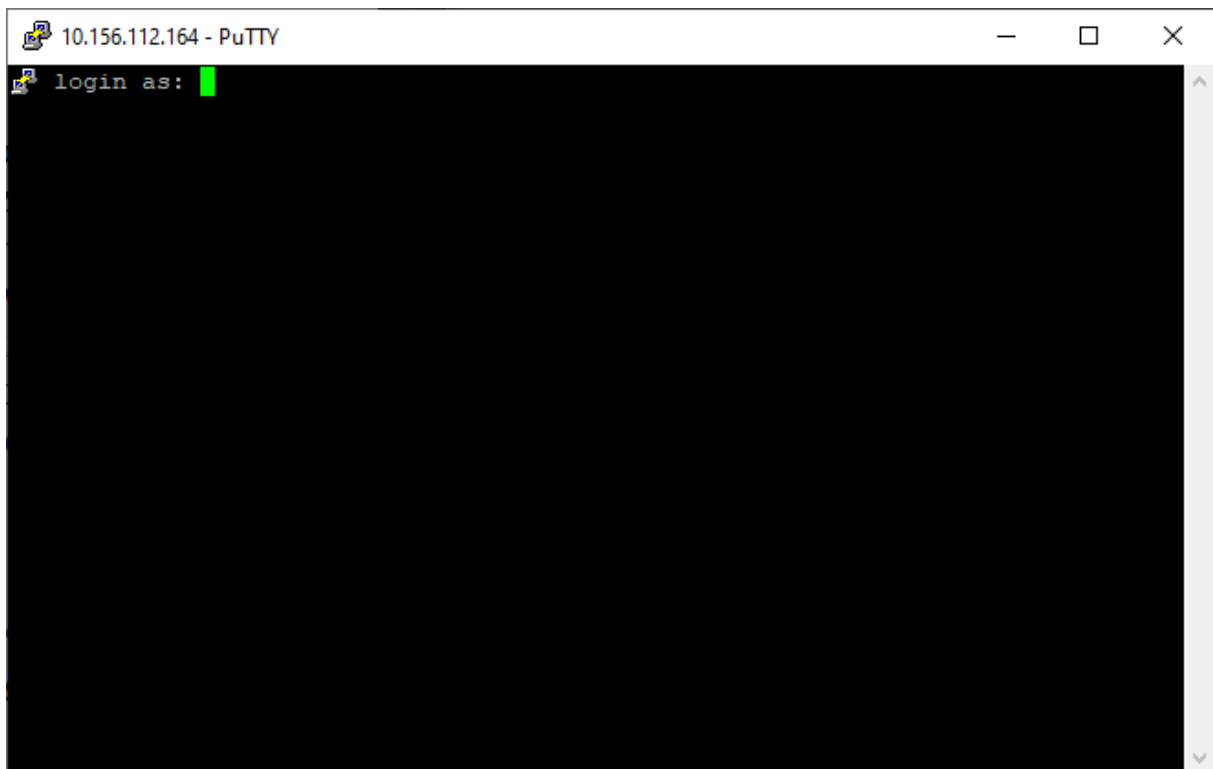


Wstęp

Celem zajęć jest wprowadzenie do obsługi zaawansowanych pakietów obliczeń naukowo-technicznych na przykładzie projektu "ModFEM" (ModFEM pozwala na obliczenia metodą elementów skończonych na klastrach z akceleratorami w postaci kart graficznych, podczas zajęć wykorzystywany będzie pojedynczy serwer "HONORATA"). W trakcie zajęć przypomniane zostaną zagadnienia związane z obsługą Linuksa z linii komend oraz kompilacja kodu źródłowego z wykorzystaniem skryptów narzędzia 'cmake'. Na zakończenie uruchomiony zostanie program 'ModFEM'

Wykonanie:

Pracę rozpoczynamy od podłączenia się do serwera z użyciem VPN oraz PuTTY



Po zalogowaniu z użyciem otrzymanego loginu i hasła, tworzona jest struktura plików w celu zapoznania się z podstawowymi komendami systemu Linux (mkdir, touch, cp, mv, rm)

```
litewka_jakub@honorata:~/lab_01/test1$ tree
.
├── plik3.c
├── test2
│   ├── plik1.c
│   └── plik2.c
└── test3

2 directories, 3 files
```

Wynikiem jest przedstawiona struktura plików i katalogów.

