Задача 9. Задача на поиск максимума в массиве. Разобрать, в чем эффективность предложенной на лекции параллельной программы (из учебника Эндрюса) с повторной проверкой.

Исследование провёл студент группы 22207 Гордеев Никита

Дата выполнения работы 25.12.2022 (Вариант 2)

Цель

- **Разобрать, в чем эффективность предложенной на лекции параллельной программы “Поиск максимума в массиве” (из учебника Эндрюса) с повторной проверкой.**

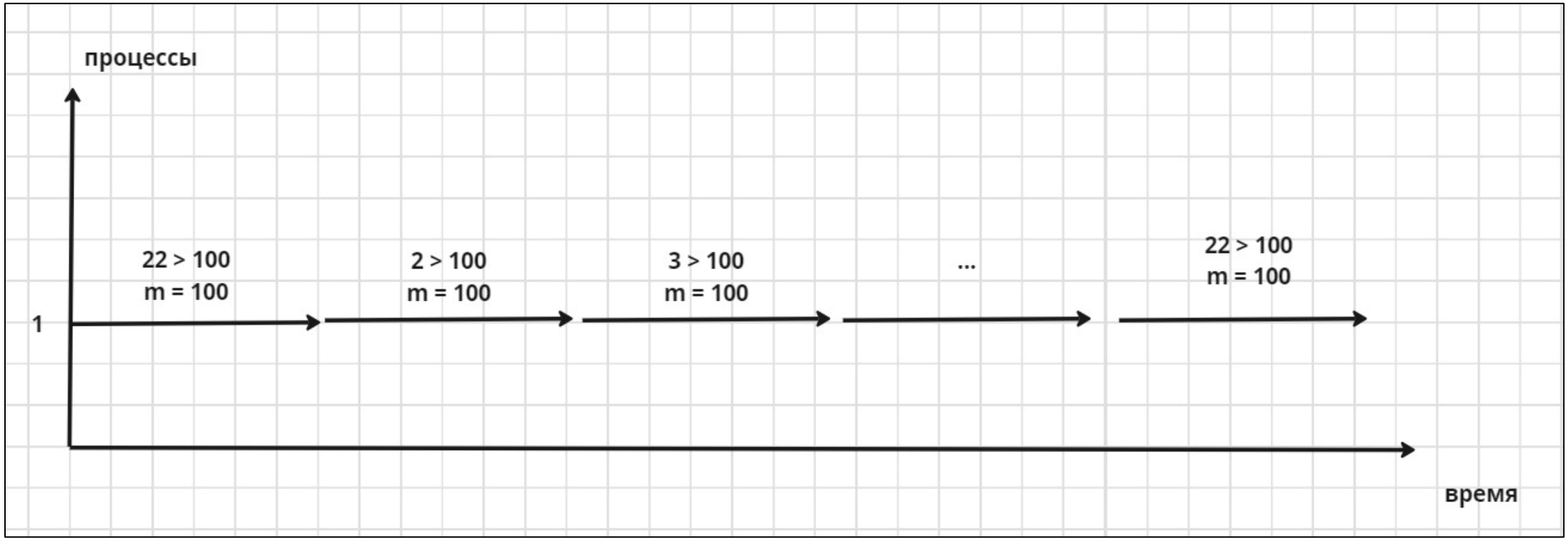
Задача

- **Дан массив a[n] из целых неотрицательных чисел.**
- **Требуется найти m – максимальное число среди a[i], $i = \overline{1, n}$**

Последовательная программа

```
int m = 0;
for [i = 1 to n] {  # по всем элементам массива
    if (a[i] > m)  # текущий элемент больше максимального?
        m = a[i];  # если да, то меняем максимальный
}
```

Схема параллельного алгоритма



Параллельная программа

Идея:

Сначала проверить неравенство, а затем, если оно выполняется, провести ещё одну проверку перед обновлением значения переменной.

```
int m = 0;
co [i = 0 to n-1]          # создаём n параллельных процессов
  if (a[i] > m)            #проверка значения m
    ⟨if (a[i] > m)         #перепроверка значения m
      m = a[i];⟩
```

