# Информация о запросе

|  |  |
| --- | --- |
| Тело | tell me about parallel regions |
| Имя проекта | Новый тестовый проект |
| Использовать обогащение | Да |
| Дата и время создания запроса | 2024-09-21 20:40:51 |

# Обогащенный результат поиска

integer(kind = omp\_integer\_kind), intent(in) ::number\_of\_threadssubroutines that are executed before entering the new dynamic extent but after leaving the previous one.  
The OpenMP standard is silent about where in a parallel region an explicit synchronization point may be placed. But it does not prevent us from adding such points to our code. They can, e.g., be used for debugging and checking  
whether all threads have completed their work at some place of interest in the program.  
  
4.2 Synchronization Points  
Explicit synchronizations are needed if there is a need to wait until all the threads complete their execution before continuing further on.  
The following constructs allow us to achieve this:• omp barrier, an

# Результаты поиска (без обогащения)

|  |  |  |  |
| --- | --- | --- | --- |
| Модель | Дистанция | Источник | Результат |
| sber | 111.27842712402344 | F95\_OpenMPv1\_v2 (1).pdf\_40 | language. Since the author believes in the superiority of Fortran 95 over Fortran77 and inthe importance of a good programming methodology, the present document only presentsthose features of OpenMP which are in agreement with such a programming philosophy.This is the reason why it is advisable to |
| sber | 113.78644561767578 | F95\_OpenMPv1\_v2 (1).pdf\_427 | logical(kind = OMP\_logical\_kind) :: OMP\_get\_dynamic end function OMP\_get\_dynamic 4.1.9 OMPsetnested This subroutine enables or disables the nested parallelism. Its interface declaration is: subroutine OMP\_set\_nested(enable) logical(kind = OMP\_logical\_kind), intent(in) :: enable |
| sber | 119.4226303100586 | F95\_OpenMPv1\_v2 (1).pdf\_213 | all the other explicit or implied synchronizations in OpenMP. |
| LaBSE | 1.246077060699463 | F95\_OpenMPv1\_v2 (1).pdf\_64 | are also so called serial regions . When a thread executing a serial region encounters a parallel region, it creates a team of threads, and it becomes the master thread of the team. The master thread is a |
| LaBSE | 1.2684730291366577 | F95\_OpenMPv1\_v2 (1).pdf\_90 | 1. All work-sharing constructs must be placed inside dynamic extends of parallel regions |
| LaBSE | 1.294600248336792 | F95\_OpenMPv1\_v2 (1).pdf\_80 | while others apply to the dynamic extent. It is possible to nest parallel regions into parallel regions. For example, if a thread in a |
| rubert | 0.6495093107223511 | F95\_OpenMPv1\_v2 (1).pdf\_80 | while others apply to the dynamic extent. It is possible to nest parallel regions into parallel regions. For example, if a thread in a |
| rubert | 0.7424559593200684 | F95\_OpenMPv1\_v2 (1).pdf\_360 | When a do-loop is parallelized and its iterations distributed over the diﬀerent threads, the |
| rubert | 0.7515613436698914 | F95\_OpenMPv1\_v2 (1).pdf\_405 | This subroutine sets the number of threads to be used by subsequent parallel regions. Therefore, it can only be called from outside of a parallel region. Its interface declaration looks as follows: subroutine OMP\_set\_num\_threads(number\_of\_threads) |