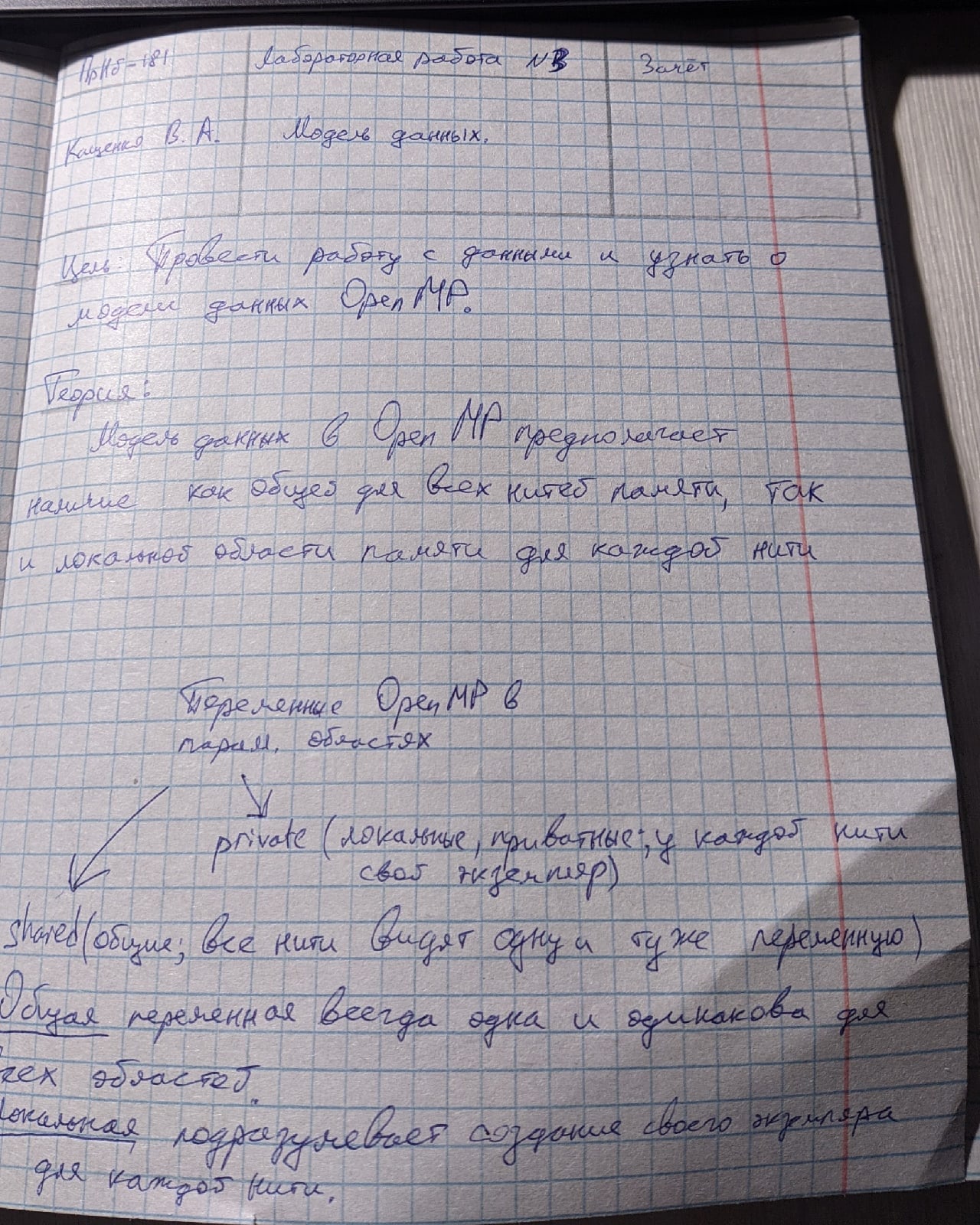
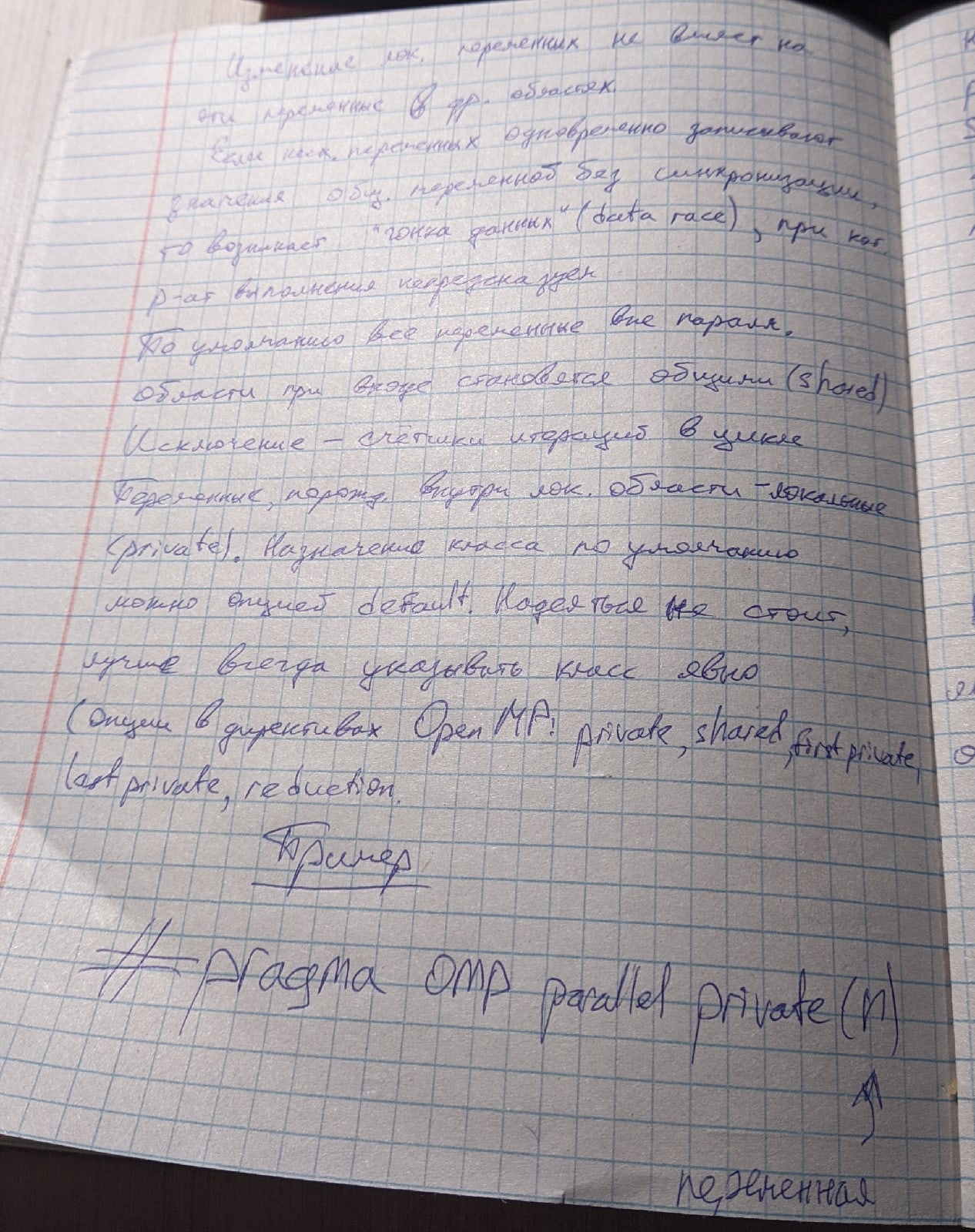
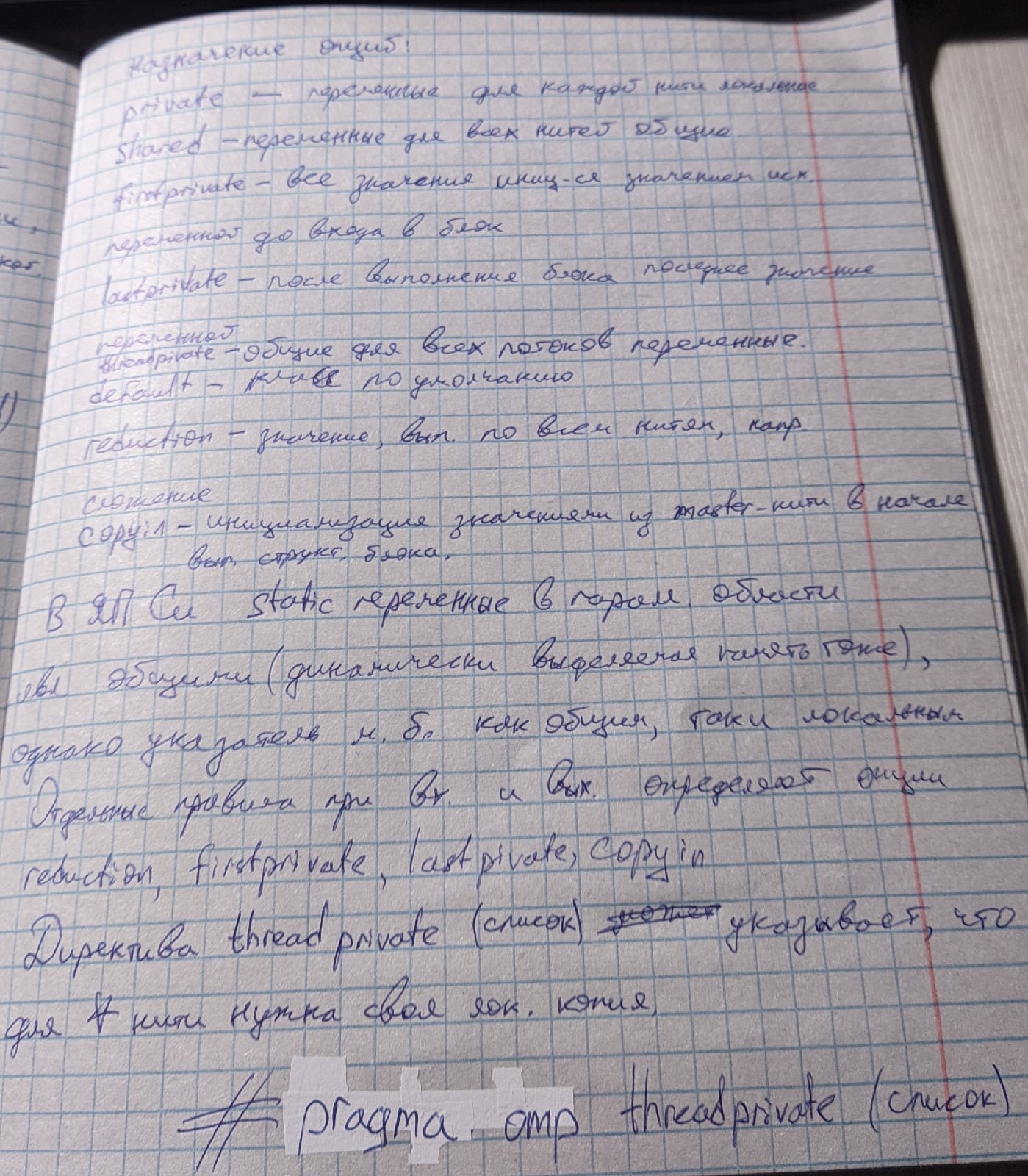
|  |  |  |
| --- | --- | --- |
| ПрИб-181 | Лабораторная работа №3 | Зачёт |
| Кащенко В. А. | Модель данных |  |