Kolejnym krokiem jest skopiowanie z katalogu serwerowego na konto pliku .tgz, a następnie wypakowanie go za pomocą tar.

```
litewka_jakub@honorata:~/ModFEM_v1/src$ ls
amg_mkb          mmd_prism          pdd_ns_mixed
apd_dg_prism     mmd_prism_2D       pdd_ns_supg
apd_std_lin      mmd_remesh         pdd_ns_supg_heat
apd_std_quad     mmd_t4_prism       pdd_plast_flow_mixed
appd_dg_prism    mmpd_adapter       pdd_plast_flow_supg
appd_std_lin     mmpd_prism         platform_files
ARCHIVE          modfem_doxy_config sid_lapack
cmake            ModFEM_OS.cmake    sid_mkb
CMakeLists.txt   mod_fem_viewer     src.kdev4
CTestConfig.cmake OpenCL_kernels      tmd_opengl
ddd_parmetis     OWNERSHIP_info     utd_system
include          pcd_mpi            utd_util
lsd_mkb          pcd_mpi_safe       utilities
Makefile_explicit pdd_conv_diff
materials_database.dat pdd_heat
```

Utworzenie pliku `src/cmake/Platforms/imie_nazwisko.cmake`
poprzez przekopiowanie pliku `REFERENCE_PLATFORM.cmake`

Przygotowany plik należy zedytować w celu ustawienia odpowiedniej konfiguracji programu ModFEM

```
#*****  
#                               #  
#               ModFEM          #  
#       Jakub Litewka grupa 3 lab01      #  
#####  
message("")  
message("**** Entering REFERENCE platform file with MKL and debug options")  
message("")  
  
#set(CMAKE_BUILD_TYPE Debug)  
set(CMAKE_BUILD_TYPE Release) # in case something goes wrong  
  
# ----- User compiler flags ----- #  
if(CMAKE_BUILD_TYPE STREQUAL "Release") #Flags for release mode  
    set(USER_CMAKE_C_FLAGS "") #C compiler flags  
    set(USER_CMAKE_CXX_FLAGS "") #C++ compiler flags  
    set(USER_CMAKE_EXE_LINKER_FLAGS "-m64") #Linker flags  
elseif(CMAKE_BUILD_TYPE STREQUAL "Debug") #Flags for debug mode  
  
    # Debug flag info  
    # Warning flags: https://gcc.gnu.org/onlinedocs/gcc/Warning-Options.html  
    # Debug flags: https://gcc.gnu.org/onlinedocs/gcc/Instrumentation-Options.html  
    # Program instrumentation options: https://gcc.gnu.org/onlinedocs/gcc/Instrumentation-Options.html  
  
    set(USER_CMAKE_C_FLAGS "-O0 -g") #C compiler flags  
    set(USER_CMAKE_CXX_FLAGS "-O0 -g") #C++ compiler flags  
    set(USER_CMAKE_EXE_LINKER_FLAGS "-O0 -g -m64") #Linker flags
```

```
set(MKL_LIBRARY_DIRS /opt/intel/oneapi/mkl/latest/lib/intel64 /opt/intel/  
# MKL library names (no extension) -  
# - consult: http://software.intel.com/en-us/articles/intel-mkl-link-line  
# Select 32-bit or 64-bit system libraries by comment unused  
  
# 32-bit system  
set(MKL_INTEL_LIB_NAME mkl_intel) #never use 'lib' prefix  
set(MKL_LAPACK_LIB_NAME mkl_lapack95) #usually not needed #never use 'l'  
set(MKL_SOLVER_LIB_NAME mkl_solver) #never use 'lib' prefix  
set(MKL_IOMP5_LIB_NAME iomp5 libiomp5md) #never use 'lib' prefix  
set(MKL_THREAD_LIB_NAME mkl_intel_thread) #never use 'lib' prefix  
set(MKL_CORE_LIB_NAME mkl_core) #never use 'lib' prefix  
  
# 64-bit system  
set(MKL_INTEL_LIB_NAME mkl_intel_lp64) #never use 'lib' prefix  
set(MKL_LAPACK_LIB_NAME mkl_lapack95_lp64) #usually not needed #never us  
set(MKL_SOLVER_LIB_NAME mkl_solver_lp64) #never use 'lib' prefix  
set(MKL_IOMP5_LIB_NAME iomp5 libiomp5md) #never use 'lib' prefix  
set(MKL_THREAD_LIB_NAME mkl_intel_thread) #never use 'lib' prefix  
set(MKL_CORE_LIB_NAME mkl_core) #never use 'lib' prefix  
  
# ----- BLAS and LAPACK configuration ----- #  
# Blas and Lapack paths and names  
# (only relevant if MODFEM_USE_MKL set to FALSE)  
# (when MKL in use blas/lapack from MKL will be used)  
#  
# Paths to look for Blas,Lapack libraries  
# (usually can be left empty)  
set(LAPACK_DIRS)  
set(BLAS_DIRS)  
  
# Blas,Lapack library names (no extension)  
set(BLAS_LIB_NAME blas) #never use 'lib' prefix  
set(LAPACK_LIB_NAME lapack) #never use 'lib' prefix  
  
# ----- LIBCONFIG configuration ----- #  
# Libconfig paths and names  
# (only important when building targets that use Libconfig)  
set(LIBCONFIG_INCLUDE_DIRS /usr/include)  
set(LIBCONFIG_LIBRARY_DIRS /usr/lib64)  
set(LIBCONFIG_LIB_NAME config) #never use 'lib' prefix
```

