Wykonał: Rafał Olech	Numer albumu: 400876	Grupa: 6	Kierunek: Informatyka Techniczna
Tytuł: Sprawozdanie z projektu	Prowadzący: dr inż. Piotr Kustra	Data oddania: 20.01.2022	Ocena:

1. Wstęp teoretyczny :

Projekt zawiera wykorzystanie równania Fouriera do wyznaczenia rozkładu temperatur w elemencie skończonym, w różnych miejscach elementu i w różnych krokach czasowych.

Równanie Fouriera opisuje zjawiska cieplne zachodzące w stanie ustalonym i przedstawia się go w postaci:

$$div\big(k(t)grad(t)\big) + Q = 0.$$

Powyższy wzór można zapisać w postaci:

$$\frac{\partial}{\partial x}\left(k_x(t)\frac{\partial t}{\partial x}\right) + \frac{\partial}{\partial y}\left(k_y(t)\frac{\partial t}{\partial y}\right) + \frac{\partial}{\partial z}\left(k_z(t)\frac{\partial t}{\partial z}\right) + Q = 0,$$

gdzie:
$k_x(t)$, $k_y(t)$, $k_z(t)$ – anizotropowe współczynniki przewodzenia ciepła zależne od temperatury t,
Q – prędkość generowania ciepła.

Aby rozwiązać równanie Fouriera należy znaleźć minimum takiego funkcjonału, dla którego powyższe równanie jest równaniem Eulera. Funkcjonał taki ma postać:

$$J = \int_V \left(\frac{k(t)}{2}\left(\left(\frac{\partial t}{\partial x}\right)^2 + \left(\frac{\partial t}{\partial y}\right)^2 + \left(\frac{\partial t}{\partial z}\right)^2\right) - Qt\right) dV,$$

Funkcja $t(x, y, z)$ musi spełniać określone warunki brzegowe na powierzchni rozpatrywanego obszaru.

W naszym problemie korzystamy z warunku brzegowego III rodzaju zwanym warunkiem brzegowym Robina dotyczącym konwekcyjnej wymiany ciepła. Warunek ten opisuje temperaturę płynu(np. powietrza, gazu, medium chłodzącego) otaczającego rozpatrywane ciało oraz współczynnik przejmowania ciepła w każdym miejscu powierzchni ciała oraz w każdej chwili. Można je opisać za pomocą wzoru:

$$\boldsymbol{q_s = \alpha_{konw} \cdot (t - t_\infty)},$$

gdzie:
$\boldsymbol{\alpha_{konw}}$ – współczynnik konwekcyjnej wymiany ciepła,
$\boldsymbol{t}$ – temperatura na powierzchni,
$\boldsymbol{t_\infty}$ – temperatura otoczenia.

Wzór ten mówi o tym, że gęstość strumienia ciepła jest wprost proporcjonalna do różnicy temperatur powierzchni ciała i otoczenia.

W naszym zadaniu na powierzchni rozpatrywanego obszaru zadany jest strumień ciepła q według prawa konwekcji.

Nie da się w sposób bezpośredni wprowadzić warunków brzegowych do funkcjonału, a więc narzuca się je poprzez dodanie do funkcjonału całki w postaci:

$$\int_S \frac{\alpha}{2}(t - t_\infty)^2 dS + \int_S qt dS,$$

gdzie:
S – powierzchnia, na której zadane są warunki brzegowe.

Po dodaniu powyższej całki do funkcjonału otrzymuje się:

$$J = \int_V \left(\frac{k(t)}{2}((\frac{\partial t}{\partial x})^2 + (\frac{\partial t}{\partial y})^2 + (\frac{\partial t}{\partial z})^2) - Qt\right) dV + \int_S \frac{\alpha}{2}(t - t_\infty)^2 dS + \int_S qt dS,$$

Dyskretyzacja przedstawionego problemu polega na podzieleniu rozpatrywanego obszaru na elementy i przedstawieniu temperatury wewnątrz elementu, jako funkcji wartości węzłowych zgodnie z zależnością:

$$t = \sum_{i=1}^{n} N_i t_i = \{N\}^T \{t\}.$$

Funkcjonał po wprowadzeniu powyższej zależności ma postać:

$$J = \int_V \left(\frac{k}{2}\left(\left(\left\{\frac{\partial\{N\}}{\partial x}\right\}^T \{t\}\right)^2 + \left(\left\{\frac{\partial\{N\}}{\partial y}\right\}^T \{t\}\right)^2 + \left(\left\{\frac{\partial\{N\}}{\partial z}\right\}^T \{t\}\right)^2\right) - Q\{N\}^T\{t\}\right) dV$$

$$+ \int_S \frac{\alpha}{2}(\{N\}^T\{t\} - t_\infty)^2 dS + \int_S q\{N\}^T\{t\}\, dS$$

Wyznaczony powyższy wzór zasadniczo opisuje problem rozwiązywany w projekcie. Aby zminimalizować funkcjonał należy obliczyć pochodne cząstkowe względem wartości węzłowych temperatury {t}, co prowadzi do następującego układu równań:

$$\frac{\partial J}{\partial\{t\}} = \int_V \left(k\left(\left\{\frac{\partial\{N\}}{\partial x}\right\}\left\{\frac{\partial\{N\}}{\partial x}\right\}^T + \left\{\frac{\partial\{N\}}{\partial y}\right\}\left\{\frac{\partial\{N\}}{\partial y}\right\}^T + \left\{\frac{\partial\{N\}}{\partial z}\right\}\left\{\frac{\partial\{N\}}{\partial z}\right\}^T\right)\{t\} - Q\{N\}\right) dV$$

$$+ \int_S \alpha(\{N\}^T\{t\} - t_\infty)\{N\} dS + \int_S q\{N\} dS = 0$$

Powyższe równanie różniczkowe Fouriera w programie jest przedstawione i dalej wyliczane w postaci macierzowej i ma postać:

$$[H]\{t\} + \{P\} = 0$$

W powyższym równaniu macierz $[H]$ opisana jest następującą zależnością:

$$[H] = \int_V k(t)\left(\left\{\frac{\partial\{N\}}{\partial x}\right\}\left\{\frac{\partial\{N\}}{\partial x}\right\}^T + \left\{\frac{\partial\{N\}}{\partial y}\right\}\left\{\frac{\partial\{N\}}{\partial y}\right\}^T + \left\{\frac{\partial\{N\}}{\partial z}\right\}\left\{\frac{\partial\{N\}}{\partial z}\right\}^T\right)dV + \int_S \alpha\{N\}\{N\}^T dS\,,$$

natomiast macierz $\{P\}$ opisana jest zależnością:

$$\{P\} = -\int_S \alpha\{N\}\, t_\infty\, dS - \int_V Q\{N\}dV + \int_S q\{N\}dS.$$

Po wyznaczeniu w programie macierzy $[H]$ i $\{P\}$ wykorzystując metodę eliminacji Gaussa wyliczany jest układ równań a jako wynik otrzymywany jest wektor temperatur w węzłach, z którego wartości są następnie przypisywane do odpowiednich węzłów siatki MES jako temperatury początkowe do następnej iteracji. Czynności się powtarzają w zależności od zdefiniowanej w programie liczby iteracji.

2. Charakterystyka kodu :

2.1 Struktury w programie:

W programie znajdują się struktury określające budowę siatki elementów skończonych i jej bardziej szczegółowych elementów takich jak elementy, węzły itd. Utworzone są także struktury zawierające dane potrzebne do symulacji oraz struktury pomagające w dalszych obliczeniach.

Struktury zaimplementowane w programie:

- **node** – struktura określająca pojedynczy węzeł w siatce. Posiada takie pola jak współrzędne węzła na osi x oraz y oraz temperaturę początkową w węźle (temp0).

- **GlobalData** – struktura, w której znajdują się dane globalne określające wartości początkowe różnych parametrów, wymiary siatki elementów skończonych oraz dane potrzebne do obliczeń składowych układu równań.
  W strukturze znajdują się poniższe pola:
  - *nPC* – liczba punktów całkowania,
  - *t0* – temperatura otoczenia [°C],
  - *initialTemp* – temperatura początkowa [°C],
  - *stime* – czas symulacji [s],
  - *sst* – czas kroku symulacji [s],
  - *c* – ciepło właściwe $[\frac{J}{kg \cdot °C}]$,
  - *ro* – gęstość $[\frac{kg}{m^3}]$,
  - *conductivity* – przewodność $[\frac{W}{m \cdot °C}]$,
  - *alfa* – alfa $[\frac{W}{m^2 \cdot K}]$.