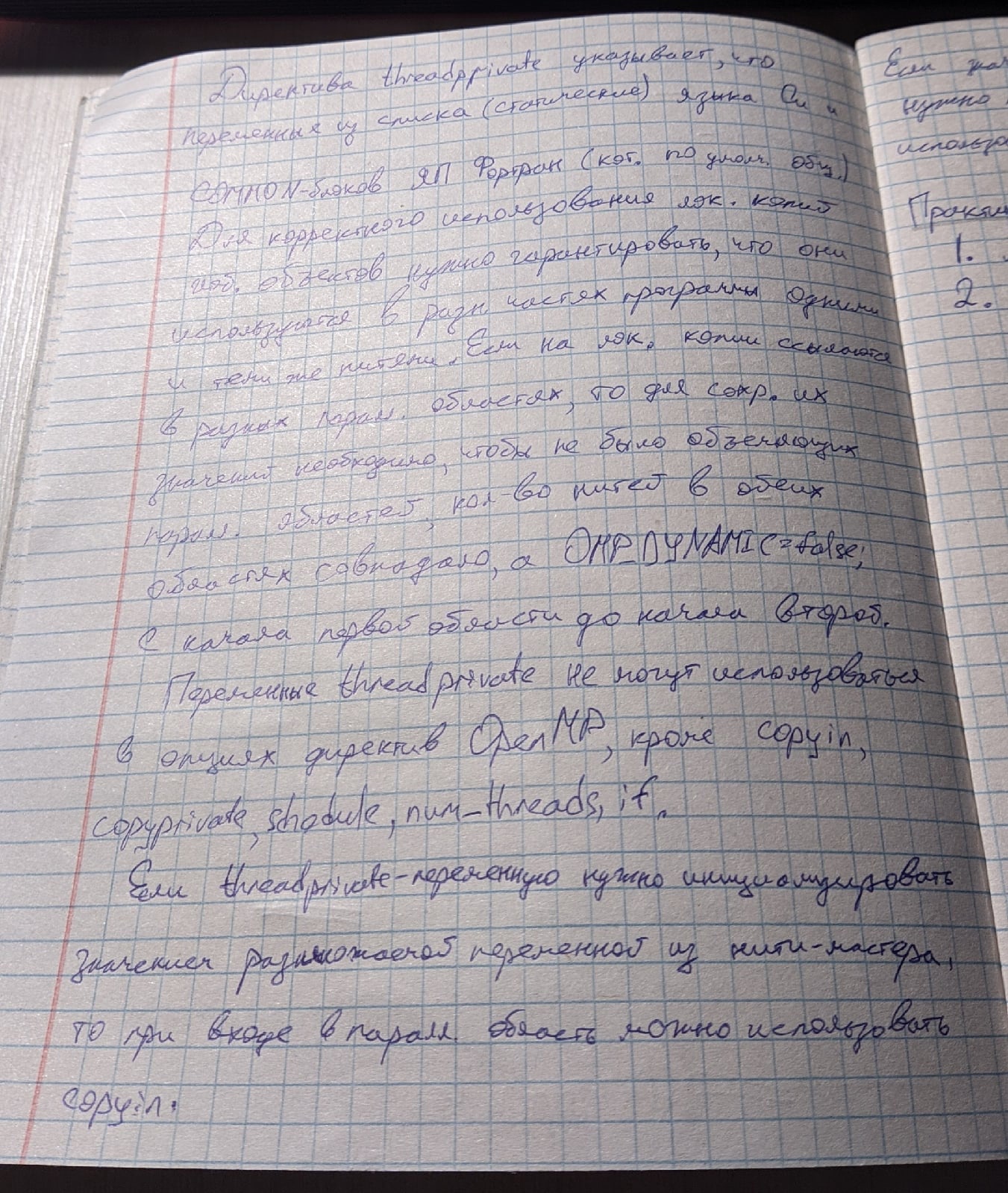
Цель работы: Провести работу с данными и выполнить ознакомление с моделью данных в OpenMP.

