Fichier `histogramme.cpp`

```
////////////////////////////////////////////////////////////////
// Versions a paralleliser.
////////////////////////////////////////////////////////////////

int nb_occurrences_par( int val, int elems[], int nb )
{
  return parallel_reduce(
         blocked_range<size_t>(0, nb),
         0,
         [=]( blocked_range<size_t> r, int nb_occs ) {
           for( size_t k = r.begin(); k < r.end(); k++ ) {
             if ( elems[k] == val ) { nb_occs += 1; }
           }
           return nb_occs;
         },
         std::plus<int>()
   );
//
```

```
int* histogramme_par( int elems[], int nb, int valMax )
{
  // On alloue et on initialise l'histogramme.
  int *histo = (int*) malloc( (valMax+1) * sizeof(int) );

  parallel_for( blocked_range<size_t>(0, valMax+1),
       [=]( const blocked_range<size_t> r ) {
         for( size_t val = r.begin(); val < r.end(); val++ ) {
           histo[val] = nb_occurrences_par( val, elems, nb );
         }
       }
  );

  return histo;
}
```