



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 \_\_\_\_\_ «Информатика и системы управления»

КАФЕДРА \_\_\_\_\_ «Теоретическая информатика и компьютерные технологии»

**Лабораторная работа №4**  
**по курсу «Разработка параллельных и распределенных**  
**программ»**  
**«Задача о пяти обедающих философах.»**

Студент группы ИУ9-51Б Киселев К.А.

Преподаватель Царев А.С.

*Москва 2023*

# Содержание

<b>1</b>	<b>Постановка задачи</b>	<b>2</b>
<b>2</b>	<b>Практическая реализация</b>	<b>4</b>
<b>3</b>	<b>Результаты</b>	<b>7</b>
3.1	Результат для первого философа . . . . .	7
3.2	Результат для второго философа . . . . .	8
3.3	Результат для третьего философа . . . . .	9
3.4	Результат для четвертого философа . . . . .	10
3.5	Результат для пятого философа . . . . .	11

# 1 Постановка задачи

Суть задачи следующая. Пять философов сидят за круглым столом. Они проводят жизнь, чередуя приёмы пищи и размышления. В центра стола находится большое блюдо спагетти. Чтобы съесть порцию, каждому философу нужно две вилки. Однако, вилок всего пять: между каждой парой рядом сидящих философов лежат по одной вилке, и каждый философ может пользоваться только теми вилками, которые лежат рядом с ним, слева и справа. Философ не может брать две вилки одновременно: сначала он тратит некоторое время на то, чтобы взять одну, затем вторую. Однако, он может одновременно положить их на место. Задача заключается в том, чтобы написать программу, моделирующую поведение философов. Очевидно, что раз вилок всего пять, то одновременно есть могут не более двух философов, и два сидящих рядом философа не могут есть одновременно. Для имитации периодов раздумий и приёмов пищи можно использовать генератор случайных чисел, позволяющий задавать времена их действий в определённом интервале. Имитация поведения каждого философа, по сути, разбивается на то, что в любой момент времени философ находится в одном из пяти состояний: размышляет, берёт левую вилку, берёт правую вилку, ест, кладёт вилки на место. Таким образом, вилки являются разделяемым ресурсом. На программу накладываются условия:

1. Каждый философ, по сути, является потоком, и модель поведения у каждого из них должна быть одинаковой, кроме того, какие вилки они могут брать.
2. Накладывание блокировки по сути является действием по взятию вилки, поэтому накладывать блокировку сразу на обе вилки нельзя; последовательность действий должна быть «наложить блокировку – взять вилку – наложить вторую блокировку – взять вторую вилку».
3. Программа должна избегать ситуации взаимоблокировки: ситуации, в которой все философы голодны, то есть ни один из них не может взять себе две вилки (например, когда каждый держит по одной и не хочет её отдавать). Запрограммировать остановку алгоритма по достижении контрольного времени (например, атомарной операцией над булевым флагом). В отчёте построить некоторый результат работы алгоритма,

которая может быть в виде графика, таблицы, лога или чего угодно ещё; главное условие состоит в том, чтобы по результатам можно было однозначно определить, чем в каждый момент времени был занят каждый философ (одно из пяти состояний).

Также рассмотреть вариант программы с увеличением количества философов до произвольного  $N$ .

## 2 Практическая реализация

---

```
1 package worker
2
3 import (
4     "context"
5     "math/rand"
6     "sync"
7     "time"
8
9     "github.com/rs/zerolog"
10 )
11
12 type WorkerState int
13
14 const (
15     Work WorkerState = iota
16     LockedFirstRes
17     LockedSecondRes
18     FreeResources
19     Workless
20 )
21
22 func (s WorkerState) String() string {
23     switch s {
24     case Work:
25         return "working"
26     case LockedFirstRes:
27         return "locked first resource"
28     case LockedSecondRes:
29         return "locked second resource"
30     case FreeResources:
31         return "freed resources"
32     case Workless:
33         return "not working"
34     default:
35         return "unknown state"
36     }
37 }
38
39 type Worker struct {
40     num      int
41     ctx      context.Context
42     firstRes *sync.Mutex
43     secondRes *sync.Mutex
44     state    WorkerState
45     logger   *zerolog.Logger
46 }
47
```

```

48 func NewWorker(workerNum int, logger *zerolog.Logger, ctx context.Context, firstRes, secondRes *sync.Mu
49     return Worker{
50         num:      workerNum,
51         ctx:      ctx,
52         firstRes:  firstRes,
53         secondRes: secondRes,
54         state:     Workless,
55         logger:    logger,
56     }
57 }
58
59 func (w *Worker) logState() {
60     w.logger.Info().Int("worker_num", w.num).Str("worker_state", w.state.String()).Send()
61     <-time.After(1 * time.Millisecond)
62 }
63
64 func (w *Worker) do(state WorkerState) {
65     t := rand.Intn(10) + 1
66     w.state = state
67
68     ch := time.After(time.Duration(t) * time.Millisecond)
69
70     for {
71         select {
72             case <-ch:
73                 return
74             case <-w.ctx.Done():
75                 return
76             default:
77                 w.logState()
78         }
79     }
80 }
81
82 func (w *Worker) freeResources() {
83     w.secondRes.Unlock()
84     w.firstRes.Unlock()
85     w.state = FreeResources
86     w.logState()
87 }
88
89 func (w *Worker) Run() {
90     for {
91         select {
92             case <-w.ctx.Done():
93                 return
94             default:
95                 for !w.firstRes.TryLock() {
96                     w.do(Workless)
97                 }

```

```
98
99     w.state = LockedFirstRes
100     w.logState()
101
102     for !w.secondRes.TryLock() {
103         w.do(Workless)
104     }
105
106     w.state = LockedSecondRes
107     w.logState()
108
109     w.do(Work)
110     w.freeResources()
111
112 }
113 }
114 }
```

---

## 3 Результаты

Результаты работы программы для  $N = 5$ . Время работы - 25 миллисекунд. Длительность приема пищи или раздумий выбирается случайно в диапазоне от 1 до 10 мс.