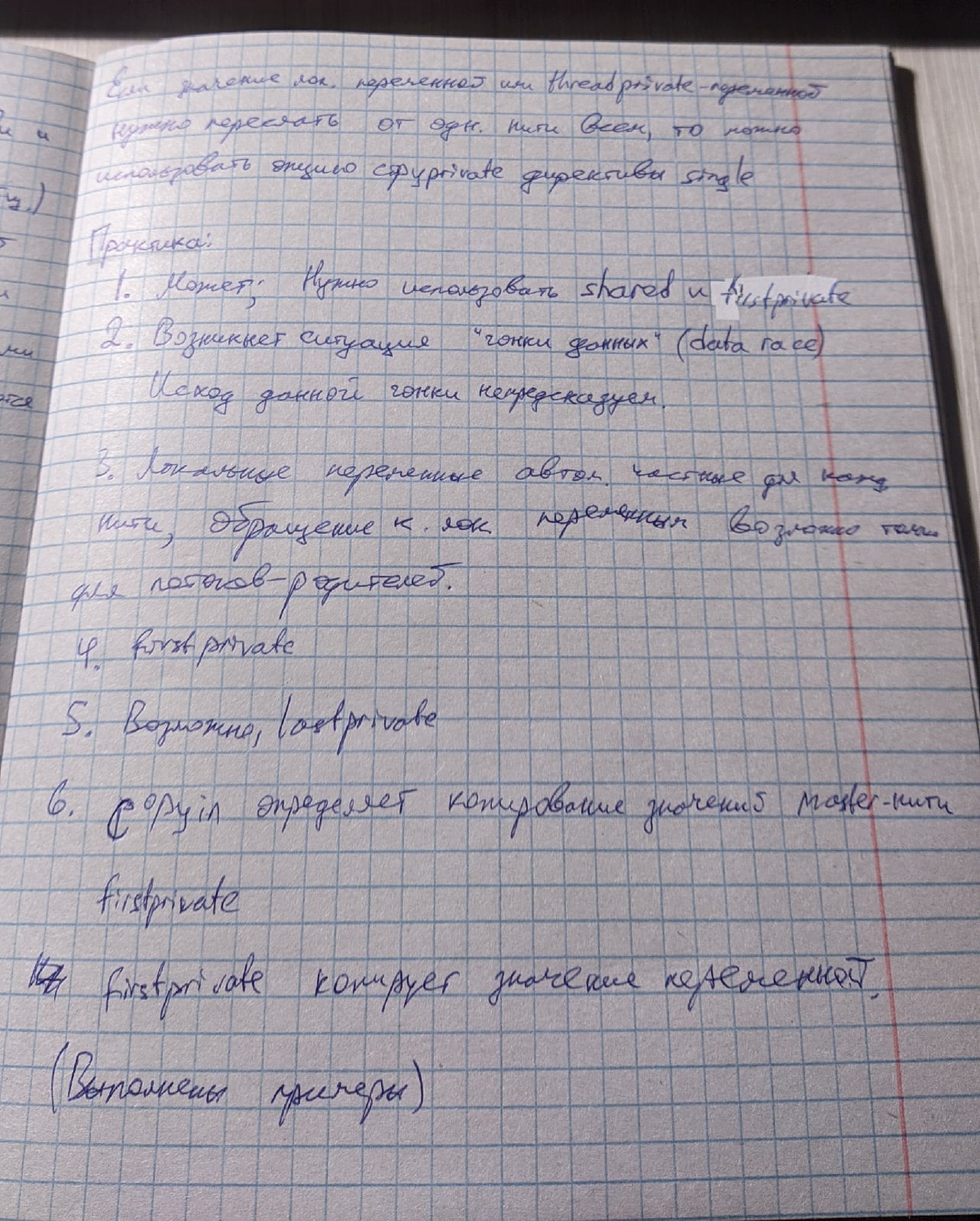
Теоретические сведения:

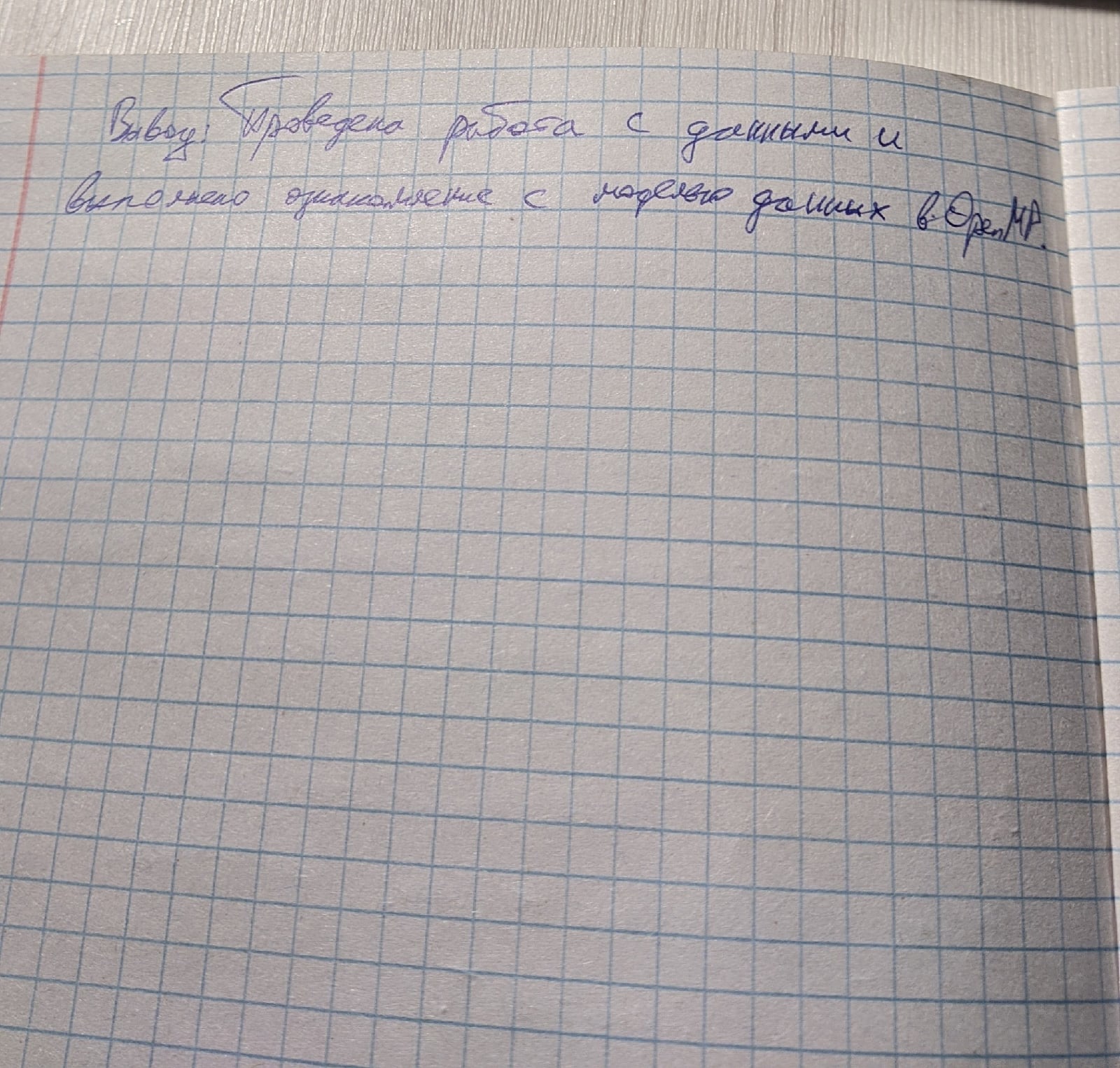












Задания (контрольные вопросы?):

1. Вопрос: «Может ли одна и та же переменная выступать в одной части программы как общая, а в другой части – как локальная?»

