

Tworzenie pliku .jk z pliku NASTRAN (.nas) za pomocą programu nas2D2jk

Znajdując się w lokalizacji pliku nas2D2jk.exe włączamy go z plikiem .nas jako argumentem
`./nas2D2jk.exe ../../lab_03/litewka_jakub_A2_1_1` (pomijamy rozszerzenie)

```

x1 1.500000, y1 1.000000, x2 1.390760, y2 0.609240, x3 1.697973, y3 0.684846
vector product for triangle 41: 0.111787
edge 0: v1 0, v2 10, bc 1, neig: 1 -111173
edge 1: v1 10, v2 4, bc 1, neig: 24 -111173
edge 2: v1 4, v2 11, bc 2, neig: 25 -111173
edge 3: v1 11, v2 3, bc 2, neig: 36 -111173
edge 4: v1 3, v2 12, bc 3, neig: 37 -111173
edge 5: v1 12, v2 1, bc 3, neig: 19 -111173
edge 6: v1 1, v2 13, bc 4, neig: 15 -111173
edge 7: v1 13, v2 14, bc 4, neig: 32 -111173
edge 8: v1 14, v2 2, bc 4, neig: 28 -111173
edge 9: v1 2, v2 15, bc 5, neig: 18 -111173
edge 10: v1 15, v2 6, bc 5, neig: 7 -111173
edge 11: v1 6, v2 16, bc 6, neig: 0 -111173
edge 12: v1 16, v2 17, bc 6, neig: 33 -111173
edge 13: v1 17, v2 8, bc 6, neig: 16 -111173
edge 14: v1 8, v2 18, bc 7, neig: 22 -111173
edge 15: v1 18, v2 9, bc 7, neig: 38 -111173
edge 16: v1 9, v2 19, bc 8, neig: 39 -111173
edge 17: v1 19, v2 5, bc 8, neig: 27 -111173
edge 18: v1 5, v2 20, bc 9, neig: 26 -111173
edge 19: v1 20, v2 7, bc 9, neig: 3 -111173
edge 20: v1 7, v2 21, bc 10, neig: 20 -111173
edge 21: v1 21, v2 0, bc 10, neig: 4 -111173
maxvert 10000, maxedge 40000, maxface 50000, maxelem 20000
nr_vert 33, lower z 0.000000, upper z 0.100000, nr_layer 1, lower BC flag 0, upper BC flag 0

```

Plik litewka_jakub_A2_1_1.jk

```

10000 40000 50000 20000
33 0 0.1 2 21 22
0 0
-1 0
0 -1
-1 1
0 1
1 1
1 -1
1 0
2 0
2 1
0 0.5
-0.5 1
-1 0.5
-0.66666 -0.33333
-0.33333 -0.66666
0.5 -1

```

Zmiana ostatnich 3 wartości drugiej linii, na 2 21 22

2 – liczba warstw

21 – numer warunku brzegowego ściany dolnej

22 – numer warunku brzegowego ściany górnej

Następnie należy skonfigurować plik **bc_heat.dat** zawierający informacje o warunkach brzegowych.

```
{
  bcnnum:1;
  normal_heat_flux:{flux=0.0;}; //izolacja
},
{
  bcnnum:2;
  radconv:{t_out=300.0;alfa=100.0;eps=0.0;}; //chlodzenie
},
{
  bcnnum:3;
  normal_heat_flux:{flux=0.0;}; //izolacja
},
{
  bcnnum:4;
  normal_heat_flux:{flux=0.0;}; //izolacja
},
{
  bcnnum:5; // bottom side
  isothermal:{temp=450;}; //grzanie
},
{
  bcnnum:6;
  normal_heat_flux:{flux=0.0;}; //izolacja
},
{
  bcnnum:7;
  normal_heat_flux:{flux=0.0;}; //izolacja
},
}
```

Do każdego warunku brzegowego należy napisać strukturę:

```
{
  bcnnum:<numer warunku brzegowego>;
  <opcja oznaczająca grzanie/chłodzenie/izolator>;
}
```

Grzanie:

```
isothermal:{temp=<temperature>;};
```

isothermal – zadanie określonej temperatury (warunek Dirichleta)

Chłodzenie

```
radconv:{T_out=<temperature>;alfa=100.0;eps=0.0;};
```

radconv – zadanie warunku radiacji (warunek Robina)

Izolacja

```
normal_heat_flux:{flux=0.0;};
```

normal_heat_flux – określenie strumienia ciepła (warunek Neumanna)

