vadd_par.c

```c
#include <stdio.h>
#include <omp.h>
#define N 10000000
#define TOL  0.0000001
//
//  This is a simple program to add two vectors
//  and verify the results.
//
//  History: Written by Tim Mattson, November 2017
//
int main()
{

    float a[N], b[N], c[N], res[N];
    int err=0;

    double init_time, compute_time, test_time;
    init_time    = -omp_get_wtime();

   // fill the arrays
   #pragma omp parallel for
   for (int i=0; i<N; i++){
      a[i] = (float)i;
      b[i] = 2.0*(float)i;
      c[i] = 0.0;
      res[i] = i + 2*i;
   }

   init_time    +=  omp_get_wtime();
   compute_time  = -omp_get_wtime();

   // add two vectors
   #pragma omp parallel for
   for (int i=0; i<N; i++){
      c[i] = a[i] + b[i];
   }

   compute_time +=  omp_get_wtime();
   test_time     = -omp_get_wtime();

   // test results
   #pragma omp parallel for reduction(+:err)
   for(int i=0;i<N;i++){
      float val = c[i] - res[i];
      val = val*val;
      if(val>TOL) err++;
   }

   test_time    +=  omp_get_wtime();

```

```c
    printf(" vectors added with %d errors\n",err);

    printf("Init time:    %.3fs\n", init_time);
    printf("Compute time: %.3fs\n", compute_time);
    printf("Test time:    %.3fs\n", test_time);
    printf("Total time:   %.3fs\n", init_time + compute_time + test_time);
    return 0;
}

```