Ответ: Да, может. Просто изначально необходимо использование переменной с опцией shared, а в следующей области firstprivate.

2. Вопрос: «Что произойдёт, если несколько нитей одновременно обратятся к об-

щей переменной?»

Ответ: При многократном обращении к общей переменной в пределах параллельной области возможна ситуация «гонки данных» (англ. Data race). С этим обстоятельством результат выполнения программы будет непредсказуем.

3. Вопрос: «Может ли произойти конфликт, если несколько нитей одновременно

обратятся к одной и той же локальной переменной?»

Ответ: Локальные переменные автоматически являются частными для каждой нити. Обращение к локальной переменной возможно только для потока-родителя.

4. Вопрос: «Каким образом при входе в параллельную область разослать всем порождаемым нитям значение некоторой переменной?»

Ответ: Для этого можно использовать опцию firstprivate, где все значения локальных копий переменных инициализируются значениями некоторой избранной для этого переменной.

(#pragma omp parallel firstprivate(n) { //parallel code })

5. Вопрос: «Можно ли сохранить значения локальных копий общих переменных

после завершения параллельной области? Если да, то что необходимо

для их использования?»

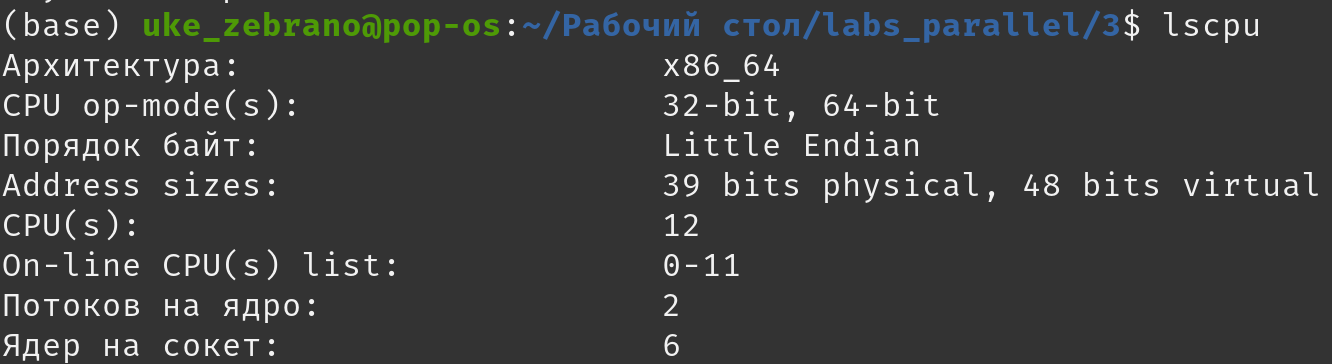
Ответ: Да, это возможно. Для этого используется опция lastprivate. Значение lastprivate-переменной после завершения блока параллельного исполнения определяется как ее значение на последней итерации цикла или в последней секции параллельной конструкции.

6. Вопрос: «В чём отличие опции copyin от опции firstprivate?»

Ответ: Опция copyin директивы parallel определяет порядок инициализации threadprivate-переменных: эти переменные инициализируются их значениями в master-нити в начале выполнения структурного блока.

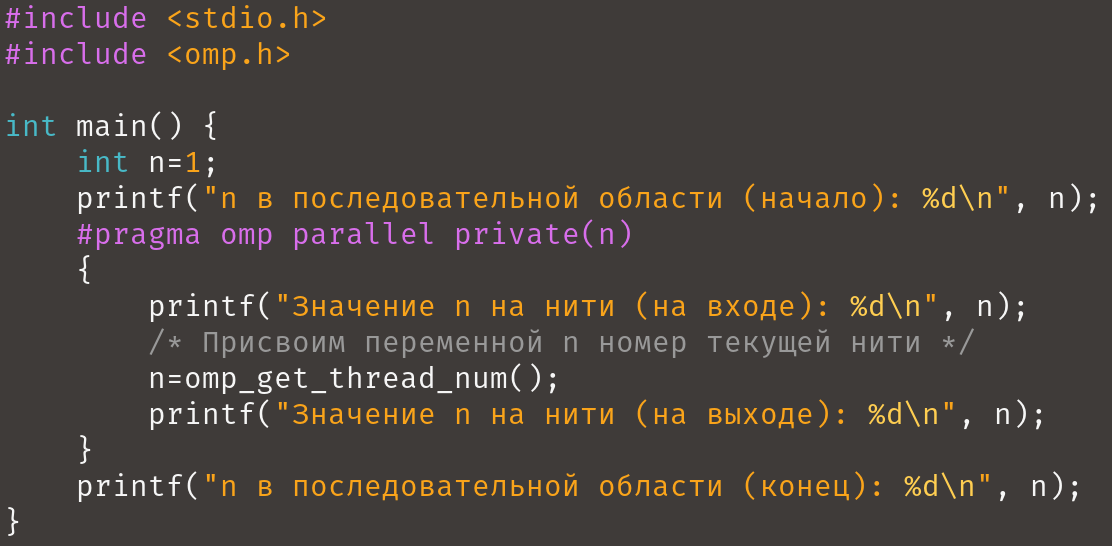
7. Вопрос (с прошлой лабораторной работы): «Можно ли с помощью OpenMP вывести количество физических процессоров»