Utworzenie katalogu kompilacji `bin_cmake/jakub_litewka_nompi_none_gcc_g++`
 Następnie w tym katalogu należy odpalić komendę `cmake ../../src/`

```

=> Config file in use: jakub_litewka.cmake
=> Iterative solver module in use: sil_mkb
=> Direct solver via MKB interface: PARDISO
=> Direct solver module in use: sil_lapack [deprecated]
=> BLAS/LAPACK from: MKL
=> Compilation:
  - CMAKE_BUILD_TYPE: Release
  - MODFEM_USE_STATIC: TRUE
  - Fortran:
    - Compiler:
    - Flags:
  - C:
    - Compiler: /usr/bin/gcc
    - Flags: -Wno-unused-variable -Wno-unused-parameter -Wno-unused-but-
set-variable -Wno-comment -pipe -ldl -O2 -DNDEBUG
  - C++:
    - Compiler: /usr/bin/g++
    - Flags: -Wno-unused-variable -Wno-unused-parameter -Wno-unused-but-
set-variable -Wno-comment -pipe -ldl -O2 -DNDEBUG
  - Link:
    - Flags for exe: -m64
    - Flags added manually:
  - List of definitions:
    - Definitions: MIXED_DOF_BY_DOF;RENUMBERING;HAS_BOOST
  - MODFEM_MPI: nomp
  - MODFEM_ACCEL: none
  - ModFEM MuPARSER is disabled
  - ModFEM TIME TEST is disabled
  - Matrix print is disabled
  - Matrix storage in binary CRS "modfem_crs.bin" file is disabled
  - MIXED approximation storage type is: MIXED_DOF_BY_DOF
  - RENUMBERING is enabled
=> List of created auxiliary tools:
-- CREATE_MOD_FEM_VIEWER flag is FALSE
=> List of created executable files:
-- CREATE_MOD_FEM_NS_SUPG_PRISM_STD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_HYBRID_STD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_REMESH_STD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_PRISM2D_STD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_PRISM_STD_QUAD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_HYBRID_STD_QUAD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_REMESH_STD_QUAD flag is FALSE
-- CREATE_MOD_FEM_NS_SUPG_PRISM2D_STD_QUAD flag is FALSE
-- CREATE_MOD_FEM_HEAT_PRISM_STD flag is TRUE
-- CREATE_MOD_FEM_HEAT_PRISM_STD_QUAD flag is TRUE
-- CREATE_MOD_FEM_HEAT_PRISM2D_STD flag is TRUE

