



**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ *Робототехники и комплексной автоматизации*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

## **ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

по дисциплине:

**Разработка PLM**

Тема лабораторной работы **Реализация поиска критического пути**

Студент: **Абидоков Р. Ш.**  
Группа: **РК6-11М.**  
Преподаватель: **Жук Д. М.**

*Москва, 2020*

## Оглавление

Задание на лабораторную работу .....	3
Описание программной реализации .....	3
Примеры работы .....	4

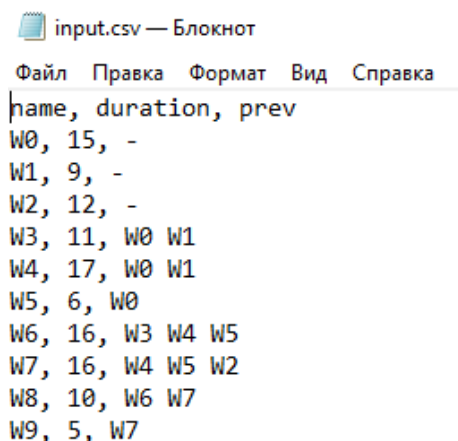
## Задание на лабораторную работу

Реализовать программу, принимающую на вход состав работ (задач) и их взаимосвязи (типа "окончание-начало"), рассчитывающую длину критического пути и запас по времени для работ, разработать структуру входного файла.

### Описание программной реализации

Программа написана на языке Python 3.7.2 с использованием библиотек NumPy и Pandas для удобной работы с массивами и таблицами.

Входные данные считываются из .csv-файла, пример которого показан на Рис. 1. Структура файла следующая: первый столбец – имя работы, второй столбец – продолжительность работы, третий столбец – разделенные пробелами имена работ, которые должны быть выполнены к началу текущей



```
input.csv — Блокнот
Файл  Правка  Формат  Вид  Справка
name, duration, prev
W0, 15, -
W1, 9, -
W2, 12, -
W3, 11, W0 W1
W4, 17, W0 W1
W5, 6, W0
W6, 16, W3 W4 W5
W7, 16, W4 W5 W2
W8, 10, W6 W7
W9, 5, W7
```

Рис. 1 Пример входного файла

Алгоритм работы программы следующий:

1. Происходит парсинг входного файла
2. Строится граф работ в виде матрицы смежности со стороной  $2n + 2$  (т.к. вершинами считаются начала и окончания работ, а также две фиктивных вершины – общее начало и общий конец).

Для каждой работы добавляется ребро между ее началом и концом, величина ребра – длительность работы. Начало работы соединяется нулевыми ребрами с концами тех работ, которые должны быть предварительно выполнены.

3. Производится обход графа от начала к концу – из предположения, что работы топологически отсортированы (т.е. ребра выходят из вершины с меньшим номером к вершине с большим номером) максимальный путь до вершины определяется как наибольшая сумма пути до одной из предыдущих вершин и пути от нее до текущей
4. Производится обход графа от конца к началу – максимальный путь до вершины определяется как наименьшая разность пути до следующей вершины и пути от текущей вершины до следующей
5. Вершины, для которых значения при обоих обходах совпали, помечаются как принадлежащие критическому пути, для остальных разность значений при обходах есть резерв по времени

Полный исходный код программы приведен в приложенных файлах.