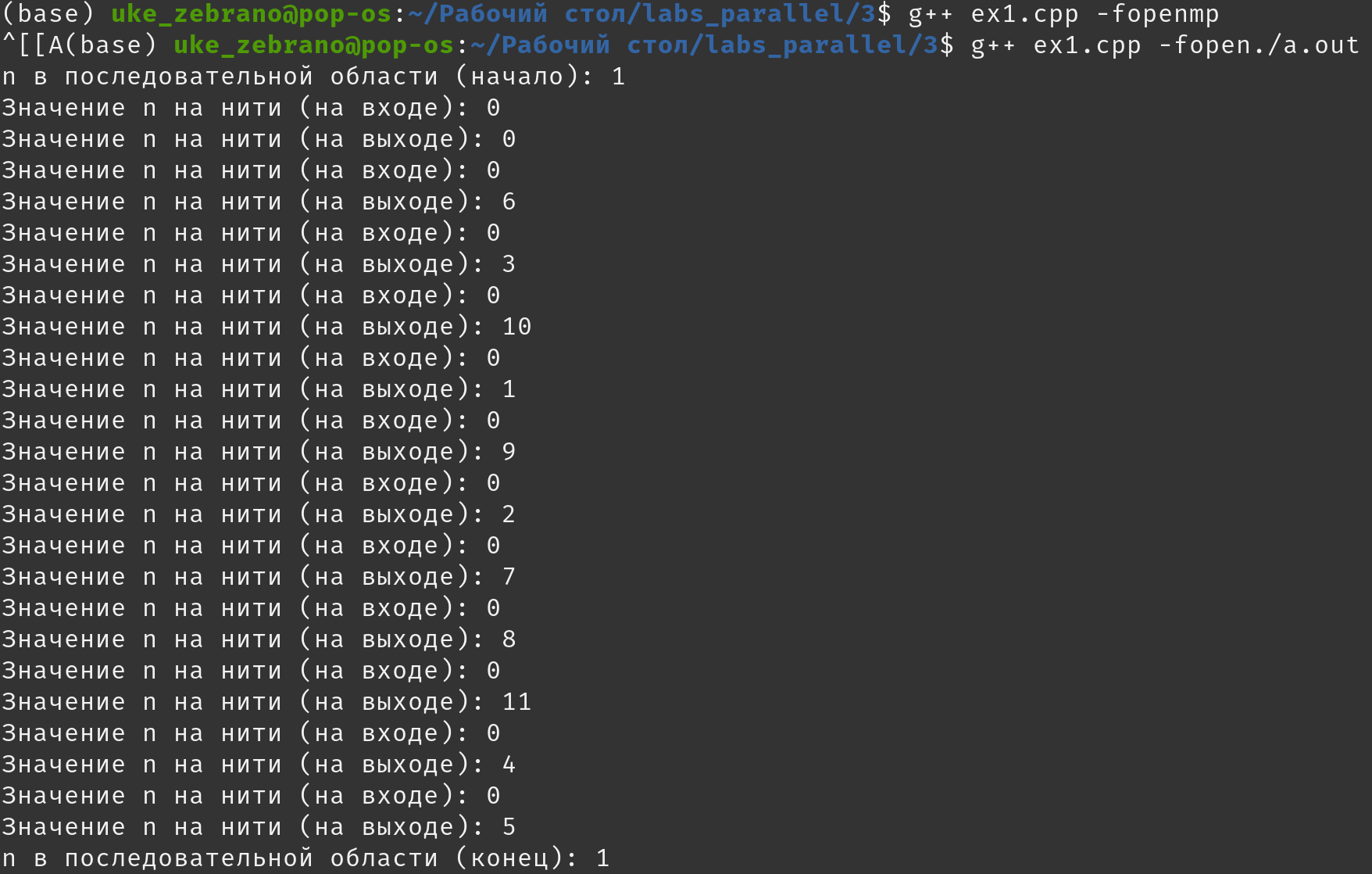
Ответ: Средствами OpenMP можно узнать только число логических процессоров (достаточно пустить выполнение параллельной программы с выводом в каждой нити). Узнать число именно физических процессоров можно с помощью средств linux (ubuntu), в числе ядер на сокет:



Задания (примеры из методички):

Пример 1 (в методичке обозначается как пример 12) Опция private.



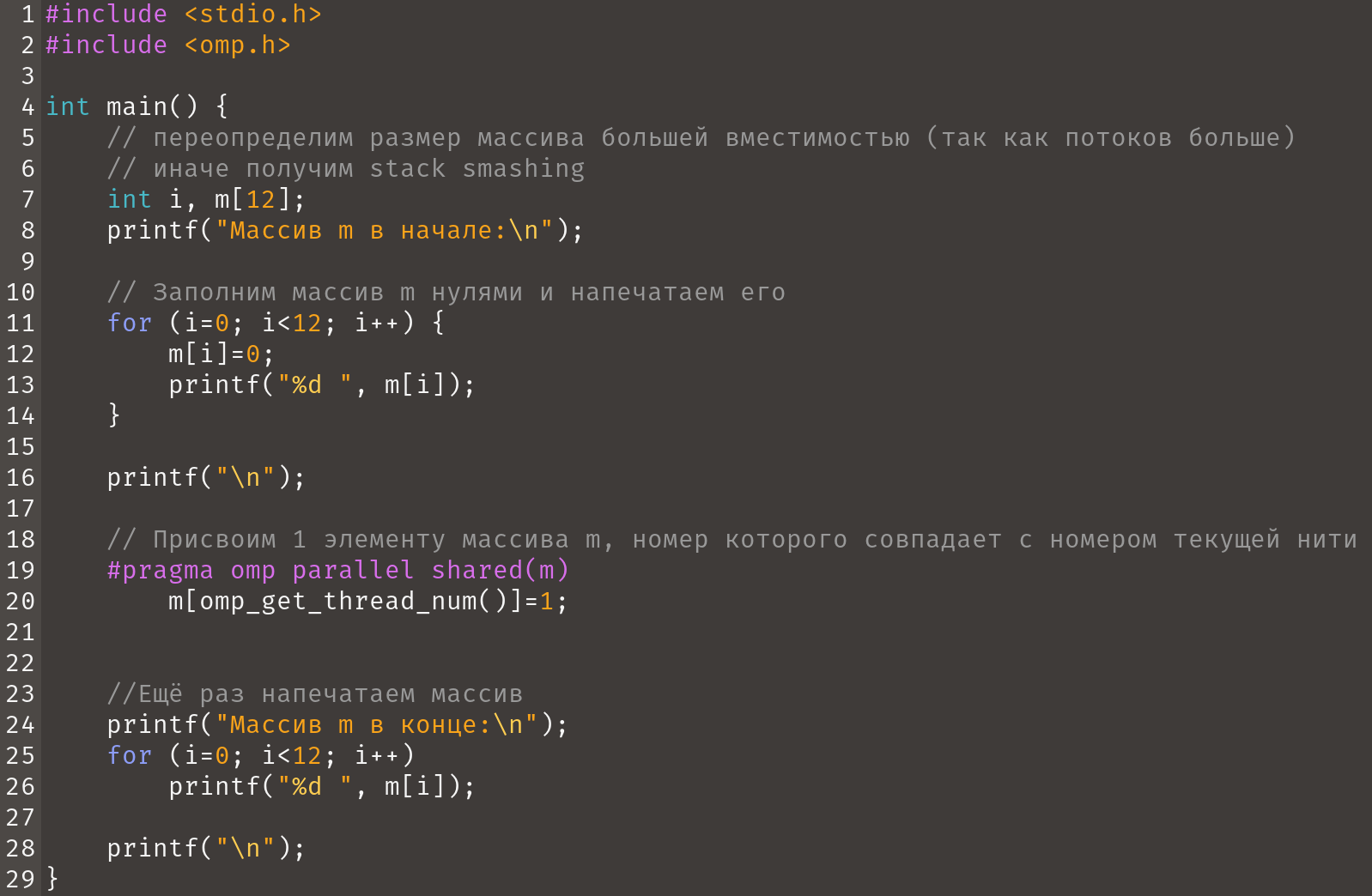


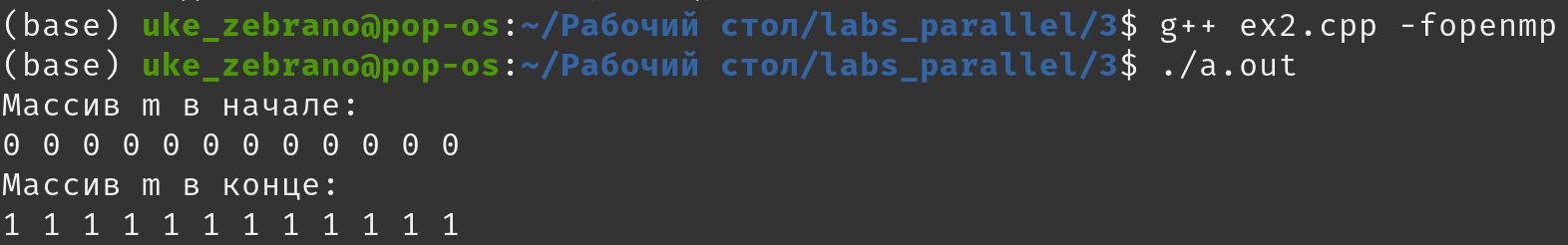
Как видно из консоли, при выполнении программы сначала выводится последовательно значение переменной n. Далее создаются локальные копии переменной n для каждого потока, они не инициализируются значением последовательной области (на входе).