- **point** – struktura określająca pojedynczy punkt całkowania. Posiada takie pola jak współrzędne punktu całkowania ksi, eta.

- **wall** – struktura określająca pojedynczą ścianę elementu. Każdy element posiada cztery ściany. Struktura posiada pola typu point oznaczające dwa punkty tworzące ścianę oraz zawiera flagę BC, która określa występowanie warunku brzegowego na ścianie. Flaga BC początkowo ustawiona jest na false i w dalszej części programu wyznaczane jest, na których ścianach jest warunek brzegowy i flaga BC jest modyfikowana.

- **element** – struktura określająca pojedynczy element w siatce MES. Każdy element posiada cztery węzły, których ID znajdują się w tablicy czteroelementowej o nazwie ID. W strukturze znajduje się także tablica ścian elementu, która również jest czteroelementowa.

- **element4** – struktura, w której znajdują się tablice, które będą w dalszej części programu przechowywały pochodne ksi oraz eta.

- **grid** – struktura, która opisuje budowę całej siatki elementów skończonych. Zawiera pola, które określają wymiary siatki takie jak wysokość, szerokość, liczba węzłów na wysokość, liczba węzłów na szerokość, całkowita liczba elementów oraz całkowita liczba węzłów. Struktura grid zawiera także tablice węzłów oraz tablice elementów. W tej strukturze wyznaczane są ID wszystkich węzłów w siatce oraz wyznaczane są ich współrzędne.

- **pcKsiEta** – struktura, która zawiera możliwe wartości współrzędnych ksi, eta punktów całkowania. Zastosowanie tej struktury ułatwia dalsze obliczenia.

- **UniversalElement** – struktura, w której znajdują się współrzędne ksi, eta punktów całkowania oraz obliczone funkcje kształtu dla każdego punktu całkowania.

2.2 Wyznaczanie warunków brzegowych:

Pierwszą rzeczą przeprowadzoną w programie jest wyznaczenie warunków brzegowych na ścianach elementów w siatce elementów skończonych. W programie fragment kodu odpowiedzialny za to wyznaczanie znajduje się w głównej funkcji programu, czyli w funkcji main. Wyznaczanie warunków brzegowych na ścianach odbywa przy pomocy współrzędnych x, y węzłów badając ich położenie w siatce. W elemencie warunek brzegowy może znajdować się na dwóch ścianach, jeżeli element znajduję się w rogu siatki. Na jednej ścianie jeżeli element ma styczność z otoczeniem np. z powietrzem tylko przez jedną ścianę i są to elementy które znajdują się na zewnętrznych ścianach siatki ale nie znajdują się w narożnikach. Elementy nie posiadające warunku brzegowego na żadnej ze ścian są to elementy, które znajdują się w środku siatki i nie mają styczności z otoczeniem przez żadną ze ścian. Rozgraniczenie węzłów ze względu na położenie w programie odbywa się przy pomocy instrukcji warunkowych if-else. Jeżeli na ścianie elementu znajduje się warunek brzegowy flaga BC zmieniana jest na true.

2.3 Całkowanie metodą Gaussa:

Następną rzeczą obliczaną w programie jest całkowanie metodą Gaussa, które w programie jest zaimplementowane w następujących wariantach:

- w przestrzeni 1d stosując 2 punktowy schemat całkowania,

- w przestrzeni 1d stosując 3 punktowy schemat całkowania,
- w przestrzeni 2d stosując 2 punktowy schemat całkowania,
- w przestrzeni 2d stosując 3 punktowy schemat całkowania.

Do przetestowania zaimplementowanej metody całkowania w programie znajdują się definicje dwóch funkcji, na których wykonywane są obliczenia. Całkowanie w przestrzeni 1d przeprowadzane jest na funkcji:

$$f(x) = 5x^2 + 3x + 6,$$

która znajduje się w programie w funkcji *fun1()*.

Całkowanie w przestrzeni 2d przeprowadzane jest na funkcji:

$$f(x, y) = 5x^2y^2 + 3xy + 6,$$

która znajduje się w programie w funkcji *fun2()*.

Całkowanie metodą Gaussa sprowadza się do określenia wartości funkcji w punkcie całkowania przemnożonej przez wagę punktu całkowania. Wzór rozwiązujący problem w przestrzeni 1d ma postać:

$$\int_{-1}^{1} f(\xi)d\xi = \sum_{i=1}^{n} w_i \cdot f(\xi_i),$$

gdzie:

n – liczba punktów całkowania.

Analogiczny całkowanie przebiega w przestrzeni 2d. W tym przypadku funkcja podcałkowa jest funkcją dwóch zmiennych, a wzór rozwiązujący problem w przestrzeni 2d ma postać:

$$\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta)d\xi\, d\eta = \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j f(\xi_i, \eta_i)$$

Wynikiem całkowania w każdym z punktów całkowania jest prostopadłościan o polu podstawy definiowanym wartością wag. Waga razy waga definiuje pole podstawy, a wysokość to wartość funkcji w punkcie całkowania.

Do całkowania w przestrzeni 1d oraz 2d używa się danych, które są opracowane oraz stabelaryzowane i nazywane są kwadraturami Gaussa. Znajdują się w nich współrzędne punktu całkowania oraz ich wagi.

W programie powyższe wzory na całkowanie 1d oraz 2d są zaimplementowane w funkcjach odpowiednio *Gauss1D* oraz *Gauss2D*. Działania w tych funkcjach są przeprowadzane przy użyciu instrukcji wielokrotnego wyboru switch, która jako argument przyjmuję liczbę ilu punktowy jest schemat całkowania.

2.4 Jakobian:

Następną rzeczą, która jest robiona w programie to obliczanie jakobianu poprzez wywołanie funkcji *Jacobian()* z odpowiednimi parametrami. Jakobian liczony jest zawsze dla każdego punktu całkowania. A więc w programie znajdują się cztery obliczone Jakobiany ponieważ są zdefiniowane cztery punkty całkowania. W funkcji *Jacobian()* po kolei dla każdego punktu całkowania obliczane są pochodne: $\frac{\partial x}{\partial \xi}$, $\frac{\partial x}{\partial \eta}$, $\frac{\partial y}{\partial \xi}$, $\frac{\partial y}{\partial \eta}$. Po obliczonych pochodnych każdy punkt całkowania posiada macierz w postaci:

$$\begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}$$

Następnie wyliczane są jakobiany det[J] dla każdego punktu całkowania. Det[J] jest to wyznacznik macierzy zawierającej obliczone wcześniej pochodne, czyli macierzy przedstawionej powyżej i jest to stosunek układu globalnego do układu lokalnego. Następnie obliczana jest odwrotność Jakobianu dla każdego punktu całkowania ($\frac{1}{\det[J]}$).

Mając obliczone Jakobiany oraz ich odwrotności dla każdego punktu całkowania są obliczane Jakobiany przekształcenia i mają postać:

$$\begin{bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{bmatrix} = \frac{1}{\det[J]} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial y}{\partial \xi} \\ -\frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \xi} \end{bmatrix} \begin{bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{bmatrix}$$

2.5 Całkowanie macierzy H:

W programie znajduję się funkcja *calculateTabH(),* która służy do wyliczania lokalnych macierzy H o rozmiarze 4x4 dla każdego punktu całkowania. Obliczane macierze H opisane są następującą zależnością:

$$[H] = \int_V k(t) \left(\left\{ \frac{\partial \{N\}}{\partial x} \right\} \left\{ \frac{\partial \{N\}}{\partial x} \right\}^T + \left\{ \frac{\partial \{N\}}{\partial y} \right\} \left\{ \frac{\partial \{N\}}{\partial y} \right\}^T \right) dV$$

Aby wyliczyć macierz H dla punktu całkowania w funkcji *calculateTabH()* na początku obliczane są pochodne: $\frac{\partial \{N\}}{\partial x}$ oraz $\frac{\partial \{N\}}{\partial y}$ z funkcji kształtu. Następnie liczone są dwa iloczyny, pierwszy z nich to mnożenie macierzy zawierającej pochodne po x z funkcji kształtu $\left\{ \frac{\partial \{N\}}{\partial x} \right\}$ z tą samą macierzą ale transponowaną $\left\{ \frac{\partial \{N\}}{\partial x} \right\}^T$. Drugi z iloczynów liczony jest analogicznie, ale pochodne z funkcji kształtu są liczone po y $\left\{ \frac{\partial \{N\}}{\partial y} \right\}$, a następnie są przemnożone przez macierz transponowaną $\left\{ \frac{\partial \{N\}}{\partial y} \right\}^T$.