### 3.1 Результат для первого философа

---

```
1 {"level":"info","worker_num":0,"worker_state":"locked first resource","time":1701800489080}
2 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489081}
3 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489082}
4 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489083}
5 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489084}
6 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489085}
7 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489086}
8 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489087}
9 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489089}
10 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489090}
11 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489091}
12 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489092}
13 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489093}
14 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489094}
15 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489095}
16 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489096}
17 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489097}
18 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489099}
19 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489100}
20 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489101}
21 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489102}
22 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489103}
23 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489104}
24 {"level":"info","worker_num":0,"worker_state":"not working","time":1701800489105}
25 {"level":"info","worker_num":0,"worker_state":"locked second resource","time":1701800489109}
26 {"level":"info","worker_num":0,"worker_state":"freed resources","time":1701800489110}
```

---



## 3.2 Результат для второго философа

---

```
1 {"level":"info","worker_num":1,"worker_state":"locked first resource","time":1701800489080}
2 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489082}
3 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489083}
4 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489084}
5 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489085}
6 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489086}
7 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489088}
8 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489089}
9 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489090}
10 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489091}
11 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489092}
12 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489093}
13 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489094}
14 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489095}
15 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489096}
16 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489097}
17 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489098}
18 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489100}
19 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489101}
20 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489102}
21 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489103}
22 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489104}
23 {"level":"info","worker_num":1,"worker_state":"not working","time":1701800489105}
24 {"level":"info","worker_num":1,"worker_state":"locked second resource","time":1701800489107}
25 {"level":"info","worker_num":1,"worker_state":"freed resources","time":1701800489108}
```

---

### 3.3 Результат для третьего философа

---

```
1 {"level":"info","worker_num":2,"worker_state":"locked first resource","time":1701800489080}
2 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489082}
3 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489083}
4 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489084}
5 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489085}
6 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489086}
7 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489088}
8 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489089}
9 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489090}
10 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489091}
11 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489092}
12 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489093}
13 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489094}
14 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489095}
15 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489096}
16 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489097}
17 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489098}
18 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489100}
19 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489101}
20 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489102}
21 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489103}
22 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489104}
23 {"level":"info","worker_num":2,"worker_state":"not working","time":1701800489105}
24 {"level":"info","worker_num":2,"worker_state":"locked second resource","time":1701800489106}
25 {"level":"info","worker_num":2,"worker_state":"freed resources","time":1701800489107}
```

---

## 3.4 Результат для четвертого философа

---

```
1 {"level":"info","worker_num":3,"worker_state":"locked first resource","time":1701800489080}
2 {"level":"info","worker_num":3,"worker_state":"locked second resource","time":1701800489081}
3 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489082}
4 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489083}
5 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489084}
6 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489085}
7 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489086}
8 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489088}
9 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489089}
10 {"level":"info","worker_num":3,"worker_state":"freed resources","time":1701800489090}
11 {"level":"info","worker_num":3,"worker_state":"locked first resource","time":1701800489091}
12 {"level":"info","worker_num":3,"worker_state":"locked second resource","time":1701800489092}
13 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489093}
14 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489094}
15 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489095}
16 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489096}
17 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489097}
18 {"level":"info","worker_num":3,"worker_state":"freed resources","time":1701800489098}
19 {"level":"info","worker_num":3,"worker_state":"locked first resource","time":1701800489100}
20 {"level":"info","worker_num":3,"worker_state":"locked second resource","time":1701800489101}
21 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489102}
22 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489103}
23 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489104}
24 {"level":"info","worker_num":3,"worker_state":"working","time":1701800489105}
25 {"level":"info","worker_num":3,"worker_state":"freed resources","time":1701800489106}
```

---

## 3.5 Результат для пятого философа

---

```
1 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489080}
2 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489081}
3 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489082}
4 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489083}
5 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489084}
6 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489085}
7 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489086}
8 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489087}
9 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489089}
10 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489090}
11 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489091}
12 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489092}
13 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489093}
14 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489094}
15 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489095}
16 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489096}
17 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489097}
18 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489098}
19 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489100}
20 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489101}
21 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489102}
22 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489103}
23 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489104}
24 {"level":"info","worker_num":4,"worker_state":"not working","time":1701800489105}
25 {"level":"info","worker_num":4,"worker_state":"locked first resource","time":1701800489110}
26 {"level":"info","worker_num":4,"worker_state":"locked second resource","time":1701800489111}
27 {"level":"info","worker_num":4,"worker_state":"freed resources","time":1701800489113}
```

---