به نام خدا

مدلسازی جابجایی طبیعی در محفظه مستطیلی به روش شبکه بولتزمن

LBMiran.blog.ir

فهرست مطالب

۱- چکیده ۲
۲- راهنمای کاربری ۴
نمونه اجرا ۶
نتایج نمونه اجرا شده و مقایسه با مقالات معتبر ۱۱
۳- متن اصلی برنامه Main ۱۵
۴- سابروتین Pre_Solution ۱۹
۵- سابروتین Fluid_Flow ۲۱
۶- سابروتین Temperature_Distribution ۲۸
۷- سابروتین Residual ۳۵
۸- سابروتین Output ۳۷
۹- سابروتین After_Solution ۴۰
۱۰-منابع و مراجع ۴۱

۱- چکیده

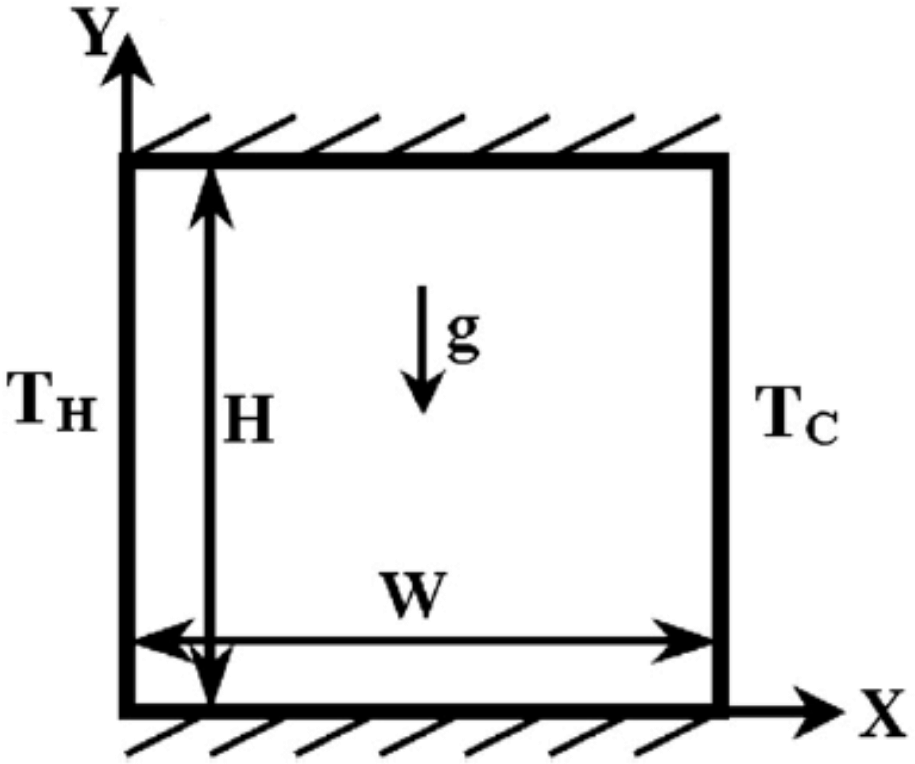
در این برنامه برای مدلسازی جابجایی طبیعی در محفظه‌ای مستطیلی از روش شبکه بولتزمن با مدل سرعتی D2Q9 برای توزیع دما و جریان سیال استفاده شده است. در هر چهار دیواره محفظه از شرط مرزی بازگشت به عقب استفاده شده است. گفتنی است که تمامی پارامترهای ورودی و خروجی برنامه بی-بعد هستند.

کلمات کلیدی: جابجایی طبیعی، روش شبکه بولتزمن

۲- راهنمای کاربری

این نرم افزار توسط وبلاگ LBMiran.blog.ir نگارش شده است. نسخه حاضر Version:1.0 می‌باشد. لازمه اجرای برنامه آشنایی با نحوه وارد کردن اطلاعات ورودی به برنامه است لذا در این بخش به طور خلاصه به این موارد اشاره خواهد شد. گفتنی است که این قسمت مخصوص کاربرانی است که فقط می‌خواهند نرم‌افزار را اجرا نموده و استفاده نمایند. لذا هیچ اشاره‌ای به محتوای برنامه اعم از سابروتین‌ها و روش حل نشده است.

هندسه مورد بررسی کد حاضر در شکل ۲-۱ نشان داده‌شده‌است. دیواره عمودی سمت چپ با دمای T_H و دیواره سمت راست دارای دمای T_C بوده و دیواره‌های افقی عایق هستند. فرض بر این است که سیال نیوتنی و جریان سیال آرام و تراکم‌ناپذیر باشد. همچنین فرض بر این است که دو فاز جامد و مایع در تعادل حرارتی باشند و نانوذرات سرعتی برابر با جریان داشته باشند.



شکل ۲-۱: هندسه مورد بررسی کد حاضر

برنامه حاضر به نحوی تهیه شده‌است که برای اجرای آن و وارد کردن ورودی نیازی به ورود به متن اصلی برنامه نیست و تمام ورودی‌ها در فایل Input_Parameters با فرمت Input وارد می‌شوند. در ابتدای این بخش مقادیر فیزیکی ورودی فایل‌های مذکور معرفی می‌شوند. این مقادیر در جدول۲-۱ و جدول۲-۲ تعریف شده است.

