 56 2: 58	31 31	36 38

36	67, 68	58	7	14, 16, 12, 15, 23, 47, 36, 37, 45, 56	1, 4, 2, 3, 5, 6	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68	2: 58	31	38
36	68	67 58	9 7	14, 16, 12, 15, 23, 47, 36, 37, 45, 56	1, 4, 2, 3, 5, 6	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68	1: 67 2: 58	36 31	45 38
38	68	67	9	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58	1, 4, 2, 3, 5, 6	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68	1: 67	36	45
38	∅	67 68	9 9	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58	1, 4, 2, 3, 5, 6	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68	1: 67 2: 68	36 38	45 47
45	78	68	9	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58, 67	1, 4, 2, 3, 5, 6, 7	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68, 78	2: 68	38	47
45	∅	78 68	7 9	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58, 67	1, 4, 2, 3, 5, 6, 7	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68, 78	1: 78 2: 68	45 38	52 47
47	∅	78	7	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58, 67, 68	1, 4, 2, 3, 5, 6, 7	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68, 78	1: 78	45	52
52	89	∅	∅	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58, 67, 68, 78	1, 4, 2, 3, 5, 6, 7, 8	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68, 78, 89	∅	∅	∅
52	∅	89	8	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58, 67, 68, 78	1, 4, 2, 3, 5, 6, 7, 8	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68, 78, 89	1: 89	52	60
60	∅	∅	∅	14, 16, 12, 15, 23, 47, 36, 37, 45, 56, 58, 67, 68, 78, 89	1, 4, 2, 3, 5, 6, 7, 8, 9	12, 14, 15, 16, 47, 45, 23, 36, 37, 56, 58, 67, 68, 78, 89	∅	∅	∅
I(60) = I – конец работы									

По результатам таблицы составим диаграмму Ганта:

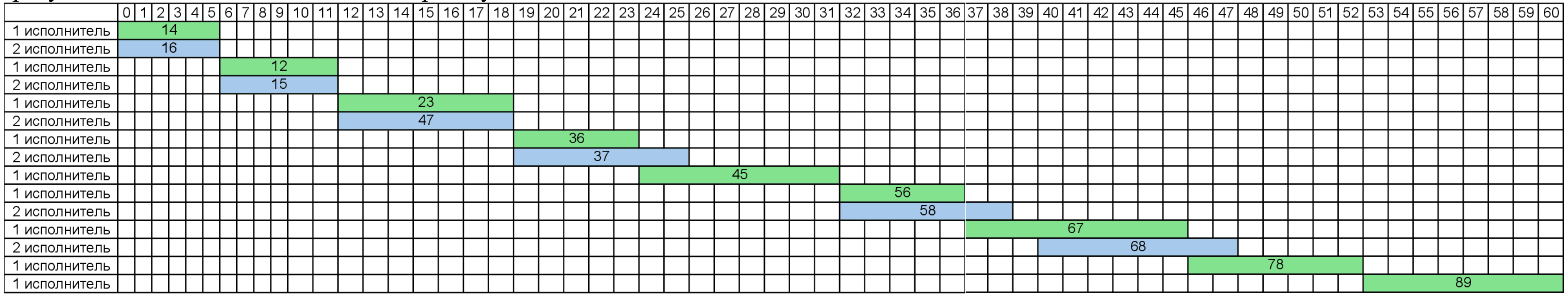

Рис. 2.6. Диаграмма Ганта.

3. Вывод:

В ходе расчетной работы были получены навыки по построению модели расписания, а также применению методов линейного и динамического программирования в рамках поставленной задачи.

4. Приложение:

Листинг на github: https://github.com/DafterT/SADM_6_1