**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5**

по дисциплине «Операционные системы»

на тему: «Планирование процессов»

Выполнил: студент гр. ИТП-11

Коркуц С.И.

Принял: преподаватель Курочка К.С.

Гомель 2019

**Цель работы:** Изучить алгоритмы планирования. Выполнить планирование задачи согласно варианту.

**Задание 1. Не вытесняющие алгоритмы планирования процессов.**

Выполнить различные алгоритмы планирований – *First-Come, First-Served (FCFS)* (прямой и обратный), *Round Robin (RR), Shortest-Job-First (SJF)* (не вытесняющий), *Shortest-Job-First (SJF)* (не вытесняющий приоритетный), для данных в соответствии со своим вариантом. Вычислить полное время выполнения всех процессов и каждого в отдельности, время ожидания для каждого процесса. Рассчитать среднее время выполнения процесса и среднее время ожидания. Результаты оформить в виде таблиц, иллюстрирующих работу процессов.

Исходные данные для 10 варианта представлены на рисунке 1.





Рисунок 1 – Исходные данные

**Ход выполнения:**

Первым выполнялся алгоритм планирования *First-Come, First-Served (FCFS)* (прямой), т.е. по принципу «первый вошёл, первый вышел».

Результат выполнения планирования представлен на рисунке 2.

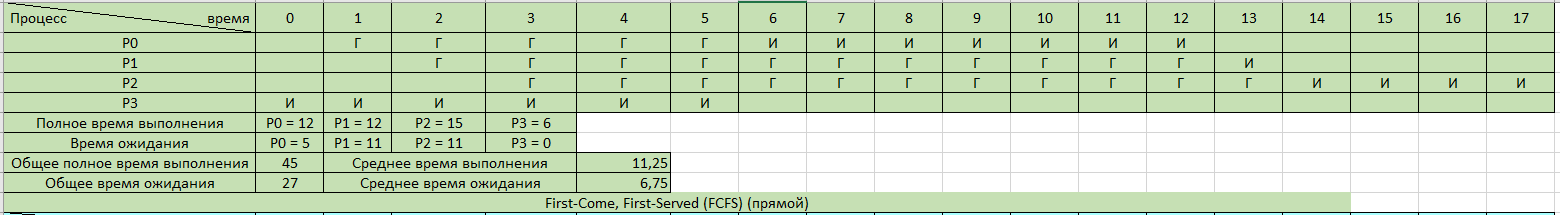


Рисунок 2 – *First-Come, First-Served (FCFS)* (прямой)

Вторым выполнялся алгоритм планирования *First-Come, First-Served (FCFS)* (обратный), который работает по принципу «последний вошёл, первый вышел).

Результат выполнения планирования представлен на рисунке 3.



Рисунок 3 – *First-Come, First-Served (FCFS)* (обратный)

Следующий составляемым алгоритмом планирования был *Round Robin (RR).* Работает по принципу алгоритма *First-Come, First-Served (FCFS)* (прямой). Но величина кванта времени для которого составляет 3.

Результат выполнения планирования представлен на рисунке 4.

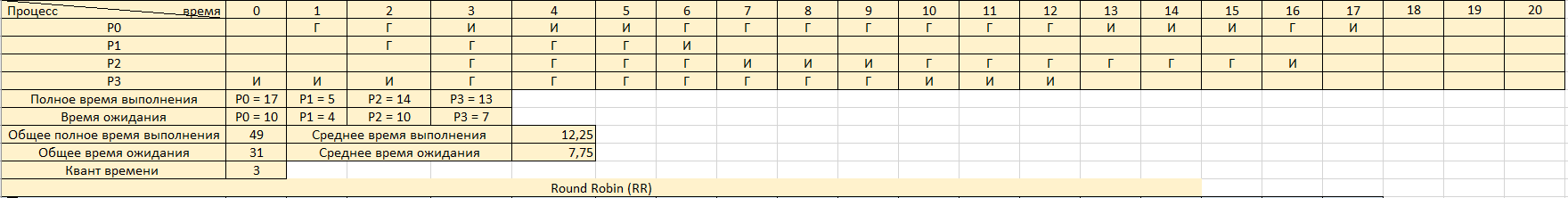


Рисунок 4 – *Round Robin (RR)*

Четвертым выполнялся алгоритм планирования *Shortest-Job-First (SJF)* (не вытесняющий). Работает по принципу «выполняется процесс с наименьшим *CPU Burst* из очереди».

Результат выполнения представлен на рисунке 5.