Mając wyliczone takie dwa iloczyny sumuje się je, a następnie przemnaża przez przewodność (conductivity), której wartość znajduję się w strukturze GlobalData. Całkowanie realizowane jest poprzez przemnożenie wyniku przez Jakobian (det[J]) punktu całkowania co zastępuje dV. Macierz H obliczana jest dla każdego punktu całkowania, a następnie macierze te są sumowane do macierzy H, która jest wynikiem wywołania tej funkcji.

$$H = H_{pc1} + H_{pc2} + H_{pc3} + H_{pc4}$$

2.6 Całkowanie macierzy H_{BC}:

Macierz H_{BC} jest liczona dla każdego punktu całkowania i określona jest wzorem:

$$[H_{BC}] = \int_S \alpha(\{N\}\{N\}^T)\, dS$$

Powyższy wzór można przedstawić także w postaci:

$$\sum_{i=1}^{n_{pc}} f(pc_i) w_i \det[J]$$

W programie problem ten został rozwiązany poprzez początkowo obliczenie funkcji kształtu dla każdego punktu całkowania. Następnie liczone są dwa iloczyny, które zawierają mnożenie macierzy zawierającej wartości funkcji kształtu $\{N\}$ punktu całkowania, dla którego obliczana jest macierz H_{BC} z tą samą macierzą ale transponowaną $\{N\}^T$ i przemnożony jeszcze razy odpowiednia waga oraz alfa (współczynnik konwekcyjnej wymiany ciepła), której wartość znajduję się w strukturze GlobalData. Na końcu sumowane są iloczyny i przemnażane przez Jakobian (det[J]), który wyznaczany jest z następującego wzoru:

$$\det[J] = \frac{L}{2},$$

gdzie:

L – długość ściany,

W powyższym wzorze det[J] określa stosunek układu globalnego do lokalnego, a długość ściany (L) jest dzielona przez 2 ponieważ w układzie lokalnym całkowanie zachodzi w przedziale <-1,1> i jest to długość przedziału.

W równaniu na H_{BC} (dS) realizowane jest poprzez przemnożenie wyniku przez Jakobian det[J].

Macierz H_{BC} jest liczona dla każdego elementu uwzględniając warunek brzegowy na ścianach. Sprawdzane są położenia węzłów w siatce na takiej samej zasadzie jak przy określaniu flagi BC i dla elementów, posiadających na ścianie warunek brzegowy liczone są odpowiednio macierze H_{BC}, jeżeli na żadnej ścianie nie ma warunku brzegowego macierz H_{BC} dla takiego elementu wypełniona jest zerami. Wszystkie działania dotyczące obliczania macierzy H_{BC} wyżej opisane w programie

znajdują się w funkcji o nazwie *wektorP()*. Funkcja ta zawiera jednocześnie wyznaczanie macierzy H_{BC} oraz wektorów ***P*** ze względu na bardzo podobny schemat ich wyliczania.

2.7 Całkowanie wektora P:

Wektor ***P*** jest liczony dla każdego punktu całkowania i określony jest wzorem:

$$[P] = \int_S \alpha\{N\}t_{ot}\, dS$$

Powyższy wzór można przedstawić także w postaci:

$$\sum_{i=1}^{n_{pc}} f(pc_i)w_i \det[J]$$

W programie wyznaczanie wektorów ***P*** znajduję się w funkcji *wektorP()*. Do obliczeń potrzebne są funkcje kształtu dla każdego punktu całkowania i w programie jest możliwe skorzystanie z wcześniej już wyliczonych na potrzeby wyznaczenia macierzy H_{BC}. Następnie liczone są dwa iloczyny, które zawierają mnożenie macierzy zawierającej wartości funkcji kształtu $\{N\}$ punktu całkowania, dla którego obliczany jest wektor ***P*** razy odpowiednia waga razy alfa (współczynnik konwekcyjnej wymiany ciepła) i przemnożona jeszcze przez t0 (temperatura otoczenia). Wartości dwóch ostatnich zmiennych znajdują się w strukturze GlobalData. Na końcu sumowane są iloczyny i przemnażane przez Jakobian (det[J]), który wyznaczony został już przy okazji wyznaczania macierzy H_{BC}. Wektor ***P*** obliczany jest dla każdego elementu na takiej samej zasadzie jak H_{BC} dla każdego elementu czyli w oparciu o położenie węzłów na siatce.

2.8 Generyczność wektora P:

Mając obliczone wektory lokalne ***P,*** które zostały wyznaczone w funkcji *wektorP()* w programie wywoływana jest funkcja *generycznoscP()*. W funkcji tej znajduję się agregacja cztero-elementowych lokalnych wektorów ***P*** do wektora globalnego, który ma rozmiar nN (liczba węzłów w siatce). Funkcja *generycznoscP()* jako argumenty przyjmuje pusty wektor globalny przygotowany na wyniki oraz tablicę zawierającą wektory ***P*** poszczególnych elementów.

Agregacja polega na sumowania lokalnych wektorów ***P*** i zapisywaniu ich do wektora globalnego, który jest wynikiem tej funkcji.

2.9 Całkowanie macierzy C:

W programie znajduję się funkcja *calculateTabC()*, która służy do wyliczania lokalnych macierzy C o rozmiarze 4x4 dla każdego punktu całkowania. Obliczane macierze C opisane są następującą zależnością:

$$[C] = \int_V \rho c_p(\{N\}\{N\}^T)dV$$

W celu wyznaczenia lokalnych macierzy ***C*** w programie na początku pobierane są już wcześniej wyliczone funkcje kształtu dla każdego punktu całkowania $\{N\}$ znajdujące się w strukturze universalElement. Drugim krokiem jest wyliczenie macierzy transponowanych z macierzy funkcji kształtu $\{N\}^T$ dla każdego punktu całkowania korzystając z utworzonej w programie pomocniczej funkcji *transponujTablice1x4()*. Mając wyznaczone macierze funkcji kształtu i ich transpozycje następuje obliczenie zasadnicze czyli całkowanie. W celu obliczenia całki mnożona jest wartość ciepła właściwego (c) razy gęstość (ro) razy iloczyn składający się z macierzy $\{N\}$ oraz macierzy $\{N\}^T$. Na końcu wszystko przemnażane jest jeszcze przez wcześniej obliczony jakobian (det[J]) dla punktu całkowania, dla którego obliczana jest macierz ***C.*** Opisaną powyżej procedurę całkowania macierzy ***C*** dla pojedynczego punktu całkowania w programie przedstawia poniższa linijka:

```
C1[i][j] = globalData.c * globalData.ro * (PC1NToTabCTransponowana[i][0] * universalElement.PC1NToTabC[0][j]) * detJTab[0];
```

Powyższa operacja powtarzana jest dla wszystkich punktów całkowania. Macierz ***C*** obliczana jest dla każdego punktu całkowania, a następnie macierze te są sumowane do macierzy ***C***, która jest wynikiem wywołania tej funkcji.

$$C = C_{pc1} + C_{pc2} + C_{pc3} + C_{pc4}$$

2.10 Agregacja:

W programie została utworzona funkcja *agregacja(),* która służy do agregowania macierzy lokalnych do macierzy globalnej. Jako argumenty funkcja ta przyjmuję siatkę oraz macierz lokalną do zagregowania. Macierz wynikowa agregacji ma wymiary nN x nN, gdzie nN oznacza liczbę węzłów w siatce elementów skończonych. Elementy macierzy przesłanej jako argument są sumowane do macierzy wynikowej.
Funkcja agregacja() została użyta w programie do zagregowania macierzy C oraz H.

2.11 Agregacja macierzy H_{BC}:

W programie znajduje się funkcja *agregacjaHBC(),* która służy do agregowania macierzy lokalnych H_{BC} do macierzy globalnej. Jako argumenty funkcja ta przyjmuję siatkę oraz tablicę wyliczonych tablic H_{BC} wszystkich elementów do zagregowania. Agregacja w tej funkcji działa na takiej samej zasadzie jak agregacja w funkcji *agregacja()* ale jest przeznaczona do agregacji tylko macierzy H_{BC} i różni się tym, że w argumencie funkcja *agregacjaHBC()* przyjmuje tablicę lokalnych tablic. Jako wynik wywołania funkcji powstaje macierz globalna o wymiarach nN x nN, gdzie nN oznacza liczbę węzłów w siatce elementów skończonych.

