Код :

package lb1;  
  
import mpi.MPI;  
  
public class OneLb {  
 public static void run(String[] args) {  
 int rank, size;  
 double startwtime, endwtime;  
 String name;  
  
 MPI.*Init*(args);  
 startwtime = MPI.*Wtime*();  
 rank = MPI.*COMM\_WORLD*.Rank();  
 size = MPI.*COMM\_WORLD*.Size();  
 name = MPI.*Get\_processor\_name*();  
// load();  
 endwtime = MPI.*Wtime*();  
 System.*out*.println(String.*format*("Hello world from process %d of %d at %s as %f second",rank, size, name, endwtime - startwtime));  
 MPI.*Finalize*();  
 }  
  
 private static void load(){  
 for(long i=0;i<20000000L;i++){  
 Math.*sqrt*(System.*currentTimeMillis*());  
 }  
 }  
}

Вывод программы:

15:01:01: Executing task 'run'...  
  
 > Task :compileJava  
 > Task :processResources NO-SOURCE  
 > Task :classes  
  
 > Task :run  
 MPJ Express (0.44) is started in the multicore configuration  
 Hello world from process 9 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 8 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 10 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 2 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 7 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 1 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 0 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 11 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 3 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 4 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 5 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
 Hello world from process 6 of 12 at MacBook-Pro-Konstantin.local as 0,000000 second  
  
 BUILD SUCCESSFUL in 4s  
 2 actionable tasks: 2 executed  
 15:01:05: Task execution finished 'run'.

Подключение библиотеки с помощью gradle:

plugins{  
 id 'application'  
}  
  
mainClassName = 'example.Main'  
  
def mpjHome = System.*getenv*('MPJ\_HOME')  
def mpjStarter = files("$mpjHome/lib/starter.jar")  
def mpjJar = files("$mpjHome/lib/mpj.jar")  
def mpjClassPath = sourceSets.main.runtimeClasspath - mpjJar  
dependencies {  
 compile mpjJar  
}  
  
run{  
 main = 'runtime.starter.MPJRun'  
 classpath = mpjStarter  
 args mainClassName, '-cp', mpjClassPath.asPath , '-np', 12  
 dependsOn classes  
}