УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

**“ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО”**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 8

по дисциплине «Операционные системы»

На тему **«**Тупиковые ситуации и подходы к их разрешению**»**

Выполнил: студент гр. ИТП-11

Воробьев В.А.

Принял: преподаватель

Карась О. В.

Гомель 2022

**Цель**: изучить причины возникновения тупиковых ситуаций и подходов к их разрешению.

**Ход работы**

**Задание 1. Один ресурс**

В соответствии с вариантом выполнить построение последовательности надежных состояний системы при удовлетворении запросов на ресурсы в соответствии с алгоритмом «Банкира». В соответствии с вариантом имеются следующие исходные данные:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ресурсы | Процесс 1 | Процесс 2 | Процесс 3 | Макс. ресурсов |
| Выдано | 1 | 2 | 0 | 6 |
| Потребность | 5 | 5 | 5 |

Согласно алгоритму, нам нужно продемонстрировать последовательность состояний системы при удовлетворении запросов на ресурсы. Чтобы состояние было надежным, над надо сделать так, чтобы текущее количество выделенного ресурса равнялось количеству максимальных ресурсов. Для этого мы просто некоторому процессу выделить ресурсы, чтобы в итого было нужное нам количество. Когда процесс завершился, то мы в резерв добавляем количество ресурсов, которое он занимал. В итоге мы выполняем шаги алгоритма до тех пор, пока не завершатся все процессы. Для исходных данных результат будет выглядеть следующим образом. Первые 3 шаги показаны на следующем изображении:

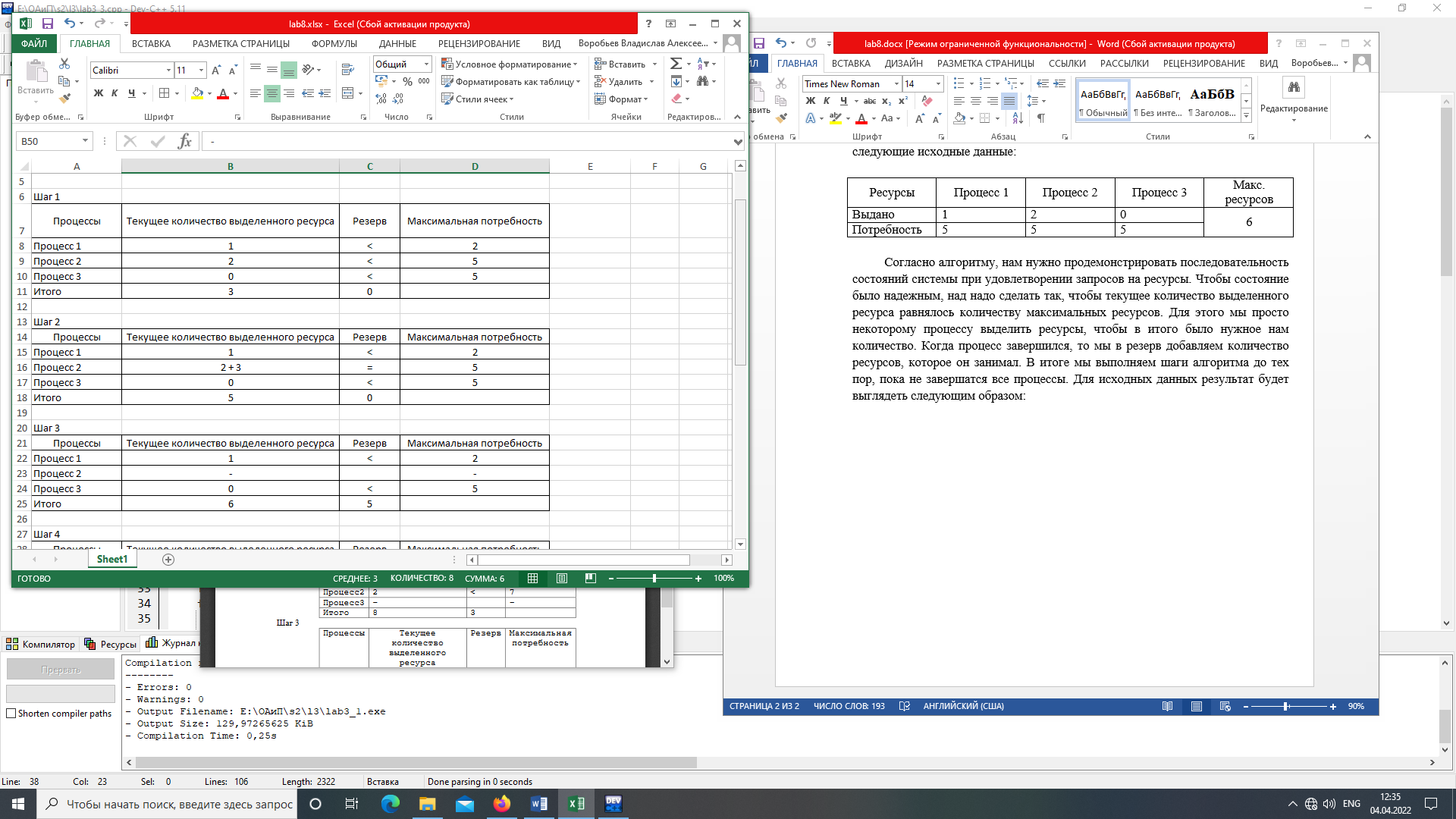


Рисунок 1 – Первые 3 шага алгоритма

На изображении ниже представлены оставшиеся шаги алгоритма:

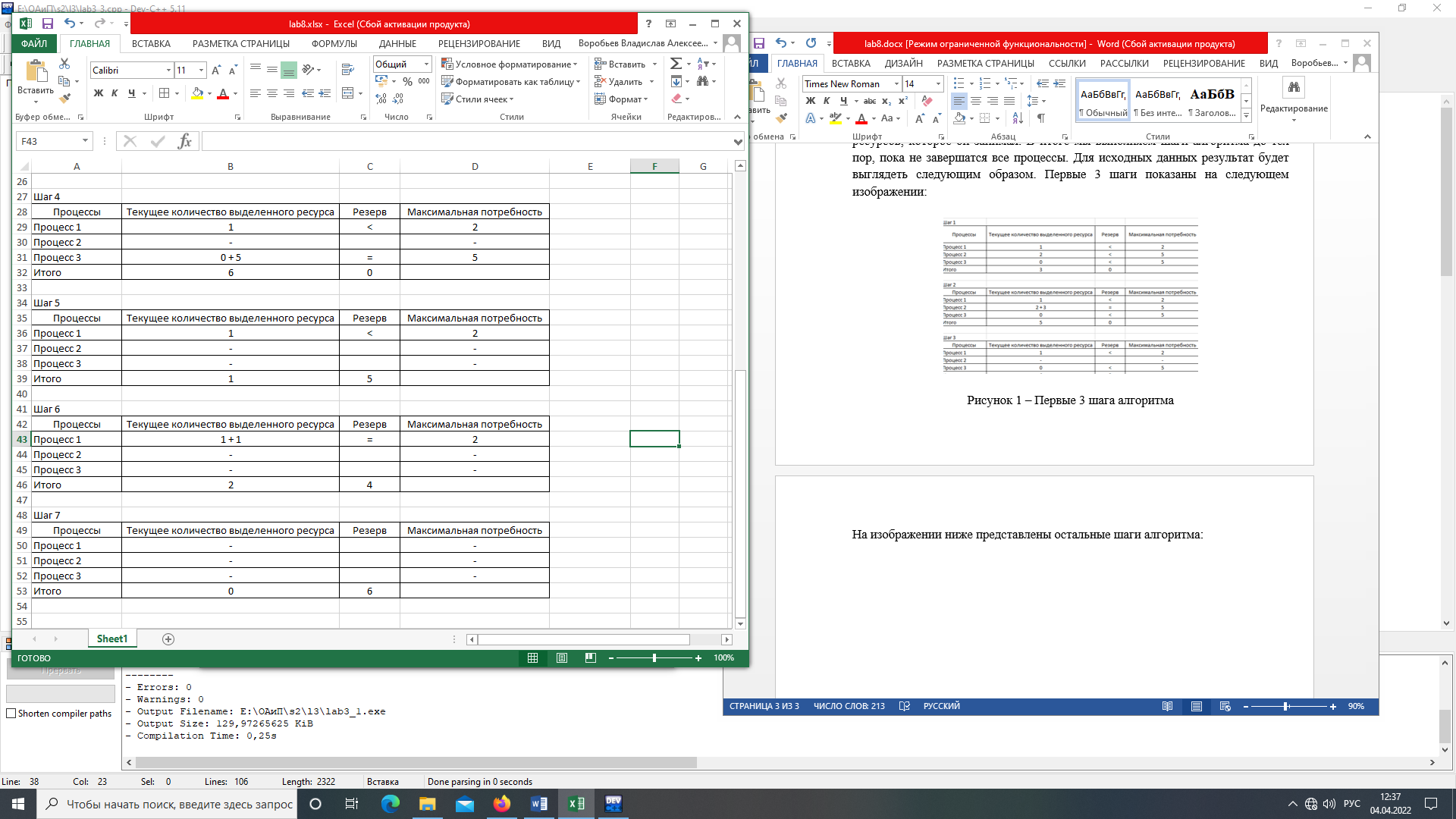


Рисунок 2 – Последние шаги алгоритма

**Задание 2. Несколько ресурсов**

Максимальное количество ресурсов P1 – 7, P2 – 6. Ресурсы выделяются последовательно (в соответствии со значениями, приведенными в таблице). В соответствии с вариантом имеются следующие исходные данные:

|  |  |  |
| --- | --- | --- |
| Процесс | P1 | P2 |
| 1 | 3(1+1+1) | 3(1+0+2) |
| 2 | 2(0+0+2) | 2(0+2+0) |
| 3 | 3(3+0+0) | 6(4+0+2) |
| 4 | 4(1+3+0) | 2(0+2+0) |
| 5 | 4(4+0+0) | 5(5+0+0) |
| 6 | 2(0+0+2) | 3(2+0+1) |

Алгоритм банкира для нескольких ресурсов выглядит следующим образом:

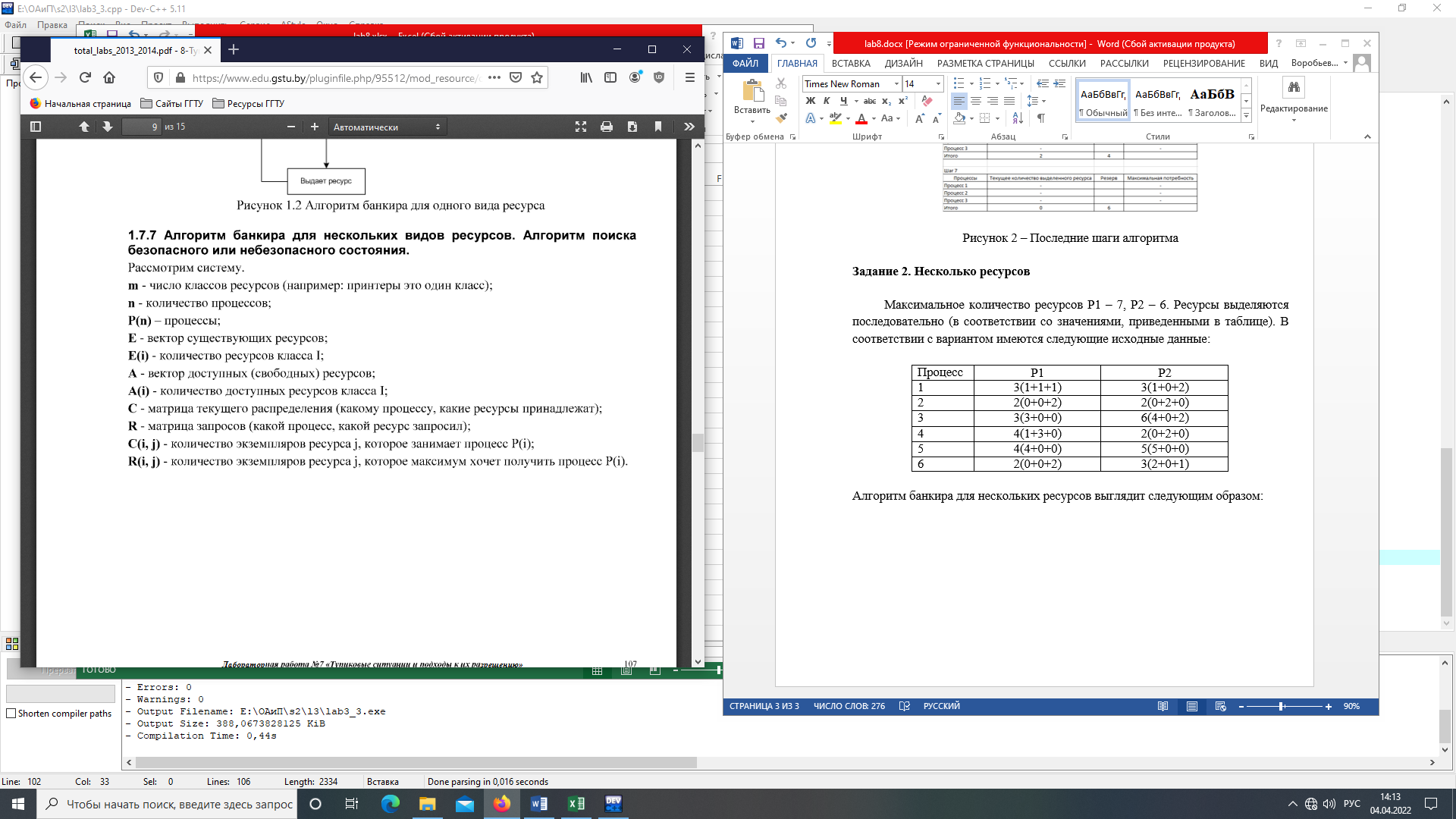


Рисунок 3 – Алгоритм банкира для

нескольких ресурсов

Алгоритм обнаружение тупиков состоит из следующих шагов:

1. Ищем немаркированные процесс Pi, для которого i-я строка матрицы R меньше вектора A или равна ему.
2. Если такой процесс найден, прибавляем f-ю строку матрицы C к вектору А, маркируем процесс и возвращаемся к шагу 1.
3. Если таких процессов не существует, работа алгоритма заканчивается.

Завершение алгоритма означает, что все немаркированные процессы попали в тупик.