2.12 Obliczanie Globalnej macierzy H:

W programie znajduje się funkcja *calculateMatrixH()*, która służy do wyliczania globalnej macierzy ***H*** potrzebnej do dalszych obliczeń. Funkcja jako parametry przyjmuje strukturę GlobalData , siatkę MES, zagregowane macierze ***C, H, HBC*** oraz puste tablice przygotowane na wyniki takie jak matrixH, do której będzie zapisywany ostateczny wynik funkcji oraz tabGlobalH, do której będą sumowane macierze ***H*** oraz ***HBC.***

Funkcja *calculateMatrixH()* składa się z dwóch części. W pierwszej części obliczana jest przesłana jako argument macierz tabGlobalH, która składa się z zsumowanych elementów macierzy ***H*** oraz ***HBC.***

W drugiej części funkcji znajduje się zasadnicze wyliczenie globalnego matrixa ***H***, który ma następującą postać:

$$Matrix\ [H] = [H] + [C]/dT$$

gdzie:

[H] – zsumowana macierz ***[H]*** i ***[HBC]***,

dT – czas kroku symulacji (w programie oznaczony jako sst).

2.13 Obliczanie Globalnego wektora P:

W programie znajduje się funkcja *P_vector()*, która służy do wyliczania globalnego wektora ***P*** potrzebnej do dalszych obliczeń. Funkcja jako parametry przyjmuje strukturę GlobalData , siatkę MES, zagregowane macierze ***C,*** zagregowany wektor ***P*** oraz pustą tablicę vectorP, przygotowaną na wyniki. W funkcji obliczany jest globalny wektor ***P***, który ma następującą postać:

$$\{P\} = \{P\} + \left\{\frac{[C]}{dT}\right\} \cdot \{T0\}$$

gdzie:

dT – czas kroku symulacji (w programie oznaczony jako sst),

{T0} – wektor temperatur początkowych w węzłach siatki.

2.14 Obliczanie macierzy HP:

Mając wyliczoną globalną macierz ***H*** z funkcji *calculateMatrixH()* oraz globalny wektor ***P*** wyliczony w funkcji *P_vector()* w programie obliczana jest macierz ***HP***, która jest potrzebna bezpośrednio do eliminacji Gaussa w celu obliczenia nowego wektora temperatur w węzłach siatki. Wyliczanie macierzy ***HP*** w programie znajduje się w funkcji *calculateHP(),* która jako argumenty

przyjmuję siatkę, globalną macierz ***H***, globalny wektor ***P*** oraz pustą tablicę HPTab, do której będą zapisywane wyniki. Macierz ***HP*** powstała przez złożenie wektora ***P*** i macierzy ***H***.

2.15 Eliminacja Gaussa:

W programie znajduje się funkcja *EliminacjaGaussa(),* która zawiera algorytm rozwiązywania układu równań metodą Gaussa bez pivotingu. W metodzie tej wyróżnia się dwa etapy: postępowanie proste oraz postępowanie odwrotne.

Etap pierwszy czyli postępowanie proste polega sprowadzeniu układu do postaci górnie trójkątnej.

W drugim etapie czyli postępowaniu odwrotnym w celu znalezienia rozwiązania układu równań, korzysta się z uzyskanej macierzy trójkątnej górnej i poniższych wzorów:

$$x_n = \frac{{b_n}^{(n)}}{{a_{nn}}^{(n)}}$$

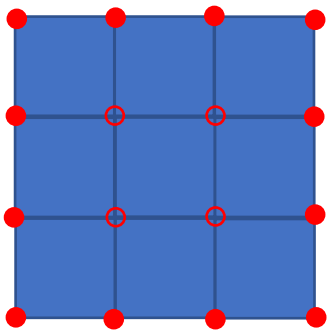
$$x_i = \frac{{b_i}^{(n)} - \sum_{k=i+1}^{n} {a_{ik}}^{(n)} x_k}{{a_{ii}}^{(n)}} \; dla\; i = n-1, \dots 0$$

gdzie:

Indeks w nawiasie okrągłym oznacza kolejny krok realizowany metodą eliminacji Gaussa.

Funkcja *EliminacjaGaussa()* potrzebna jest do wyznaczania nowych temperatur w węzłach siatki i jako argumenty przyjmuje macierz ***HP*** oraz liczbę węzłów w siatce MES. Wynikiem jest wektor temperatur w węzłach siatki, który przy następnej iteracji stanowi wektor z temperaturami początkowymi w węzłach.

3. Model siatki MES 4x4:



gdzie:

- ● - BC = 1 (występuje warunek brzegowy)
- ○ - BC = 0 (nie występuje warunek brzegowy)

3 Przeprowadzenie testów (dla siatki 4x4):

Poniżej znajduje się zestawienie oraz porównanie wyników uzyskanych za pomocą napisanego programu z wynikami znajdującymi się na stronie przedmiotu.

3.1 Macierz [C]:

Wynik uzyskany w programie:

```
Macierz C:
[674.074] [337.037] [0] [0] [337.037] [168.519] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]
[337.037] [1348.15] [337.037] [0] [168.519] [674.074] [168.519] [0] [0] [0] [0] [0] [0] [0] [0] [0]
[0] [337.037] [1348.15] [337.037] [0] [168.519] [674.074] [168.519] [0] [0] [0] [0] [0] [0] [0] [0]
[0] [0] [337.037] [674.074] [0] [0] [168.519] [337.037] [0] [0] [0] [0] [0] [0] [0] [0]
[337.037] [168.519] [0] [0] [1348.15] [674.074] [0] [0] [337.037] [168.519] [0] [0] [0] [0] [0] [0]
[168.519] [674.074] [168.519] [0] [674.074] [2696.3] [674.074] [0] [168.519] [674.074] [168.519] [0] [0] [0] [0] [0]
[0] [168.519] [674.074] [168.519] [0] [674.074] [2696.3] [674.074] [0] [168.519] [674.074] [168.519] [0] [0] [0] [0]
[0] [0] [168.519] [337.037] [0] [0] [674.074] [1348.15] [0] [0] [168.519] [337.037] [0] [0] [0] [0]
[0] [0] [0] [0] [337.037] [168.519] [0] [0] [1348.15] [674.074] [0] [0] [337.037] [168.519] [0] [0]
[0] [0] [0] [0] [168.519] [674.074] [168.519] [0] [674.074] [2696.3] [674.074] [0] [168.519] [674.074] [168.519] [0]
[0] [0] [0] [0] [0] [168.519] [674.074] [168.519] [0] [674.074] [2696.3] [674.074] [0] [168.519] [674.074] [168.519]
[0] [0] [0] [0] [0] [0] [168.519] [337.037] [0] [0] [674.074] [1348.15] [0] [0] [168.519] [337.037]
[0] [0] [0] [0] [0] [0] [0] [0] [337.037] [168.519] [0] [0] [674.074] [337.037] [0] [0]
[0] [0] [0] [0] [0] [0] [0] [0] [168.519] [674.074] [168.519] [0] [337.037] [1348.15] [337.037] [0]
[0] [0] [0] [0] [0] [0] [0] [0] [0] [168.519] [674.074] [168.519] [0] [337.037] [1348.15] [337.037]
[0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [168.519] [337.037] [0] [0] [337.037] [674.074]
```

Wynik znajdujący się na stronie przedmiotu:

```
_____ Interation 0 _____
_____Matrix [C]_____
674.074 337.037 0 0 337.037 168.519 0 0 0 0 0 0 0 0 0 0
337.037 1348.15 337.037 0 168.519 674.074 168.519 0 0 0 0 0 0 0 0 0
0 337.037 1348.15 337.037 0 168.519 674.074 168.519 0 0 0 0 0 0 0 0
0 0 337.037 674.074 0 0 168.519 337.037 0 0 0 0 0 0 0 0
337.037 168.519 0 0 1348.15 674.074 0 0 337.037 168.519 0 0 0 0 0 0
168.519 674.074 168.519 0 674.074 2696.3 674.074 0 168.519 674.074 168.519 0 0 0 0 0
0 168.519 674.074 168.519 0 674.074 2696.3 674.074 0 168.519 674.074 168.519 0 0 0 0
0 0 168.519 337.037 0 0 674.074 1348.15 0 0 168.519 337.037 0 0 0 0
0 0 0 0 337.037 168.519 0 0 1348.15 674.074 0 0 337.037 168.519 0 0
0 0 0 0 168.519 674.074 168.519 0 674.074 2696.3 674.074 0 168.519 674.074 168.519 0
0 0 0 0 0 168.519 674.074 168.519 0 674.074 2696.3 674.074 0 168.519 674.074 168.519
0 0 0 0 0 0 168.519 337.037 0 0 674.074 1348.15 0 0 168.519 337.037
0 0 0 0 0 0 0 0 337.037 168.519 0 0 674.074 337.037 0 0
0 0 0 0 0 0 0 0 168.519 674.074 168.519 0 337.037 1348.15 337.037 0
0 0 0 0 0 0 0 0 0 168.519 674.074 168.519 0 337.037 1348.15 337.037
0 0 0 0 0 0 0 0 0 0 168.519 337.037 0 0 337.037 674.074
```