Далее каждая локальная переменная приравнивается к номеру своей нити (на выходе).

После параллельной области выводится исходное значение переменной n.

Пример 2 (в методичке обозначается как пример 13) Опция shared

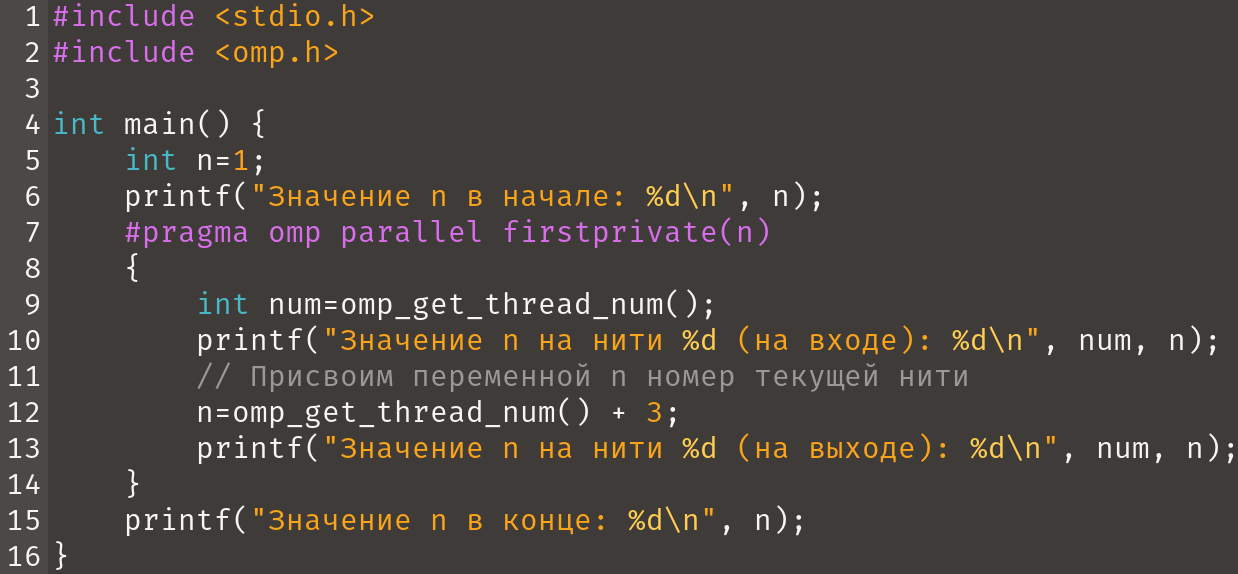


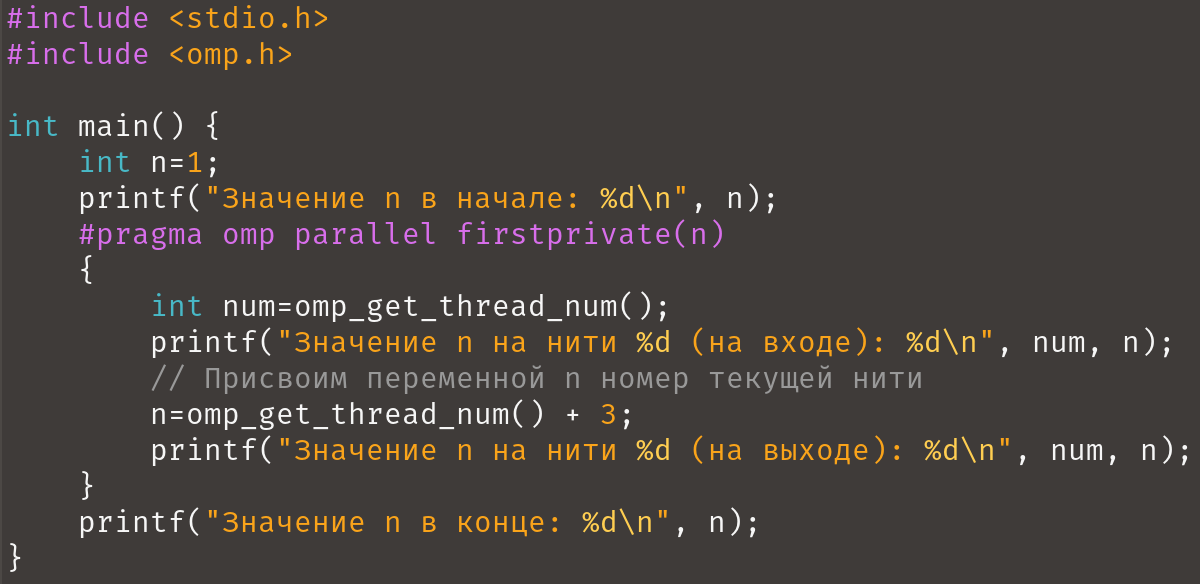
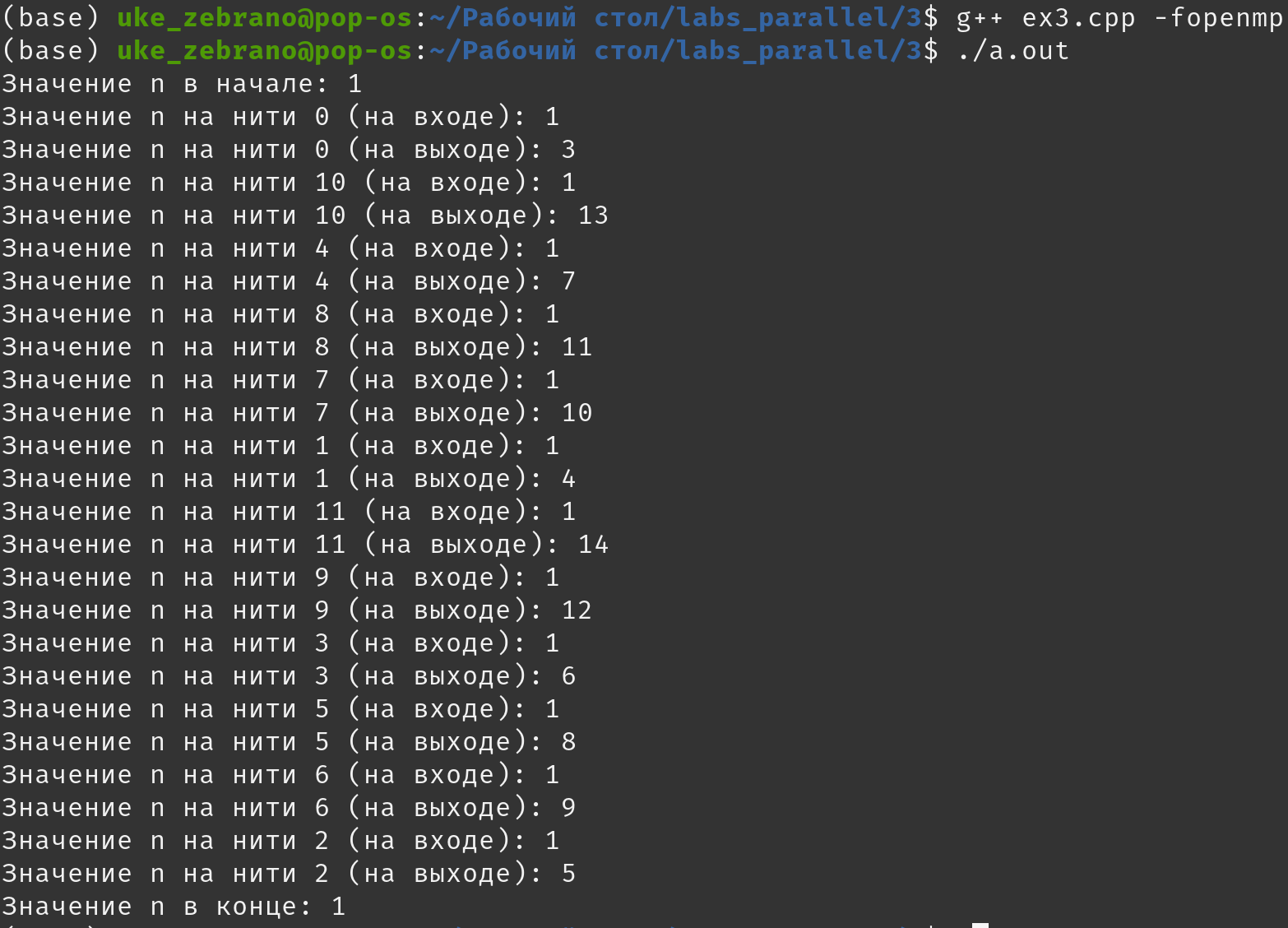
  