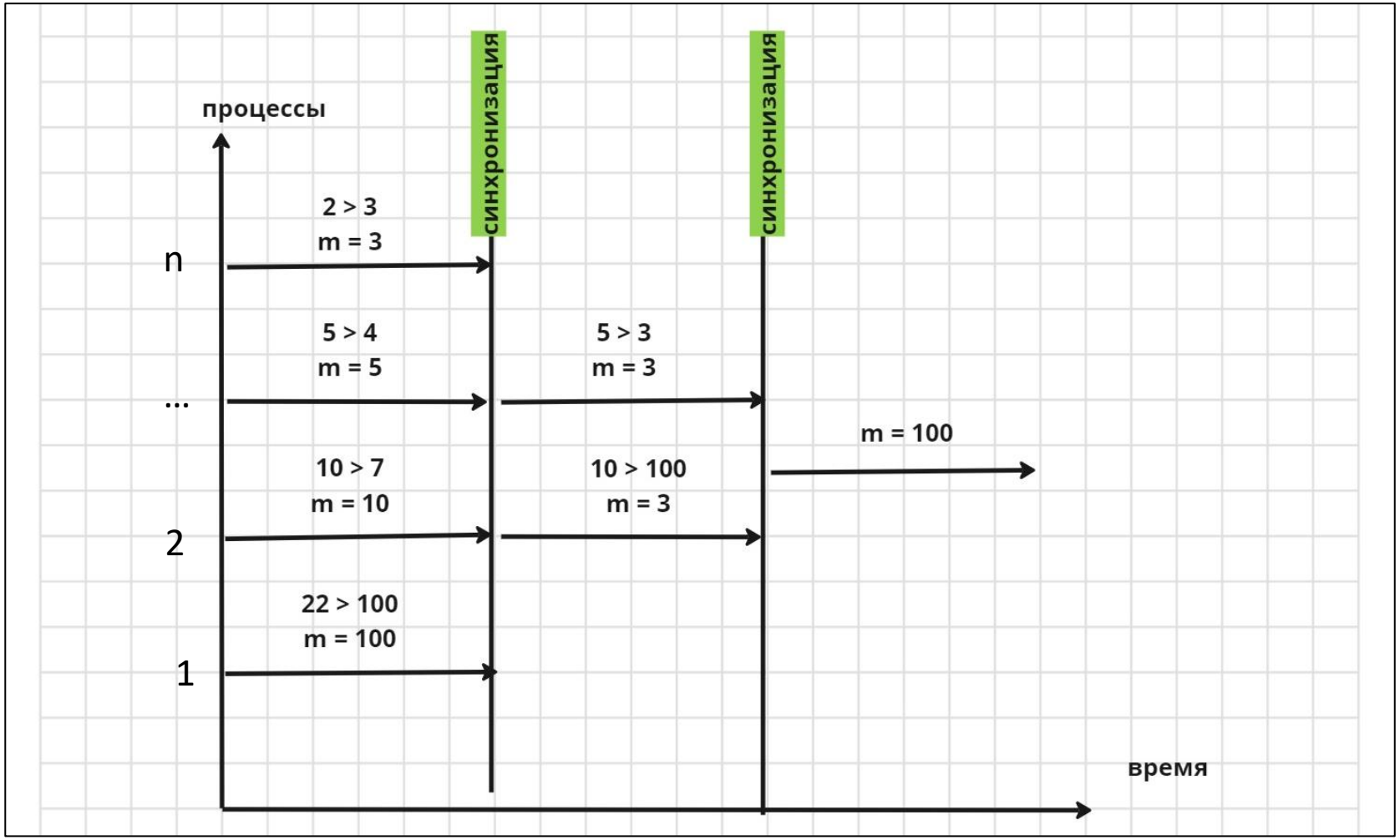
Общий анализ алгоритма

- **Первая проверка может осуществляться одновременно несколькими процессами, что сокращает общее время работы**
- **Вторая проверка утверждает корректность получаемого результата**
- **Временная сложность параллельного алгоритма будет меньше за счёт того, что первые проверки проводятся параллельно, и существенная часть из них будет проваливаться.**