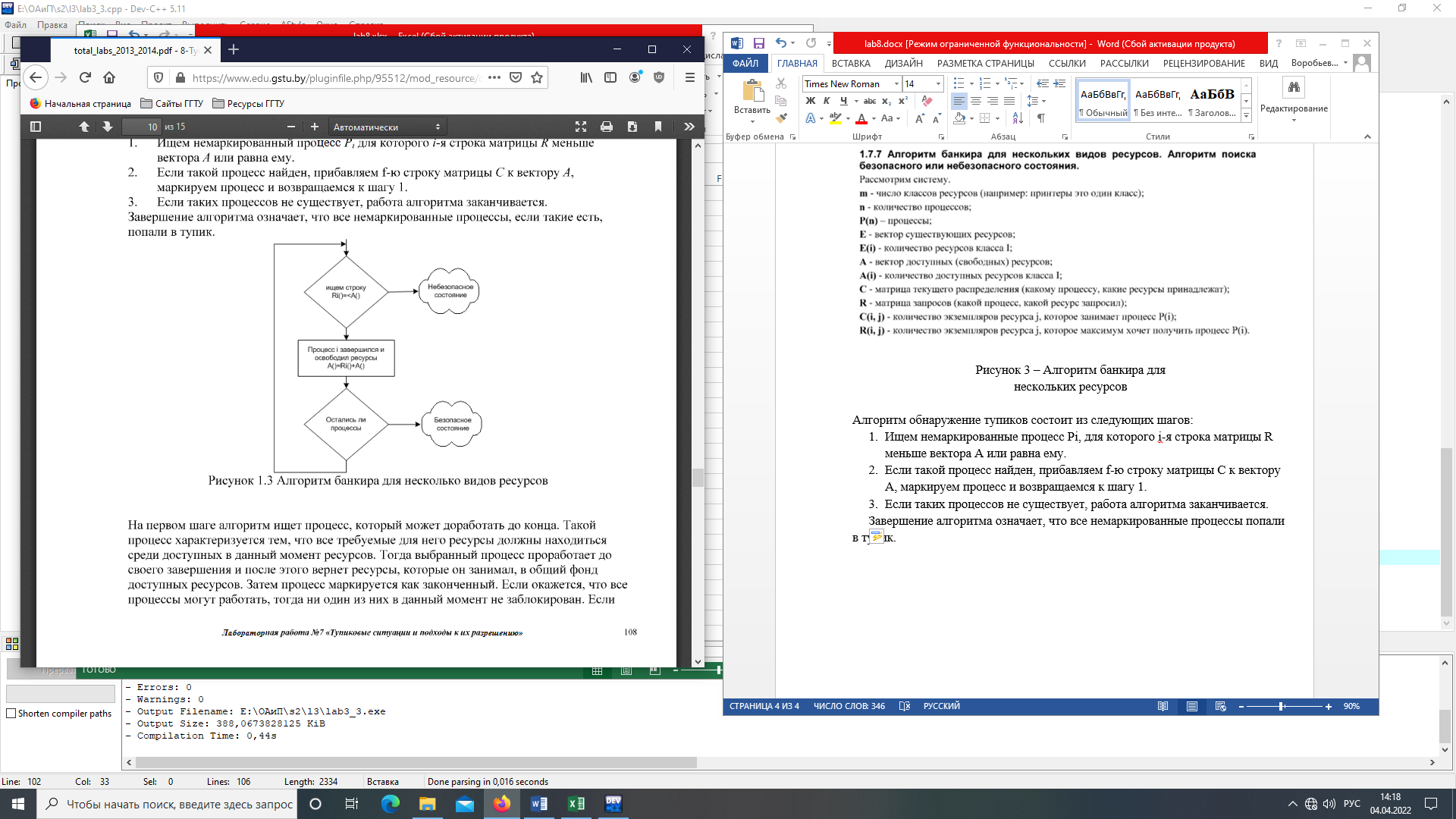


Рисунок 4 – Блок-схема алгоритма банкира для

нескольких ресурсов

Результаты работы алгоритма для исходных данных приведены в книге *lab8*. Если вкратце, то мы выполняем шаги алгоритма до тех пор, пока не закончат свою работу все процессы. Например, на первом шаге мы ищем процессы, которые мы можем на данные момент. Для нас это будут процессы 1 и 4, так как 3 + 4 = 7 и 3 + 2 = 5, а этих ресурсов нам как раз хватает, так максимальное количество P1 – 7, а P2 – 6. И таким образом мы проделаем подобные действия до тех пор, пока все процессы не отработают. У нас в итоге получится 16 шагов.

**Вывод**: в ходе работы были изучены причины возникновения тупиковых ситуаций и подходы к их разрешению.