Здесь изначальный массив m, размер которого был доведен до количества потоков, последовательно выводится (инициализируясь нулями).

Далее используется вход в параллельную область с указанием общего массива m (присвоение значений «1» ячейкам).

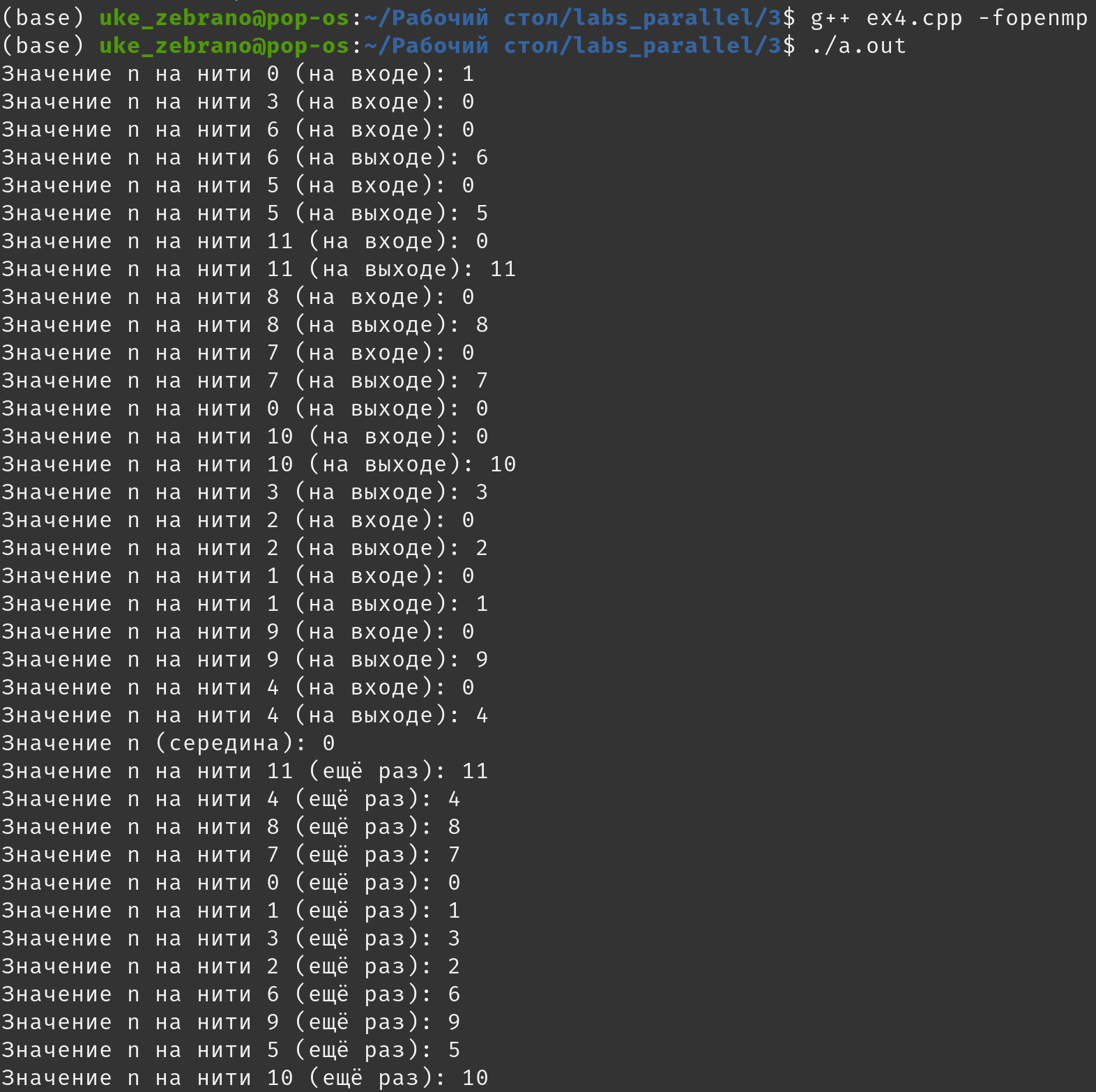
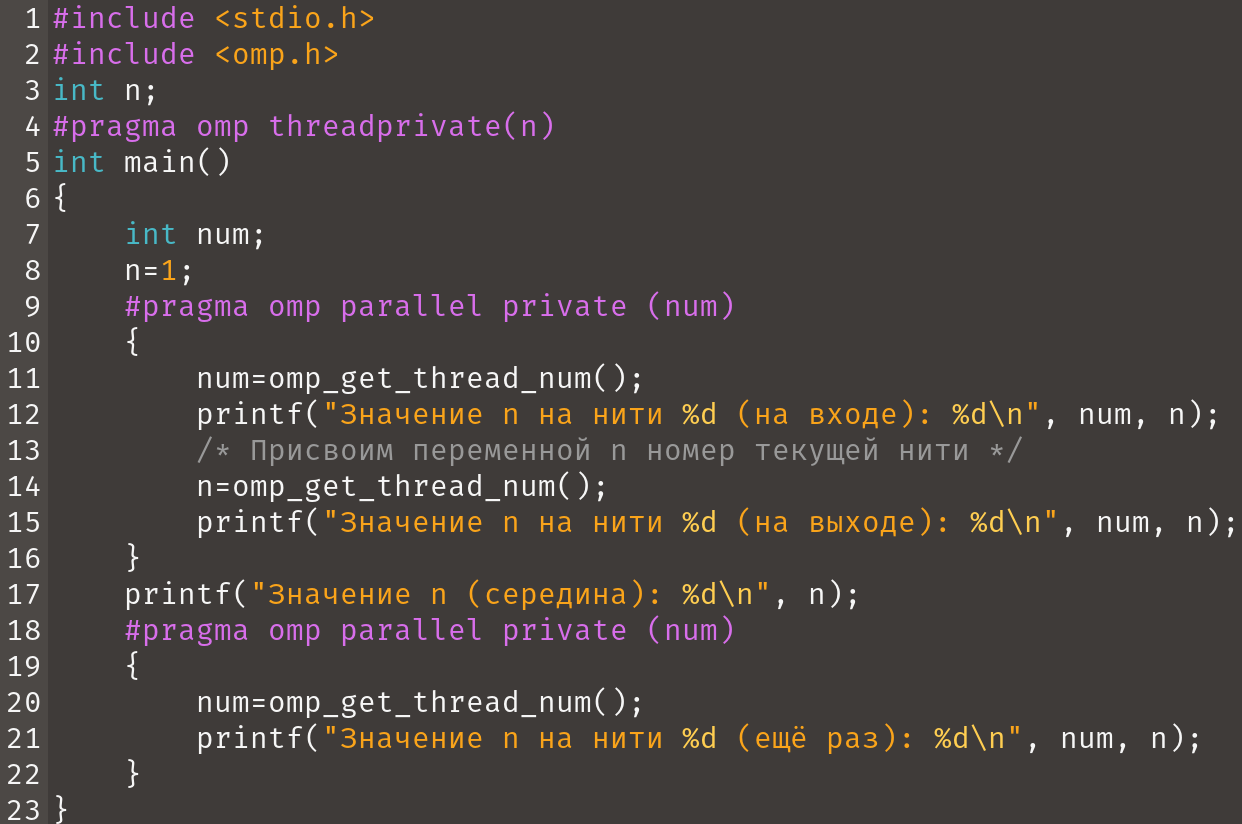
(значения массива меняются номерами нитей, которые в момент выполнения получали доступ к ячейкам). Гонки данных (data race) не происходит, ведь потоки обращались в ячейки с номером потока.