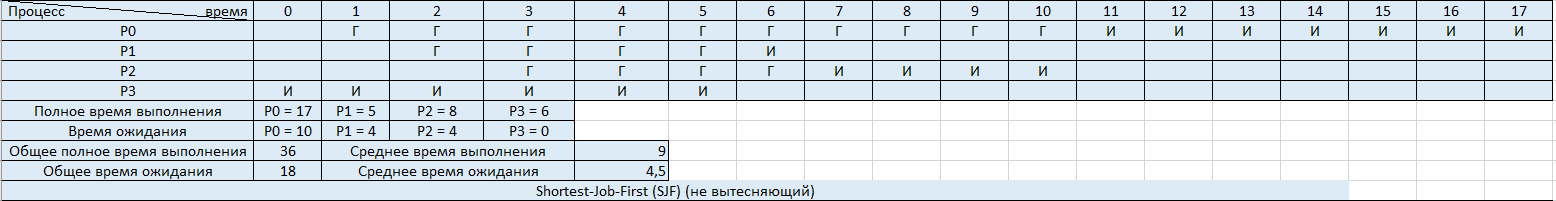


Рисунок 5 – *Shortest-Job-First (SJF)* (не вытесняющий)

Пятым выполнялся алгоритм *Shortest-Job-First (SJF)* (не вытесняющий приоритетный). Работает по принципу *Shortest-Job-First (SJF)* (не вытесняющий), но в случае появления в очереди процесса с более высоким приоритетом, он уступает место ему.

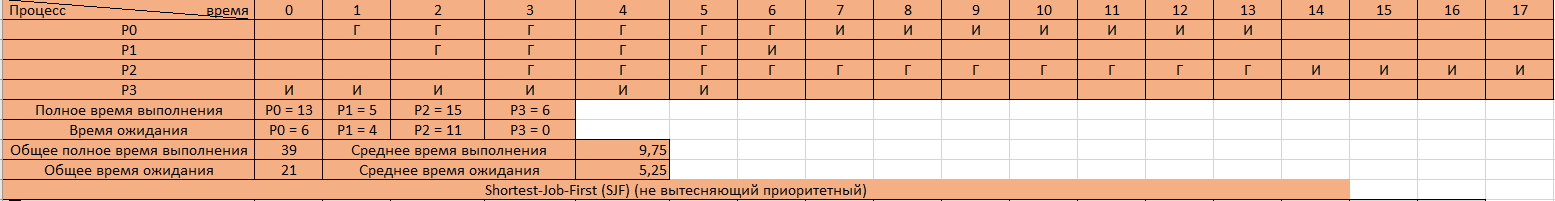


Рисунок 6 – *Shortest-Job-First (SJF)* (не вытесняющий приоритетный)

**Задание 2. Вытесняющие алгоритмы планирования процессов.**

Выполнить различные алгоритмы планирований –*Shortest-Job-First (SJF)* (вытесняющий), *Shortest-Job-First (SJF)* (приоритетный), для данных в соответствии со своим вариантом. Вычислить полное время выполнения всех процессов и каждого в отдельности, время ожидания для каждого процесса. Рассчитать среднее время выполнения процесса и среднее время ожидания. Результаты оформить в виде таблиц, иллюстрирующих работу процессов.

**Ход выполнения:**

Первым выполнялся алгоритм *Shortest-Job-First (SJF)* (вытесняющий). Принцип его работы в следующем: если приходит процесс, *CPU Burst* которого меньше остаточного *CPU Burst* исполняемого процесса, то он его вытесняет.

Результат выполнения представлен на рисунке 7.

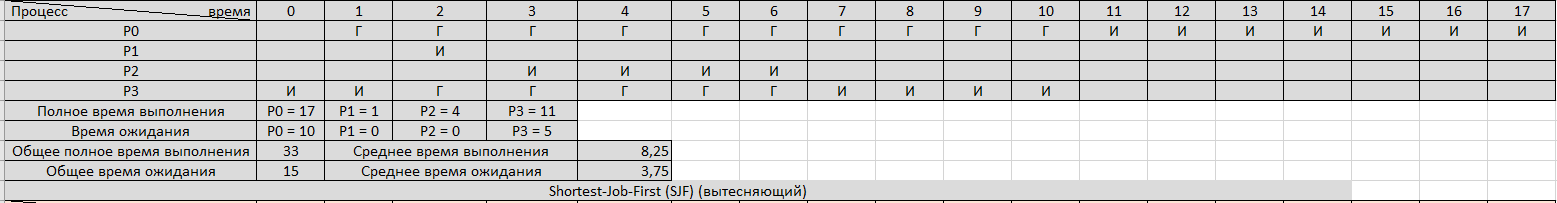


Рисунок 7 – *Shortest-Job-First (SJF)* (вытесняющий)

Вторым выполнялся алгоритм *Shortest-Job-First (SJF)* (приоритетный). Принцип его работы в следующем: если приходит в очередь процесс, у которого приоритет выше исполняемого, то он вытесняет исполняемый процесс.

Результат выполнения представлен на рисунке 8.



Рисунок 8 – *Shortest-Job-First (SJF)* (вытесняющий приоритетный)

**Вывод работы:** в результате выполнения работы были изучены алгоритмы планирования и теоретический материал. Выполнено задание согласно варианту.