3.2 Macierz [H]:

Wynik uzyskany w programie:

```
Macierz H:
[16.6667] [-4.16667] [0] [0] [-4.16667] [-8.33333] [0] [0] [0] [0] [0] [0] [0] [0] [0] [0]
[-4.16667] [33.3333] [-4.16667] [0] [-8.33333] [-8.33333] [-8.33333] [0] [0] [0] [0] [0] [0] [0] [0] [0]
[0] [-4.16667] [33.3333] [-4.16667] [0] [-8.33333] [-8.33333] [-8.33333] [0] [0] [0] [0] [0] [0] [0] [0]
[0] [0] [-4.16667] [16.6667] [0] [0] [-8.33333] [-4.16667] [0] [0] [0] [0] [0] [0] [0] [0]
[-4.16667] [-8.33333] [0] [0] [33.3333] [-8.33333] [0] [0] [-4.16667] [-8.33333] [0] [0] [0] [0] [0] [0]
[-8.33333] [-8.33333] [-8.33333] [0] [-8.33333] [66.6667] [-8.33333] [0] [-8.33333] [-8.33333] [-8.33333] [0] [0] [0] [0] [0]
[0] [-8.33333] [-8.33333] [-8.33333] [0] [-8.33333] [66.6667] [-8.33333] [0] [-8.33333] [-8.33333] [-8.33333] [0] [0] [0] [0]
[0] [0] [-8.33333] [-4.16667] [0] [0] [-8.33333] [33.3333] [0] [0] [-8.33333] [-4.16667] [0] [0] [0] [0]
[0] [0] [0] [0] [-4.16667] [-8.33333] [0] [0] [33.3333] [-8.33333] [0] [0] [-4.16667] [-8.33333] [0] [0]
[0] [0] [0] [0] [-8.33333] [-8.33333] [-8.33333] [0] [-8.33333] [66.6667] [-8.33333] [0] [-8.33333] [-8.33333] [-8.33333] [0]
[0] [0] [0] [0] [0] [-8.33333] [-8.33333] [-8.33333] [0] [-8.33333] [66.6667] [-8.33333] [0] [-8.33333] [-8.33333] [-8.33333]
[0] [0] [0] [0] [0] [0] [-8.33333] [-4.16667] [0] [0] [-8.33333] [33.3333] [0] [0] [-8.33333] [-4.16667]
[0] [0] [0] [0] [0] [0] [0] [0] [-4.16667] [-8.33333] [0] [0] [16.6667] [-4.16667] [0] [0]
[0] [0] [0] [0] [0] [0] [0] [0] [-8.33333] [-8.33333] [-8.33333] [0] [-4.16667] [33.3333] [-4.16667] [0]
[0] [0] [0] [0] [0] [0] [0] [0] [0] [-8.33333] [-8.33333] [-8.33333] [0] [-4.16667] [33.3333] [-4.16667]
[0] [0] [0] [0] [0] [0] [0] [0] [0] [0] [-8.33333] [-4.16667] [0] [0] [-4.16667] [16.6667]
```

Wynik znajdujący się na stronie przedmiotu:

```
_____ Interation 0 _____
_____ Matrix [H] _____
16.6667 -4.16667 0 0 -4.16667 -8.33333 0 0 0 0 0 0 0 0 0 0
-4.16667 33.3333 -4.16667 0 -8.33333 -8.33333 -8.33333 0 0 0 0 0 0 0 0 0
0 -4.16667 33.3333 -4.16667 0 -8.33333 -8.33333 -8.33333 0 0 0 0 0 0 0 0
0 0 -4.16667 16.6667 0 0 -8.33333 -4.16667 0 0 0 0 0 0 0 0
-4.16667 -8.33333 0 0 33.3333 -8.33333 0 0 -4.16667 -8.33333 0 0 0 0 0 0
-8.33333 -8.33333 -8.33333 0 -8.33333 66.6667 -8.33333 0 -8.33333 -8.33333 -8.33333 0 0 0 0 0
0 -8.33333 -8.33333 -8.33333 0 -8.33333 66.6667 -8.33333 0 -8.33333 -8.33333 -8.33333 0 0 0 0
0 0 -8.33333 -4.16667 0 0 -8.33333 33.3333 0 0 -8.33333 -4.16667 0 0 0 0
0 0 0 0 -4.16667 -8.33333 0 0 33.3333 -8.33333 0 0 -4.16667 -8.33333 0 0
0 0 0 0 -8.33333 -8.33333 -8.33333 0 -8.33333 66.6667 -8.33333 0 -8.33333 -8.33333 -8.33333 0
0 0 0 0 0 -8.33333 -8.33333 -8.33333 0 -8.33333 66.6667 -8.33333 0 -8.33333 -8.33333 -8.33333
0 0 0 0 0 0 -8.33333 -4.16667 0 0 -8.33333 33.3333 0 0 -8.33333 -4.16667
0 0 0 0 0 0 0 0 -4.16667 -8.33333 0 0 16.6667 -4.16667 0 0
0 0 0 0 0 0 0 0 -8.33333 -8.33333 -8.33333 0 -4.16667 33.3333 -4.16667 0
0 0 0 0 0 0 0 0 0 -8.33333 -8.33333 -8.33333 0 -4.16667 33.3333 -4.16667
0 0 0 0 0 0 0 0 0 0 -8.33333 -4.16667 0 0 -4.16667 16.6667
```

3.3 $Matrix\ [H] = [H] + [C]/dT$ (iteracja 0):

Wynik uzyskany w programie:

```
Ostatateczny Matrix H:
[36.81][4.241][0][0][4.241][-4.963][0][0][0][0][0][0][0][0][0][0]
[4.241][66.96][4.241][0][-4.963][5.148][-4.963][0][0][0][0][0][0][0][0][0]
[0][4.241][66.96][4.241][0][-4.963][5.148][-4.963][0][0][0][0][0][0][0][0]
[0][0][4.241][36.81][0][0][-4.963][4.241][0][0][0][0][0][0][0][0]
[4.241][-4.963][0][0][66.96][5.148][0][0][4.241][-4.963][0][0][0][0][0][0]
[-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963][0][0][0][0][0]
[0][-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963][0][0][0][0]
[0][0][-4.963][4.241][0][0][5.148][66.96][0][0][-4.963][4.241][0][0][0][0]
[0][0][0][0][4.241][-4.963][0][0][66.96][5.148][0][0][4.241][-4.963][0][0]
[0][0][0][0][-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963][0]
[0][0][0][0][0][-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963]
[0][0][0][0][0][0][-4.963][4.241][0][0][5.148][66.96][0][0][-4.963][4.241]
[0][0][0][0][0][0][0][0][4.241][-4.963][0][0][36.81][4.241][0][0]
[0][0][0][0][0][0][0][0][-4.963][5.148][-4.963][0][4.241][66.96][4.241][0]
[0][0][0][0][0][0][0][0][0][-4.963][5.148][-4.963][0][4.241][66.96][4.241]
[0][0][0][0][0][0][0][0][0][0][-4.963][4.241][0][0][4.241][36.81]
```

Wynik znajdujący się na stronie przedmiotu:

_____ Interation 0 _____
_____ Matrix ([H]+[C]/dT)_____
36.8148 4.24074 0 0 4.24074 -4.96296 0 0 0 0 0 0 0 0 0 0
4.24074 66.963 4.24074 0 -4.96296 5.14815 -4.96296 0 0 0 0 0 0 0 0 0
0 4.24074 66.963 4.24074 0 -4.96296 5.14815 -4.96296 0 0 0 0 0 0 0 0
0 0 4.24074 36.8148 0 0 -4.96296 4.24074 0 0 0 0 0 0 0 0
4.24074 -4.96296 0 0 66.963 5.14815 0 0 4.24074 -4.96296 0 0 0 0 0 0
-4.96296 5.14815 -4.96296 0 5.14815 120.593 5.14815 0 -4.96296 5.14815 -4.96296 0 0 0 0 0
0 -4.96296 5.14815 -4.96296 0 5.14815 120.593 5.14815 0 -4.96296 5.14815 -4.96296 0 0 0 0
0 0 -4.96296 4.24074 0 0 5.14815 66.963 0 0 -4.96296 4.24074 0 0 0 0
0 0 0 0 4.24074 -4.96296 0 0 66.963 5.14815 0 0 4.24074 -4.96296 0 0
0 0 0 0 -4.96296 5.14815 -4.96296 0 5.14815 120.593 5.14815 0 -4.96296 5.14815 -4.96296 0
0 0 0 0 0 -4.96296 5.14815 -4.96296 0 5.14815 120.593 5.14815 0 -4.96296 5.14815 -4.96296
0 0 0 0 0 0 -4.96296 4.24074 0 0 5.14815 66.963 0 0 -4.96296 4.24074
0 0 0 0 0 0 0 0 4.24074 -4.96296 0 0 36.8148 4.24074 0 0
0 0 0 0 0 0 0 0 -4.96296 5.14815 -4.96296 0 4.24074 66.963 4.24074 0
0 0 0 0 0 0 0 0 0 -4.96296 5.14815 -4.96296 0 4.24074 66.963 4.24074
0 0 0 0 0 0 0 0 0 0 -4.96296 4.24074 0 0 4.24074 36.8148

3.4 Wektor $\{P\} = \{P\} + \left\{\frac{[C]}{dT}\right\} \cdot \{T0\}$ (iteracja 0):

Wynik uzyskany w programie:

```
P_vector:
[1.503e+04]
[1.807e+04]
[1.807e+04]
[1.503e+04]
[1.807e+04]
[1.213e+04]
[1.213e+04]
[1.807e+04]
[1.807e+04]
[1.213e+04]
[1.213e+04]
[1.807e+04]
[1.503e+04]
[1.807e+04]
[1.807e+04]
[1.503e+04]
```

Wynik znajdujący się na stronie przedmiotu:

_____ Vector ([{P}+{[C]/dT}*{T0})_____
15033.3 18066.7 18066.7 15033.3 18066.7 12133.3 12133.3 18066.7 18066.7 12133.3 12133.3 18066.7 15033.3 18066.7 18066.7 15033.350 110.04 365.82

3.5 Wyznaczony wektor nowych temperatur w węzłach siatki MES (iteracja 0):

Wynik uzyskany w programie:

```
Rozwiazanie ukladu rownan:
temp_0 = 365.8
temp_1 = 249
temp_2 = 249
temp_3 = 365.8
temp_4 = 249
temp_5 = 110
temp_6 = 110
temp_7 = 249
temp_8 = 249
temp_9 = 110
temp_10 = 110
temp_11 = 249
temp_12 = 365.8
temp_13 = 249
temp_14 = 249
temp_15 = 365.8
```

Wynik znajdujący się na stronie przedmiotu:

Time[s]	MinTemp[s]	MaxTemp[s]
50	110.038	365.815
100	168.837	502.592
150	242.801	587.373
200	318.615	649.387
250	391.256	700.068
300	459.037	744.063
350	521.586	783.383
400	579.034	818.992
450	631.689	851.431
500	679.908	881.058

3.6 $Matrix\ [H] = [H] + [C]/dT$ (iteracja 1):

Wynik uzyskany w programie:

```
Ostatateczny Matrix H:
[36.81][4.241][0][0][4.241][-4.963][0][0][0][0][0][0][0][0][0][0]
[4.241][66.96][4.241][0][-4.963][5.148][-4.963][0][0][0][0][0][0][0][0][0]
[0][4.241][66.96][4.241][0][-4.963][5.148][-4.963][0][0][0][0][0][0][0][0]
[0][0][4.241][36.81][0][0][-4.963][4.241][0][0][0][0][0][0][0][0]
[4.241][-4.963][0][0][66.96][5.148][0][0][4.241][-4.963][0][0][0][0][0][0]
[-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963][0][0][0][0][0]
[0][-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963][0][0][0][0]
[0][0][-4.963][4.241][0][0][5.148][66.96][0][0][-4.963][4.241][0][0][0][0]
[0][0][0][0][4.241][-4.963][0][0][66.96][5.148][0][0][4.241][-4.963][0][0]
[0][0][0][0][-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963][0]
[0][0][0][0][0][-4.963][5.148][-4.963][0][5.148][120.6][5.148][0][-4.963][5.148][-4.963]
[0][0][0][0][0][0][-4.963][4.241][0][0][5.148][66.96][0][0][-4.963][4.241]
[0][0][0][0][0][0][0][0][4.241][-4.963][0][0][36.81][4.241][0][0]
[0][0][0][0][0][0][0][0][-4.963][5.148][-4.963][0][4.241][66.96][4.241][0]
[0][0][0][0][0][0][0][0][0][-4.963][5.148][-4.963][0][4.241][66.96][4.241]
[0][0][0][0][0][0][0][0][0][-4.963][4.241][0][0][4.241][36.81]
```

Wynik znajdujący się na stronie przedmiotu:

_____ Interation 1 _____
_____H Matrix ([H]+[C]/dT)_____
36.815 4.2407 0 0 4.2407 -4.963 0 0 0 0 0 0 0 0 0 0
4.2407 66.963 4.2407 0 -4.963 5.1481 -4.963 0 0 0 0 0 0 0 0 0
0 4.2407 66.963 4.2407 0 -4.963 5.1481 -4.963 0 0 0 0 0 0 0 0
0 0 4.2407 36.815 0 0 -4.963 4.2407 0 0 0 0 0 0 0 0
4.2407 -4.963 0 0 66.963 5.1481 0 0 4.2407 -4.963 0 0 0 0 0 0
-4.963 5.1481 -4.963 0 5.1481 120.59 5.1481 0 -4.963 5.1481 -4.963 0 0 0 0 0
0 -4.963 5.1481 -4.963 0 5.1481 120.59 5.1481 0 -4.963 5.1481 -4.963 0 0 0 0
0 0 -4.963 4.2407 0 0 5.1481 66.963 0 0 -4.963 4.2407 0 0 0 0
0 0 0 0 4.2407 -4.963 0 0 66.963 5.1481 0 0 4.2407 -4.963 0 0
0 0 0 0 -4.963 5.1481 -4.963 0 5.1481 120.59 5.1481 0 -4.963 5.1481 -4.963 0
0 0 0 0 0 -4.963 5.1481 -4.963 0 5.1481 120.59 5.1481 0 -4.963 5.1481 -4.963
0 0 0 0 0 0 -4.963 4.2407 0 0 5.1481 66.963 0 0 -4.963 4.2407
0 0 0 0 0 0 0 0 4.2407 -4.963 0 0 36.815 4.2407 0 0
0 0 0 0 0 0 0 0 -4.963 5.1481 -4.963 0 4.2407 66.963 4.2407 0
0 0 0 0 0 0 0 0 0 -4.963 5.1481 -4.963 0 4.2407 66.963 4.2407
0 0 0 0 0 0 0 0 0 0 -4.963 4.2407 0 0 4.2407 36.815

3.7 Wektor $\{P\} = \{P\} + \left\{\frac{[C]}{dT}\right\} \cdot \{T0\}$ (iteracja 1):

Wynik uzyskany w programie:

```
P_vector:
[2.066e+04]
[2.555e+04]
[2.555e+04]
[2.066e+04]
[2.555e+04]
[1.89e+04]
[1.89e+04]
[2.555e+04]
[2.555e+04]
[1.89e+04]
[1.89e+04]
[2.555e+04]
[2.066e+04]
[2.555e+04]
[2.555e+04]
[2.066e+04]
```

Wynik znajdujący się na stronie przedmiotu:

_____P_Vector ([{P}+{[C]/dT}*{T0})_____
20660 25552 25552 20660 25552 18897 18897 25552 25552 18897 18897 25552 20660 25552 25552 20660

3.8 Wyznaczony wektor nowych temperatur w węzłach siatki MES (iteracja 1):

Wynik uzyskany w programie:

```
Rozwiazanie ukladu rownan:
temp_0 = 502.6
temp_1 = 353.1
temp_2 = 353.1
temp_3 = 502.6
temp_4 = 353.1
temp_5 = 168.8
temp_6 = 168.8
temp_7 = 353.1
temp_8 = 353.1
temp_9 = 168.8
temp_10 = 168.8
temp_11 = 353.1
temp_12 = 502.6
temp_13 = 353.1
temp_14 = 353.1
temp_15 = 502.6
```

Wynik znajdujący się na stronie przedmiotu:

Time[s]	MinTemp[s]	MaxTemp[s]
50	110.038	365.815
100	168.837	502.592
150	242.801	587.373
200	318.615	649.387
250	391.256	700.068
300	459.037	744.063
350	521.586	783.383
400	579.034	818.992
450	631.689	851.431
500	679.908	881.058

4 Przeprowadzenie testów (dla siatki 31x31):

Ze względu na bardzo duży rozmiar siatki, obliczone macierze oraz wektor wyznaczonych temperatur w węzłach mają tak duże rozmiary, że poniżej przedstawiam tylko fragmenty wyznaczonego wektora temperatur w zerowej oraz pierwszej iteracji.