Zmiana wartości w pliku **problem_heat.dat**

```
/* CONTROL PARAMETERS - GENERIC: FIELDS REQUIRED FOR ALL PROBLEMS */

name = "nazwa_problemu";

/* mesh, field, materials & bc input filenames */

mesh_type = "j"; // "j" - jk mesh, "p" - standard prismatic mesh format
// work_dir: specified by the program input argument
// interactive_input: initialized by the main procedure
// interactive_output: initialized by the main procedure

mesh_file_in = "litewka_jakub_A2_1_1.jk"; // mesh_filename (TUTAJ WSKAZUJEMY UTWORZONY PLIK SIATKI )
field_file_in = "z"; // field filename or "z" for zero field
// or "i" for initialize with provided function
```

mesh_type ustawiamy na „j”, mesh_file_in na plik .jk

Uruchamianie programu MOD_FEM_heat_prism_std

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

```
Maximal generation level set to 9, maximal generation difference set to 1

Initializing heat field with 0

Approximation type: LINEAR
NR DOF ENTS/VERTs: 99 ; nreq = 1 ; nr_sol = 3
After create CONSTRAINTS DATA:
NR DOF ENTS/VERTs: 99 ; nreq = 1 ; nr_sol = 3

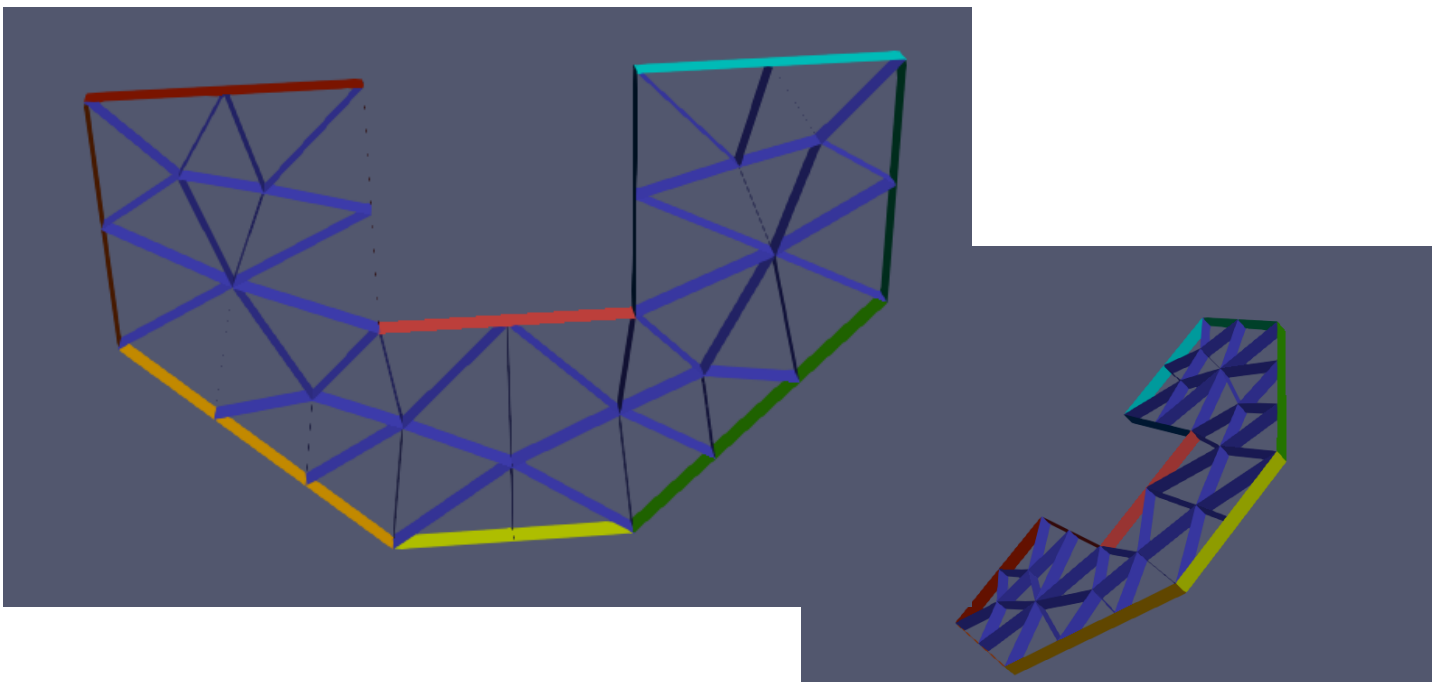
Choose a command from the menu:
  t - solve the problem (time integration)
  s - solve single heat problem
  e - compute error for test problems with known exact solution
  z - compute error (ZZ aproximation)
  p - postprocessing
  g - launch graphics module
  m - perform uniform mesh refinement
  a - 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
```