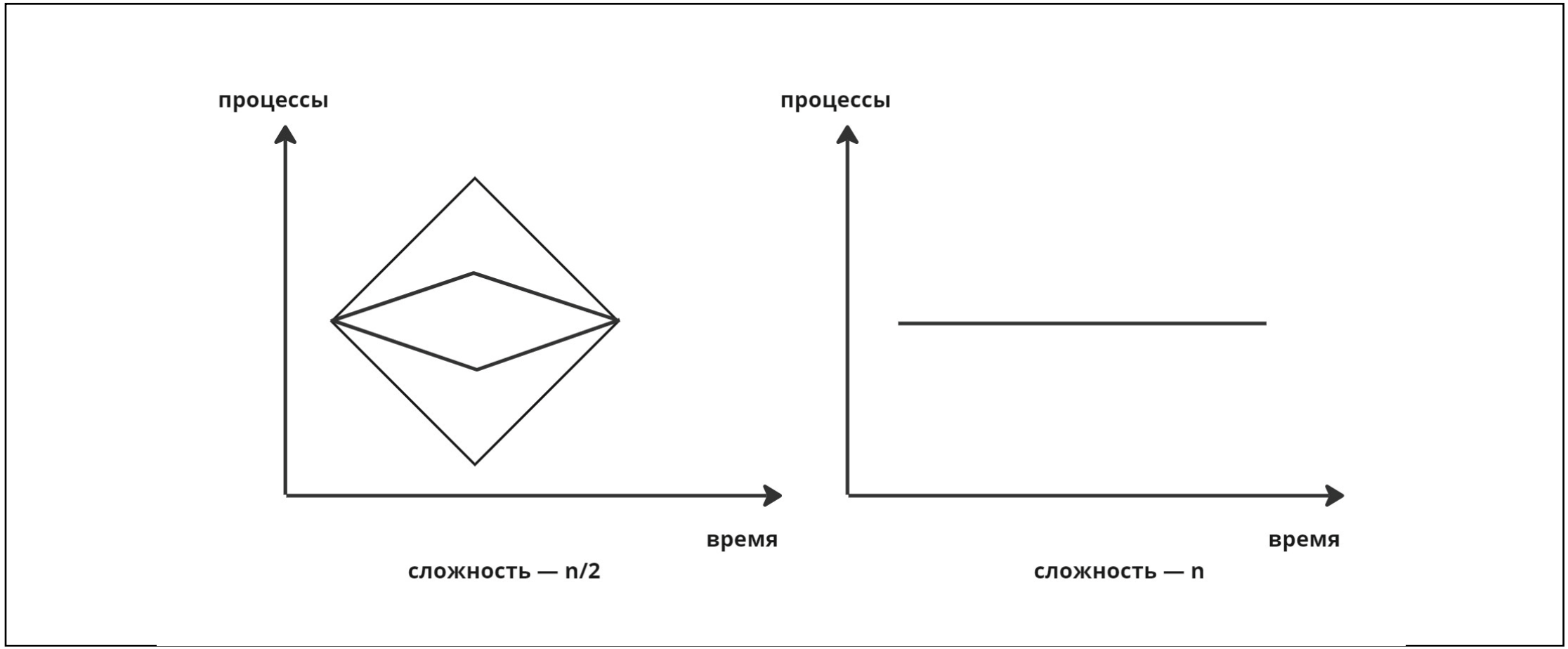
Анализ двойной проверки

- **Многие процессы закончат работу в результате первой проверки и не будут выполнять вторую, т.к. их опередят процессы с большим значением a[i]. Таким образом, дорогая операция присваивания будет выполняться достаточно редко.**

Схема параллельного алгоритма



Временная сложность



Материалы:

- **2.2 Распараллеливание: поиск образца в файле // Грегори Р. Эндрюс - Основы многопоточного, параллельного и распределенного программирования (дата обращения: 18.12.2022).**
- **2.3 Синхронизация: поиск максимального элемента массива // Грегори Р. Эндрюс - Основы многопоточного, параллельного и распределенного программирования (дата обращения: 18.12.2022).**
- **Самый быстрый алгоритм поиска максимума в массиве // proglib URL: https://proglib.io/p/fastest-max-algorithm (дата обращения: 18.12.2022).**

Изменения

- **Версия 2**
  - **Добавил последовательную версию программы**
  - **Добавил в начало цели и задачи**
  - **Добавил проверки и схемы проверок**