4.1 Wyznaczony wektor nowych temperatur w węzłach siatki MES (iteracja 0):

Wynik uzyskany w programie:

```
Rozwiazanie ukladu rownan:
temp_0 = 149.6
temp_1 = 129.2
temp_2 = 125.9
temp_3 = 125.3
temp_4 = 125.2
temp_5 = 125.2
temp_6 = 125.2
temp_7 = 125.2
temp_8 = 125.2
temp_9 = 125.2
temp_10 = 125.2
temp_11 = 125.2
temp_12 = 125.2
temp_13 = 125.2
temp_14 = 125.2
temp_15 = 125.2
temp_16 = 125.2
temp_17 = 125.2
temp_18 = 125.2
temp_19 = 125.2
temp_20 = 125.2
temp_21 = 125.2
temp_22 = 125.2
temp_23 = 125.2
temp_24 = 125.2
temp_25 = 125.2
temp_26 = 125.2
temp_27 = 125.3
temp_28 = 125.9
temp_29 = 129.2
temp_30 = 149.6
temp_31 = 129.2
temp_32 = 108.5
temp_33 = 105
temp_34 = 104.4
temp_35 = 104.3
temp_36 = 104.3
temp_37 = 104.3
temp_38 = 104.3
temp_39 = 104.3
temp_40 = 104.3
temp_41 = 104.3
temp_42 = 104.3
temp_43 = 104.3
temp_44 = 104.3
temp_45 = 104.3
temp_46 = 104.3
temp_47 = 104.3
temp_48 = 104.3
temp_49 = 104.3
temp_50 = 104.3
temp_51 = 104.3
temp_52 = 104.3
temp_53 = 104.3
temp_54 = 104.3
temp_55 = 104.3
temp_56 = 104.3
temp_57 = 104.3
temp_58 = 104.4
temp_59 = 105
temp_60 = 108.5
temp_61 = 129.2
temp_62 = 125.9
temp_63 = 105
temp_64 = 101.4
temp_65 = 100.8
temp_66 = 100.7
temp_67 = 100.7
temp_68 = 100.7
temp_69 = 100.7
temp_70 = 100.7
temp_71 = 100.7
temp_72 = 100.7
temp_73 = 100.7
temp_74 = 100.7
temp_75 = 100.7
temp_76 = 100.7
temp_77 = 100.7
temp_78 = 100.7
temp_79 = 100.7
temp_80 = 100.7
temp_81 = 100.7
temp_82 = 100.7
temp_83 = 100.7
temp_84 = 100.7
temp_85 = 100.7
temp_86 = 100.7
temp_87 = 100.7
temp_88 = 100.7
temp_89 = 100.8
temp_90 = 101.4
temp_91 = 105
temp_92 = 125.9
temp_93 = 125.3
temp_94 = 104.4
temp_95 = 100.8
temp_96 = 100.2
temp_97 = 100.1
temp_98 = 100.1
temp_99 = 100.1
temp_100 = 100.1
temp_101 = 100.1
temp_102 = 100.1
temp_103 = 100.1
temp_104 = 100.1
temp_105 = 100.1
temp_106 = 100.1
temp_107 = 100.1
temp_108 = 100.1
temp_109 = 100.1
temp_110 = 100.1
temp_111 = 100.1
temp_112 = 100.1
temp_113 = 100.1
temp_114 = 100.1
temp_115 = 100.1
temp_116 = 100.1
temp_117 = 100.1
temp_118 = 100.1
temp_119 = 100.1
temp_120 = 100.2
temp_121 = 100.8
temp_122 = 104.4
temp_123 = 125.3
temp_124 = 125.2
temp_125 = 104.3
temp_126 = 100.7
temp_127 = 100.1
temp_128 = 100
temp_129 = 100
temp_130 = 100
temp_131 = 100
temp_132 = 100
temp_133 = 100
temp_134 = 100
temp_135 = 100
temp_136 = 100
temp_137 = 100
temp_138 = 100
temp_139 = 100
temp_140 = 100
temp_141 = 100
temp_142 = 100
temp_143 = 100
temp_144 = 100
temp_145 = 100
temp_146 = 100
temp_147 = 100
temp_148 = 100
temp_149 = 100
temp_150 = 100
temp_151 = 100.1
temp_152 = 100.7
temp_153 = 104.3
temp_154 = 125.2
temp_155 = 125.2
temp_156 = 104.3
temp_157 = 100.7
temp_158 = 100.1
temp_159 = 100
temp_160 = 100
temp_161 = 100
temp_162 = 100
temp_163 = 100
temp_164 = 100
temp_165 = 100
temp_166 = 100
temp_167 = 100
temp_168 = 100
temp_169 = 100
temp_170 = 100
temp_171 = 100
temp_172 = 100
temp_173 = 100
temp_174 = 100
temp_175 = 100
temp_176 = 100
temp_177 = 100
temp_178 = 100
temp_179 = 100
temp_180 = 100
temp_181 = 100
temp_182 = 100.1
temp_183 = 100.7
temp_184 = 104.3
temp_185 = 125.2
temp_186 = 125.2
temp_187 = 104.3
temp_188 = 100.7
temp_189 = 100.1
temp_190 = 100
temp_191 = 100
temp_192 = 100
temp_193 = 100
temp_194 = 100
temp_195 = 100
temp_196 = 100
temp_912 = 104.3
temp_913 = 104.3
temp_914 = 104.3
temp_915 = 104.3
temp_916 = 104.3
temp_917 = 104.3
temp_918 = 104.3
temp_919 = 104.3
temp_920 = 104.3
temp_921 = 104.3
temp_922 = 104.3
temp_923 = 104.3
temp_924 = 104.3
temp_925 = 104.3
temp_926 = 104.4
temp_927 = 105
temp_928 = 108.5
temp_929 = 129.2
temp_930 = 149.6
temp_931 = 129.2
temp_932 = 125.9
temp_933 = 125.3
temp_934 = 125.2
temp_935 = 125.2
temp_936 = 125.2
temp_937 = 125.2
temp_938 = 125.2
temp_939 = 125.2
temp_940 = 125.2
temp_941 = 125.2
temp_942 = 125.2
temp_943 = 125.2
temp_944 = 125.2
temp_945 = 125.2
temp_946 = 125.2
temp_947 = 125.2
temp_948 = 125.2
temp_949 = 125.2
temp_950 = 125.2
temp_951 = 125.2
temp_952 = 125.2
temp_953 = 125.2
temp_954 = 125.2
temp_955 = 125.2
temp_956 = 125.2
temp_957 = 125.3
temp_958 = 125.9
temp_959 = 129.2
temp_960 = 149.6
```

Wynik znajdujący się na stronie przedmiotu:

Time[s]	MinTemp	MaxTemp
1	100	149.56
2	100	177.44
3	100	197.27
4	100	213.15
5	100	226.68
6	100	238.61
7	100	249.35
8	100	259.17
9	100	268.24
10	100	276.7
11	100	284.64
12	100	292.13
13	100	299.24
14	100.01	306
15	100.01	312.45
16	100.01	318.63
17	100.02	324.56
18	100.03	330.27
19	100.05	335.77
20	100.06	341.08