## Примеры работы

В качестве пример рассмотрена схема работ, приведенная на Рис. 2

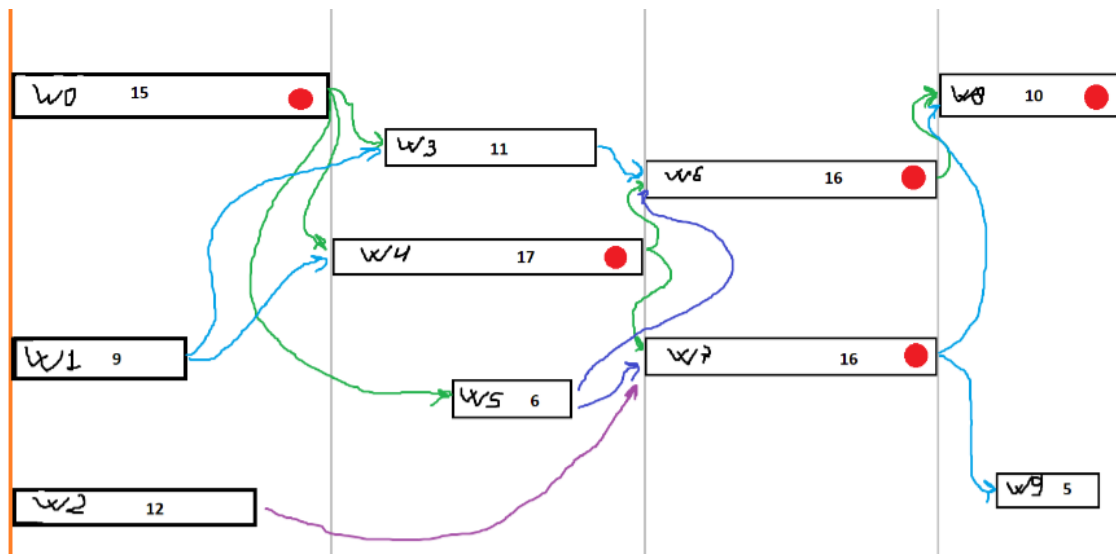


Рис. 2

Результат её обхода, полученный вручную, приведен на Рис. 3

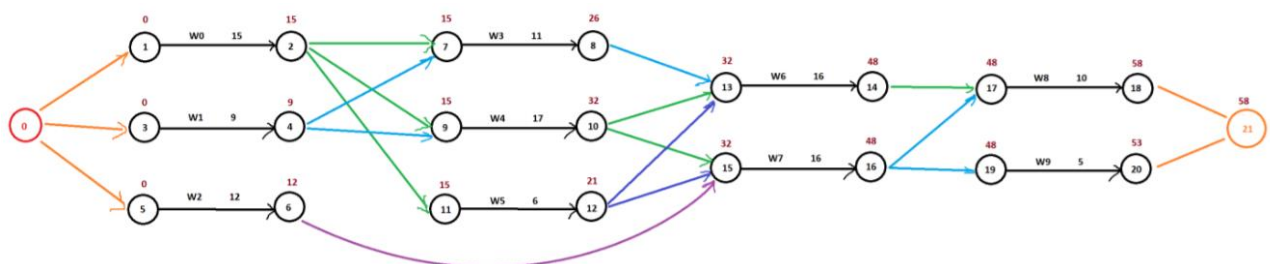


Рис. 3

Результат её обхода в обратную сторону приведен на Рис. 4

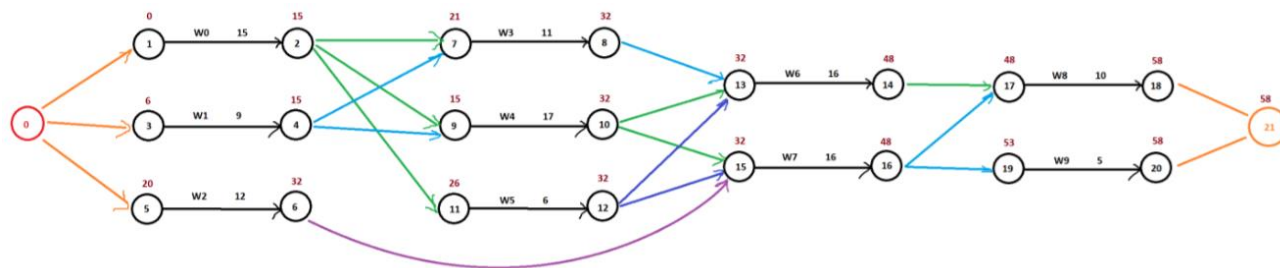


Рис. 4

Вывод программы приведен на Рис. 5

```

Работы, составляющие критический путь:
name duration start finish
0 w0 15 0 15
4 w4 17 15 32
6 w6 16 32 48
7 w7 16 32 48
8 w8 10 48 58

Работы, имеющие запас по времени:
name duration early start late start early finish late finish time margin
1 w1 9 0 6 9 15 6
2 w2 12 0 20 12 32 20
3 w3 11 15 21 26 32 6
5 w5 6 15 26 21 32 11
9 w9 5 48 53 53 58 5

Process returned 0 (0x0) execution time : 0.686 s
Для продолжения нажмите любую клавишу . . .
  
```

Рис. 5