```

W efekcie powstają pliki w obecnym katalogu. Po wykonaniu komendy make -j4 zostaną utworzone również pliki

```

apd_dg_prism      Makefile      pdd_heat
apd_std_lin       mmd_prism      pdd_ns_supg
apd_std_quad      mmd_prism_2D   pdd_ns_supg_heat
CMakeCache.txt    mmd_remesh      sid_lapack
CMakeFiles        mmd_t4_prism    sid_mkb
cmake_install.cmake MOD_FEM_heat_prism2d_std  Testing
CTestTestfile.cmake MOD_FEM_heat_prism_std   utd_system
DartConfiguration.tcl MOD_FEM_heat_prism_std_quad utd_util
lsd_mkb           MOD_FEM_ns_supg_heat_prism2d_std

```

Należy przekopiować wszystkie pliki za pomocą wget do katalogu lab_01

wget http://ww1.metal.agh.edu.pl/~banas/MMNT/stationary_linear_problem/problem_heat.dat

File Name	Date	Time	Size
bc_heat.dat	28-Dec-2021	16:27	1k
mesh_prism.dat	28-Dec-2021	16:27	1k
problem_heat.dat	14-May-2021	18:27	7k

Uruchamianie program ModFEM za pomocą polecenie

```
~/ModFEM/bin_cmake/imie_nazwisko_nompi_none_gcc_g++/MOD_FEM_heat_prism_std .
```

```
fter reading initial mesh data.

esh entities (number of active, maximal index):
lements: nrel 2, nmel 2
aces:    nrfa 9, nmfa 9
dges:    nred 14, nmed 14
odes:    nrno 8, nmno 8

aximal generation level set to 9, maximal generation difference set to 1

nitializing heat field with 0

pproximation type: LINEAR
R DOF ENTS/VERTs: 8 ; nreq = 1 ; nr_sol = 3
fter create CONSTRAINTS DATA:
R DOF ENTS/VERTs: 8 ; nreq = 1 ; nr_sol = 3

hoose a command from the menu:
  t - solve the problem (time integration)
  s - solve single heat problem
  e - compute error for test problems with known exact solution
  z - compute error (ZZ aproximation)
  p - postprocessing
  g - launch graphics module
  m - perform uniform mesh refinement
  a - perform automatic mesh adaptation
  r - print profile
  d - dump out data
  v - write ParaView graphics data
  c - change control data
  h - perform random test of h-adaptivity
  q - exit the program
```

Po kliknięciu kolejno opcji s, z, otrzymuje się wynik

```
z
Patch creation - time 0.000014
Derivative recovery - time 0.000148

Zienkiewicz-Zhu error estimator = 1.207858240564
```

A po naciśnięciu q, wychodzi się z programu Zadanie – konfiguracja plików do programu ModFEM	OCENA własna w % (0-100)	OCENA prowadzącego w % (0-100)
Zad. 1 – tworzenie plików i katalogów w systemie Linux	100	
Zad. 2/3/4 – praca zdalna i przesyłanie plików	100	
Zad. 5 – konfiguracja plików ModFEM	100	
Zad. 6/7/8 – kompilacja i uruchomienie	100	
łącznie (400)	400	
Ocena końcowa:		