4.2 Wyznaczony wektor nowych temperatur w węzłach siatki MES (iteracja 1):

Wynik uzyskany w programie:

```
Rozwiazanie ukladu rownan:
temp_0 = 177.4
temp_1 = 150.8
temp_2 = 142.2
temp_3 = 140.1
temp_4 = 139.7
temp_5 = 139.6
temp_6 = 139.5
temp_7 = 139.5
temp_8 = 139.5
temp_9 = 139.5
temp_10 = 139.5
temp_11 = 139.5
temp_12 = 139.5
temp_13 = 139.5
temp_14 = 139.5
temp_15 = 139.5
temp_16 = 139.5
temp_17 = 139.5
temp_18 = 139.5
temp_19 = 139.5
temp_20 = 139.5
temp_21 = 139.5
temp_22 = 139.5
temp_23 = 139.5
temp_24 = 139.5
temp_25 = 139.6
temp_26 = 139.7
temp_27 = 140.1
temp_28 = 142.2
temp_29 = 150.8
temp_30 = 177.4
temp_31 = 150.8
temp_32 = 123.3
temp_33 = 114.6
temp_34 = 112.4
temp_35 = 111.9
temp_36 = 111.8
temp_37 = 111.8
temp_38 = 111.8
temp_39 = 111.8
temp_40 = 111.8
temp_41 = 111.8
temp_42 = 111.8
temp_43 = 111.8
temp_44 = 111.8
temp_45 = 111.8
temp_46 = 111.8
temp_47 = 111.8
temp_48 = 111.8
temp_49 = 111.8
temp_50 = 111.8
temp_51 = 111.8
temp_52 = 111.8
temp_53 = 111.8
temp_54 = 111.8
temp_55 = 111.8
temp_56 = 111.8
temp_57 = 111.9
temp_58 = 112.4
temp_59 = 114.6
temp_60 = 123.3
temp_61 = 150.8
temp_62 = 142.2
temp_63 = 114.6
temp_64 = 105.7
temp_65 = 103.5
temp_66 = 103
temp_67 = 102.9
temp_68 = 102.9
temp_69 = 102.9
temp_70 = 102.9
temp_71 = 102.9
temp_72 = 102.9
temp_73 = 102.9
temp_74 = 102.9
temp_75 = 102.9
temp_76 = 102.9
temp_77 = 102.9
temp_78 = 102.9
temp_79 = 102.9
temp_80 = 102.9
temp_81 = 102.9
temp_82 = 102.9
temp_83 = 102.9
temp_84 = 102.9
temp_85 = 102.9
temp_86 = 102.9
temp_87 = 102.9
temp_88 = 103
temp_89 = 103.5
temp_90 = 105.7
temp_91 = 114.6
temp_92 = 142.2
temp_93 = 140.1
temp_94 = 112.4
temp_95 = 103.5
temp_96 = 101.3
temp_97 = 100.8
temp_98 = 100.7
temp_99 = 100.6
temp_100 = 100.6
temp_101 = 100.6
temp_102 = 100.6
temp_103 = 100.6
temp_104 = 100.6
temp_105 = 100.6
temp_106 = 100.6
temp_107 = 100.6
temp_108 = 100.6
temp_109 = 100.6
temp_110 = 100.6
temp_111 = 100.6
temp_112 = 100.6
temp_113 = 100.6
temp_114 = 100.6
temp_115 = 100.6
temp_116 = 100.6
temp_117 = 100.6
temp_118 = 100.7
temp_119 = 100.8
temp_120 = 101.3
temp_121 = 103.5
temp_122 = 112.4
temp_123 = 140.1
temp_124 = 139.7
temp_125 = 111.9
temp_126 = 103
temp_127 = 100.8
temp_128 = 100.3
temp_129 = 100.2
temp_130 = 100.1
temp_131 = 100.1
temp_132 = 100.1
temp_133 = 100.1
temp_134 = 100.1
temp_135 = 100.1
temp_136 = 100.1
temp_137 = 100.1
temp_138 = 100.1
temp_139 = 100.1
temp_140 = 100.1
temp_141 = 100.1
temp_142 = 100.1
temp_143 = 100.1
temp_144 = 100.1
temp_145 = 100.1
temp_146 = 100.1
temp_147 = 100.1
temp_148 = 100.1
temp_149 = 100.2
temp_150 = 100.3
temp_151 = 100.8
temp_152 = 103
temp_153 = 111.9
temp_154 = 139.7
temp_155 = 139.6
temp_156 = 111.8
temp_157 = 102.9
temp_158 = 100.7
temp_159 = 100.2
temp_160 = 100.1
temp_161 = 100
temp_162 = 100
temp_163 = 100
temp_164 = 100
temp_165 = 100
temp_166 = 100
temp_167 = 100
temp_168 = 100
temp_169 = 100
temp_170 = 100
temp_171 = 100
temp_172 = 100
temp_173 = 100
temp_174 = 100
temp_175 = 100
temp_176 = 100
temp_177 = 100
temp_178 = 100
temp_179 = 100
temp_180 = 100.1
temp_181 = 100.2
temp_182 = 100.7
temp_183 = 102.9
temp_184 = 111.8
temp_185 = 139.6
temp_186 = 139.5
temp_187 = 111.8
temp_188 = 102.9
temp_189 = 100.6
temp_190 = 100.1
temp_191 = 100
temp_192 = 100
temp_193 = 100
temp_194 = 100
temp_195 = 100
temp_196 = 100
temp_197 = 100
temp_912 = 111.8
temp_913 = 111.8
temp_914 = 111.8
temp_915 = 111.8
temp_916 = 111.8
temp_917 = 111.8
temp_918 = 111.8
temp_919 = 111.8
temp_920 = 111.8
temp_921 = 111.8
temp_922 = 111.8
temp_923 = 111.8
temp_924 = 111.8
temp_925 = 111.9
temp_926 = 112.4
temp_927 = 114.6
temp_928 = 123.3
temp_929 = 150.8
temp_930 = 177.4
temp_931 = 150.8
temp_932 = 142.2
temp_933 = 140.1
temp_934 = 139.7
temp_935 = 139.6
temp_936 = 139.5
temp_937 = 139.5
temp_938 = 139.5
temp_939 = 139.5
temp_940 = 139.5
temp_941 = 139.5
temp_942 = 139.5
temp_943 = 139.5
temp_944 = 139.5
temp_945 = 139.5
temp_946 = 139.5
temp_947 = 139.5
temp_948 = 139.5
temp_949 = 139.5
temp_950 = 139.5
temp_951 = 139.5
temp_952 = 139.5
temp_953 = 139.5
temp_954 = 139.5
temp_955 = 139.6
temp_956 = 139.7
temp_957 = 140.1
temp_958 = 142.2
temp_959 = 150.8
temp_960 = 177.4
```

Wynik znajdujący się na stronie przedmiotu:

Time[s]	MinTemp	MaxTemp
1	100	149.56
2	100	177.44
3	100	197.27
4	100	213.15
5	100	226.68
6	100	238.61
7	100	249.35
8	100	259.17
9	100	268.24
10	100	276.7
11	100	284.64
12	100	292.13
13	100	299.24
14	100.01	306
15	100.01	312.45
16	100.01	318.63
17	100.02	324.56
18	100.03	330.27
19	100.05	335.77
20	100.06	341.08

5 Wnioski :

Program został uruchomiony dwukrotnie. Pierwszy raz został uruchomiony dla siatki o rozmiarze 4x4, a następnie drugi raz dla siatki o rozmiarze 31x31. Analizując uzyskane wyniki z programu i porównując je z wynikami znajdującymi się na stronie można stwierdzić, że wyliczone macierze, które są potrzebne do późniejszego wyznaczenia wektora temperatur w węzłach są takie same. Porównując także wyniki uzyskane w eliminacji Gaussa czyli wektor nowych temperatur w węzłach siatki są takie same jak te znajdujące się na stronie przedmiotu. Dla siatki o rozmiarze 4x4 w zerowej iteracji czyli po 50 sekundach (tyle wynosi krok czasowy) minimalna temperatura, jaka znajduję się w węzłach siatki to w przybliżeniu 110°C, natomiast maksymalna temperatura, jaka znajduję się w węzłach siatki to w przybliżeniu 365,8°C. Dla następnej iteracji minimalna temperatura wynosi w przybliżeniu 168,8°C, a maksymalna temperatura wynosi 502,6°C.
Dla siatki o rozmiarze 31x31 w zerowej iteracji minimalna temperatura, jaka znajduję się w węzłach siatki to 100°C, a maksymalna temperatura wynosi w przybliżeniu 149,6°C. W następnej iteracji minimalna wyznaczona temperatura wynosi nadal 100°C, a maksymalna wzrosła do 177,4°C. Analizując jak rozkładają się wyliczone temperatury można zauważyć, że najwyższe temperatury znajdują się w węzłach w rogach siatki, a najniższe temperatury znajdują się w węzłach w środku siatki. Dzieje się tak ponieważ węzły w środku siatki nie posiadają warunku brzegowego czyli nie biorą udziału w wymianie ciepła, a więc temperatura w tych węzłach nie zmienia się i pozostała taka sama jak temperatura początkowa. W następnych iteracjach temperatura w węzłach biorących udział w wymianie ciepła wzrasta, ponieważ wcześniej wyliczone temperatury stały się w tej iteracji wartościami temperatury początkowej.