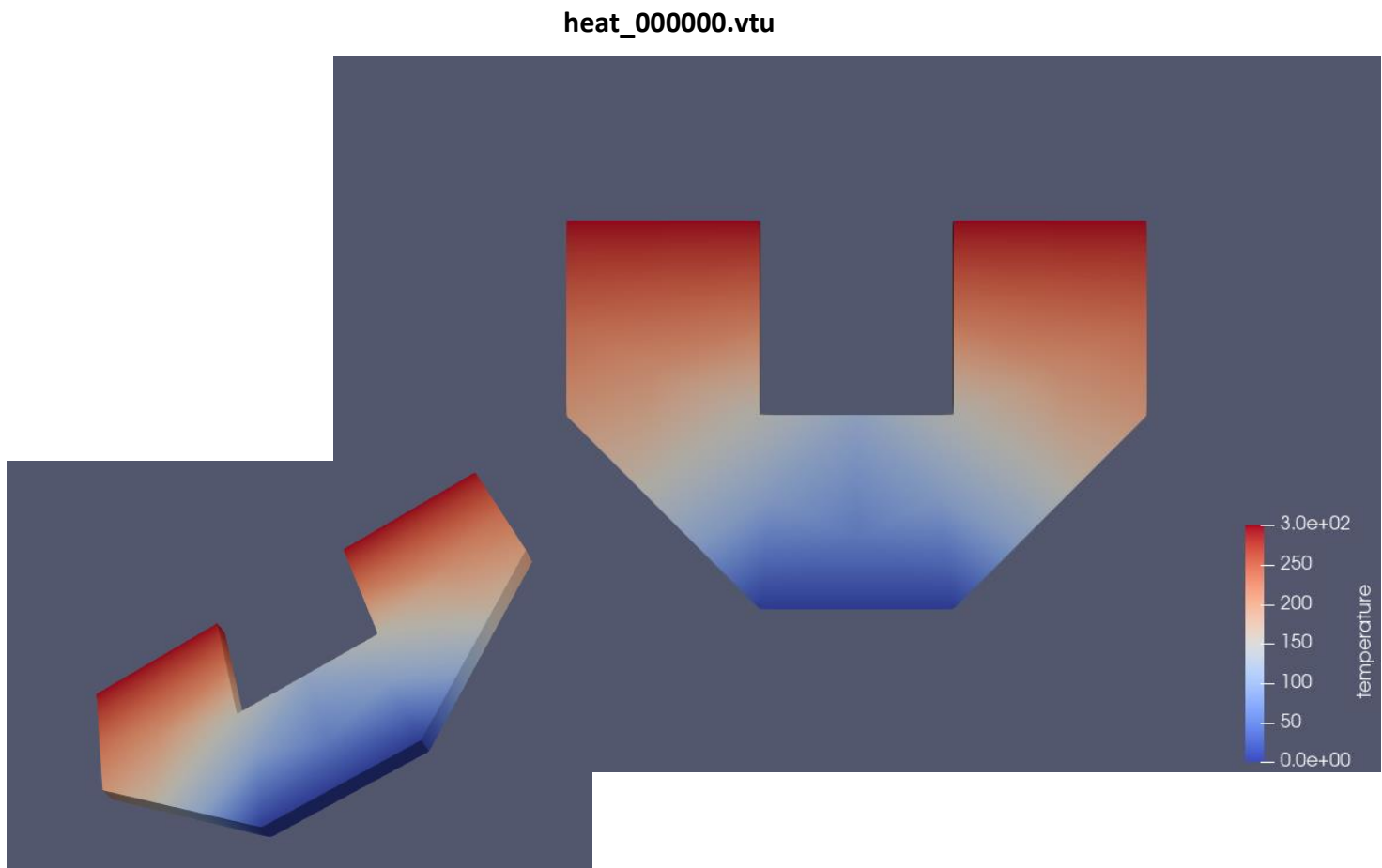
Kolejno odpala się opcje t, s, v, q. W efekcie powstają 4 pliki

```
-rw-r--r-- 1 litewka_jakub users 3545 mar 14 22:46 heat_000000_BC.vtu
-rw-r--r-- 1 litewka_jakub users 2581 mar 14 22:46 heat_000000.partmesh
-rw-r--r-- 1 litewka_jakub users 3347 mar 14 22:46 heat_000000.vtu
-rw-r--r-- 1 litewka_jakub users 252 mar 15 14:51 heat_0000.pvd
```

W programie PARAVIEW można otworzyć powstałe pliki otrzymując następujący efekt:

heat_000000_BC.vtu





Wnioski: Z połączenia programów Gmsh, ModFEM oraz PARAVIEW można stworzyć figurę wraz z rozkładem temperaturowym.

Sam program Gmsh pomaga w zobrazowaniu samej figury w 2D, bądź 3D.

ModFEM zajmuje się obliczeniami dotyczącymi rozkładu temperatury (heat problem) oraz stworzenia plików obrazowanych przez PARAVIEW.

Zadanie: Tworzenie modelu temperatury o określonym kształcie	OCENA własna studenta w % (0-100)	OCENA prowadzącego w % (0-100)
Zad. 1 – utworzenie plików siatek i umieszczenie na serwerze HONORATA, kilka pierwszych linijek pliku powinno znaleźć się w sprawozdaniu	100%	
Zad. 2 – przygotowanie pliku z warunkami brzegowym i umieszczenia go na serwerze HONORATA, opis pliku w sprawozdaniu	100%	
Zad. 3 – utworzenie zrzutów ekranu z programu Paraview oraz umieszczenie ich wraz z opisem w sprawozdaniu	100%	
Zad. 4 - dodatkowa praca z programem Paraview (odpowiednie zrzuty ekranu powinny znaleźć się w sprawozdaniu)	100%	
ŁĄCZNIE (400):	400	