Пример 3 (в методичке обозначается как пример 14) Опция firstprivate



  
В данном примере видна суть работы опции firstprivate. Здесь переменной n присваивается значение «1». Далее в параллельной области создаются локальные копии этой переменной и инициализируются значением «1» переменной n из последовательной области (на входе). Далее локальные переменные берут в качестве значений номера потоков.   
После прохождения параллельной области значение n возвращается к изначальному, «1».

Пример 4 (в методичке обозначается как пример 15) Опция threadprivate

Y

;

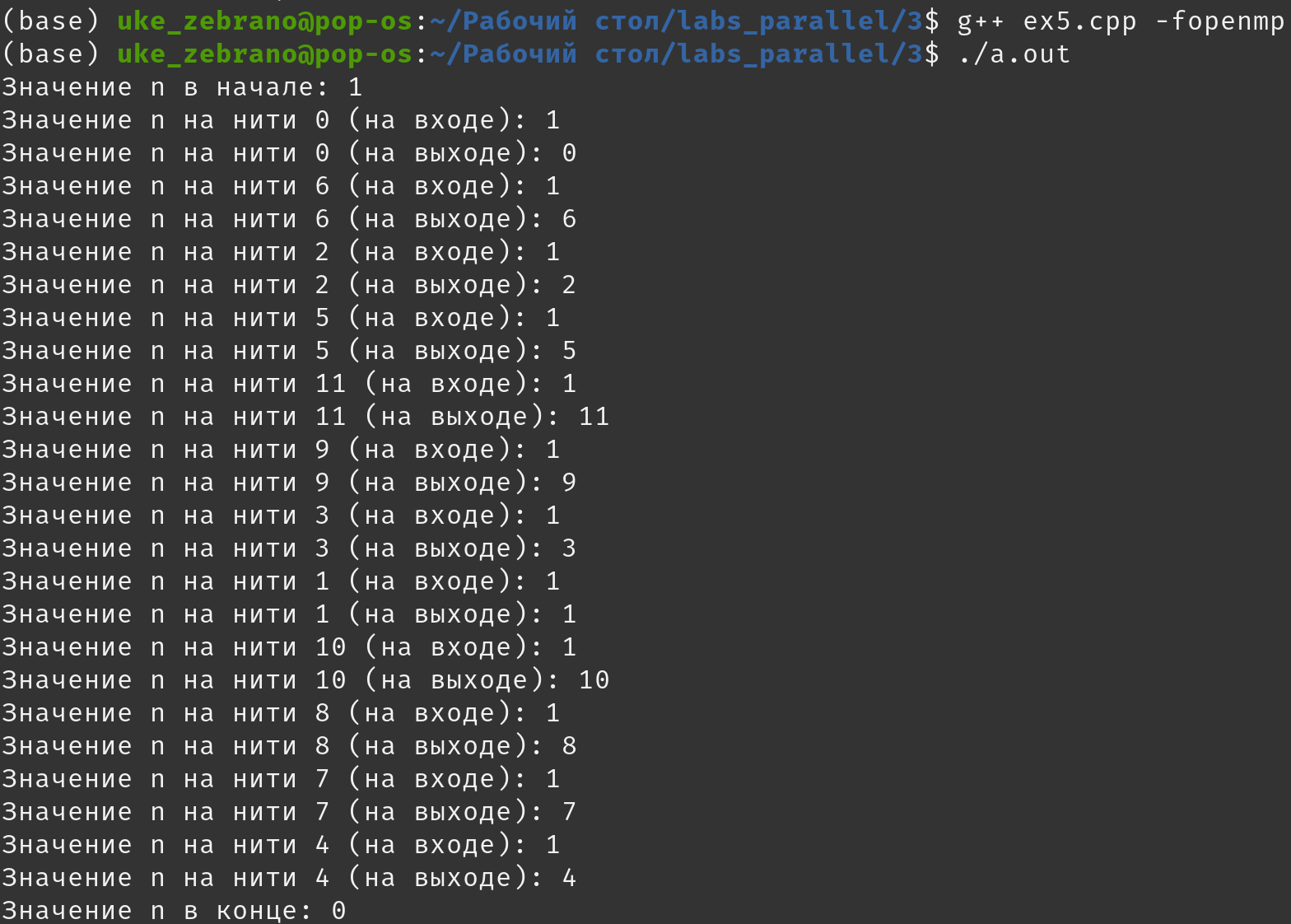
На данном примере видна работа опции threadprivate. Здесь переменная n — threadprivate.

Сначала выводятся локальные копии n (на входе), далее выводятся новые значения n (на выходе), номера нитей. Далее выводится значение n между параллельными областями (оно стало 0, такое значение было у нити master под номером 0).

В итоге, выводятся номера текущих нитей. Так как n — это threadprivate переменная, мы видим, что вывод выполнялся теми же потоками, что и в первый раз.

Пример 5 (в методичке обозначается как пример 16) Опция copyin

(copyin берет значение из мастер-нити и копирует его в остальные нити)  
В данном примере на входе значение из последовательной области, начальное. Значение на выходе становится номером текущего потока.

  
Вывод: Проведена работа с данными и выполнено ознакомление с моделью данных в OpenMP