Introduction to High-Performance Computing Exercise/2

Giorgio Amati

Corso di dottorato in Ingegneria Aeronautica e Spaziale 2024

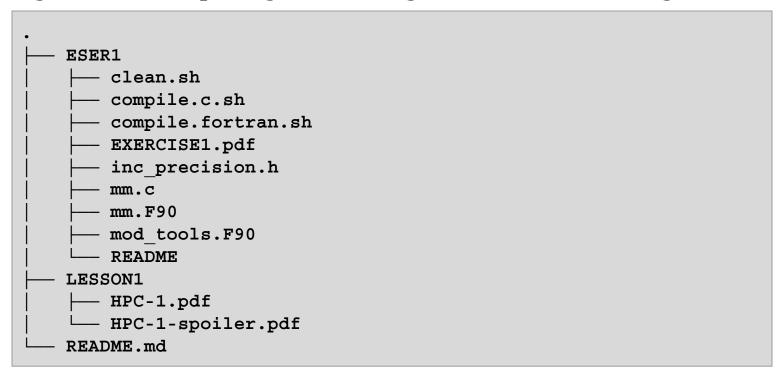
g.amati@cineca.it / g.amaticode@gmail.com

Agenda

- ✓ Exploit performance for matrix-matrix Multiplication
 - Fortran/C
- ✓ Always check the results
- ✓ Extract some Performance figure (in MFLOPs)
 - Change order of loops
 - Change size
 - With blocking
 - With unrolling
 - All together!!!!!
- ✓ Any available Compiler
- ✓ Medium optimization level (-02)

How to do

- Clone the repository
 - git clone https://github.com/gamati01/HPCLessons.git



Homework/1: Fill the table

Loop order	Size	Fortran	С
i,k,j	512*512		
j,k,i	512*512		
i,k,j	4096*4096		
j,k,i	4096*4096		

- ✓ Compiler used:
- ✓ Compiler option used:
- ✓ HW used:

Homework/2: Fill the table

✓ Unrolling external loop

#unrolling	Size	Fortran	С
2	512*512		
4	512*512		
2	4096*4096		
4	4096*4096		

- ✓ Compiler used:
- ✓ Compiler option used:
- ✓ HW used:

Homework/3: Fill the table

✓ Cache blocking: try a couple of blocking size...

#blocking	Size	Fortran	С
?	512*512		
?	512*512		
?	4096*4096		
?	4096*4096		

- ✓ Compiler used:
- ✓ Compiler option used:
- ✓ HW used:

Homework/4: Fill the table

✓ Put all together (blocking, unrolling,....)

#unrolling	#blocking	Size	Fortran	С
?	?	4096*4096		
?	?	8192*8192		

- ✓ Compiler used:
- ✓ Compiler option used:
- ✓ HW used:

Homework/1

Loop order	Size	Fortran (MFLOPs)
i,k,j	512*512	221
j,k,i	512*512	6871
i,k,j	4096*4096	380
j,k,i	4096*4096	4170

- ✓ Compiler used: gcc, rel. 11.4.0
- ✓ Compiler option used: -O2
- ✓ HW used: AMD Ryzen 5 5625U

Homework/2

Unrolling	Blocking	Fortran (Mflops)
-	-	5978
4*1*1	-	9626
-	64	10183
4*2*1	128	15623
4*4*1	128	17680

- ✓ Size=4096^2
- ✓ Compiler used: gcc, rel. 11.4.0
- ✓ Compiler option used: -O3
- ✓ HW used: AMD Ryzen 5 5625U

The code....

```
do jj = 1, n, step
do kk = 1, n, step
do ii = 1, n, step
   do j = jj, jj+step-1, 4
   do k = kk, kk+step-1, 4
   do i = ii, ii+step-1
      temp0 = a(i,k+0)
      temp1 = a(i,k+1)
      temp2 = a(i,k+2)
      temp3 = a(i,k+3)
      c(i,j+0) = c(i,j+0) + temp0*b(k+0,j+0) + temp1*b(k+1,j+0) + temp2*b(k+2,j+0) + temp3*b(k+3,j+0)
      c(i,j+1) = c(i,j+1) + temp0*b(k+0,j+1) + temp1*b(k+1,j+1) + temp2*b(k+2,j+1) + temp3*b(k+3,j+1)
      c(i,j+2) = c(i,j+2) + temp0*b(k+0,j+2) + temp1*b(k+1,j+2) + temp2*b(k+2,j+2) + temp3*b(k+3,j+2)
      c(i,j+3) = c(i,j+3) + temp0*b(k+0,j+3) + temp1*b(k+1,j+3) + temp2*b(k+2,j+3) + temp3*b(k+3,j+3)
   enddo
   enddo
   enddo
enddo
enddo
          enddo
```