

## Исходный алгоритм

### НАЧАЛЬНОЕ СОСТОЯНИЕ:

1. Рабочие(акторы) образуют цепочку, начинающуюся рабочим типа Начальный, заканчивающуюся рабочим типа Конечный.  
Внутри цепочки находятся рабочие типа Средний. Цепь из рабочих располагается слева-направо.
2. У рабочего типа Начальный имеется непустая куча кирпичей.
3. Рабочий типа Средний получил подтверждение о готовности взять кирпич от рабочего справа и сам готов взять кирпич у рабочего слева. Рабочий типа Конечный готов взять кирпич от рабочего слева.
4. Рабочий типа Начальный взял первый кирпич и подал его рабочему типа Средний справа (который готов взять кирпич).

ВХОД: сообщение - поданный кирпич (слева) или готовность взять кирпич (справа);  
актор - рабочий, получающий это "сообщение".

1. Если тип рабочего - Начальный.
  - 1.1. Если куча не пустая, то взять кирпич из кучи (иначе ничего не делать) и выполнять 1.1.1.
    - 1.1.1. Подать кирпич рабочему справа.
2. Если тип рабочего - Средний.
  - 2.1. Если рабочий слева подал кирпич и рабочий справа готов взять кирпич (иначе ничего не делать), то выполнять 2.1.\*.
    - 2.1.1. Взять кирпич у рабочего слева, подать кирпич рабочему справа.
    - 2.1.2. Показать рабочему слева, что готов взять следующих кирпич.
3. Если тип рабочего - Конечный.
  - 3.1. Взять кирпич у рабочего слева, положить в новую кучу.
  - 3.2. Если только что положенный кирпич - не последний, то показать рабочему слева, что готов взять следующий кирпич.
  - 3.3. Если только что положенный кирпич - последний, то КОНЕЦ работы алгоритма.

Алгоритм "с тележками" (сообщениями).

-----  
НАЧАЛЬНОЕ СОСТОЯНИЕ:

1. Рабочие образуют цепочку, начинающуюся рабочим типа Начальный, заканчивающуюся рабочим типа Конечный. Внутри цепочки находятся рабочие типа Средний. Цепь из рабочих располагается слева-направо.
2. У рабочего типа Начальный имеется непустая куча кирпичей.
3. У каждого рабочего, кроме рабочего типа Конечный, имеется тележка (объект-сообщение). Тележка может находиться рядом с некоторым рабочим (`access(тележка)=true`) или катиться к рабочему (`access(тележка)=false`). Чтобы передать кирпич рабочий кладет его на тележку и толкает тележку некоторому рабочему (`send(тележка, рабочий)`). Чтобы получить следующий кирпич рабочий толкает пустую тележку назад. Все тележки в начале пусты.
3. Тележки рабочих типа Средний находятся у своих владельцев.
4. Рабочий типа Начальный взял первый кирпич, положил в свою тележку и толкнул её рабочему типа Средний справа от него.

ВХОД: сообщение - пустая тележка или тележка с кирпичом;  
актор - рабочий, к которому подкатилась тележка.

1. Если тип рабочего - Начальный.
  - 1.1. Если куча не пустая, то взять кирпич из кучи (иначе ничего не делать) и выполнять 1.1.\*.
  - 1.1.1. Положить кирпич в тележку.
  - 1.1.2. Толкнуть тележку рабочему справа.
2. Если тип рабочего - Средний.
  - 2.1. Если подъехала тележка от рабочего слева и подъехала (или стоит в начале) собственная тележка, то выполнять 2.1.\*.
  - 2.1.1. Взять кирпич из тележки слева, толкнуть тележку рабочему слева.
  - 2.1.2. Обработать кирпич (это - выполнение задачи).
  - 2.1.3. Положить кирпич в свою тележку, толкнуть её рабочему справа.
3. Если тип рабочего - Конечный.
  - 3.1. Взять кирпич из тележки, положить в новую кучу.
  - 3.2. Если только что положенный кирпич - не последний, то толкнуть пустую тележку рабочему слева.
  - 3.3. Если только что положенный кирпич - последний, то КОНЕЦ работы алгоритма.