Subject: Re: AsyncWork, IsFinished() may not be working properly
Posted by JeyCi on Sat, 18 Jul 2020 15:06:06 GMT

JeyCi wrote on Sat, 18 July 2020 16:44 while(!workers.IsEmpty())
  for(int i = 0; i < workers.GetCount(); i++) {
? is it duplication of loop & what for? why not use only one any row from these 2, not both?
sorry to hurry with my question - I understood -
1 cycle "for" from the beginning till the end of the container;
second cycle (while) - to repeat again & again for-cycle from the beginning till the end of the
container - until container IsEmpty... because after first for-loop some worker-threads can be still
not finished... really logically

JeyCi wrote on Sat, 18 July 2020 16:44
- it always seems to me that counter itself is always shared & should be locked somehow when
making i++...

but frankly speaking, if you insist that not, I can agree with you now... because this i (counter) is
still in one (primary) thread, from where it is taken to workers - Geniusly... thanks a lot
===
after your example - more clear is now my view at threads...