جدول۲-۱: تعریف متغیرهای فیزیکی ورودی فایل Input_Parameters.input

متغیر داخل فایل	تعریف متغیر
Prandtl Number(Pr)	عدد پرانتل
Rayleigh Number(Ra)	عدد رایلی
Temperature Of Hot Boundary (Th)	دمای دیوار گرم(برای بی‌بعد بودن باید برابر یک باشد)
Temperature Of Cold Boundary (Tc)	دمای دیوار سرد(برای بی‌بعد بودن باید برابر صفر باشد)
Initial Temperature(T_0)	دمای اولیه نانوسیال(برای بی‌بعد بودن باید نسبتی از دمای دیواره گرم باشد)
Relaxation Time Of Temperatuer(τ_T)	زمان آرامش دما(این پارامتر بهتر است تاحد امکان به یک نزدیک باشد به این صورت که در پرانتل‌های بالاتر از یک کمی کمتر و برای پرانتل‌های کوچکتر از یک کمی بزرگتر از یک باشد.)

پارامترهای هندسی که شامل تعداد شبکه در راستای افقی n و تعداد شبکه در راستای عمودی m هستند، در حین اجرای برنامه در کاربر خواسته می‌شوند. نکته قابل توجه این است که این مقادیر هرچه بیشتر باشند دقت پاسخ‌های مسئله بیشتر می‌شود ولی به همان میزان زمان همگرایی و هزینه محاسباتی افزایش می‌یابد. در ضمن نسبت طول به عرض محفظه باید در وارد کردن پارامترهای n و m لحاظ شود.

گفتنی است که در حقیقت پارامترهای بیشتری مثل معیار همگرایی و ... هستند که می‌توانند بر پاسخ-ها اثر بگذارند ولی با توجه به اینکه وارد کردن آنها نیاز به تجربه کافی در روش شبکه بولتزمن دارد، برای

اجتناب از پیچیدگی کار با کد مناسب‌ترین آنها در داخل کد ثابت شده و در خود برنامه توضیح داده خواهد شد.

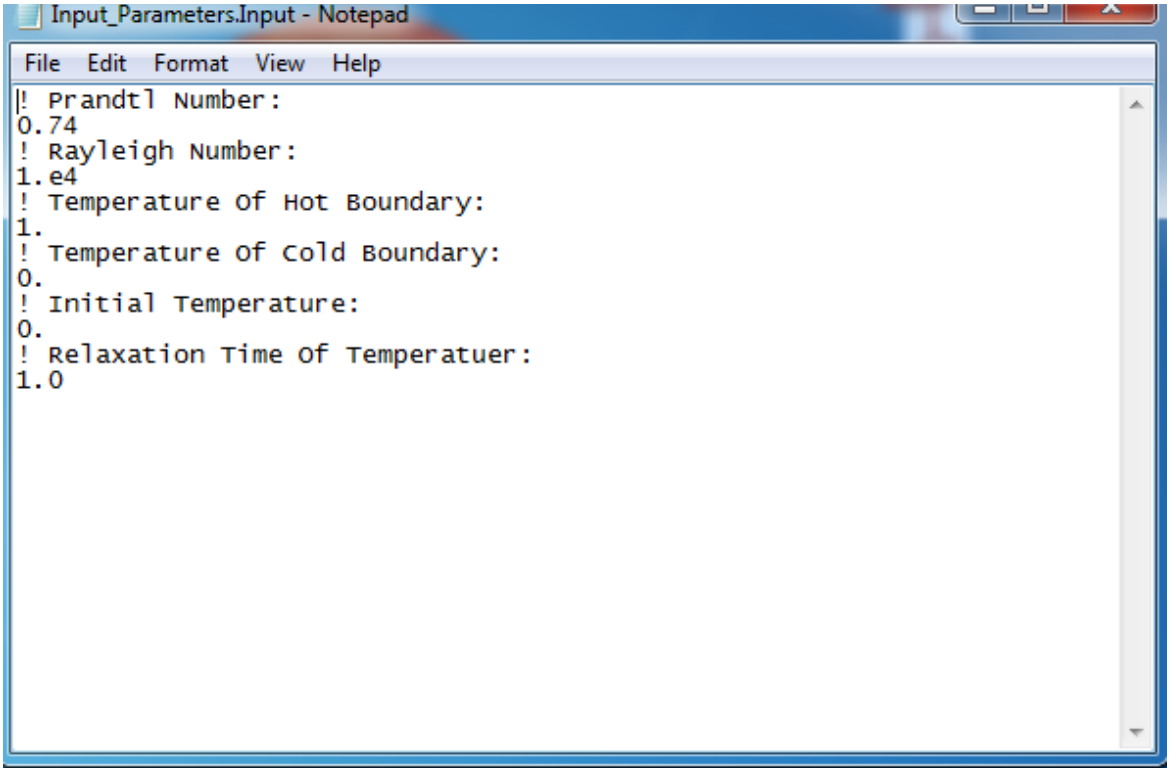
توضیحات فوق برای اجرای برنامه کافی است. پس از اجرا و همگرایی برنامه چندین فایل خروجی تشکیل می‌گردد. این فایل‌ها شامل کانتورهای دما، فشار، سرعت‌ها و خطوط جریان و همینطور نمودار سرعت و دما خطوط مرکزی محفظه هستند که در نمونه اجرا نمایش داده خواهد شد.

نمونه اجرا

در این بخش به منظور تمرین عملی موارد بالا به یک مثال اشاره کرده و جهت اعتبارسنجی با مقالات معتبر مقایسه خواهیم نمود.

گام اول:

باید فایل Input_Parameters.input را با نرم‌افزار Notepad باز کرده و پارامترهای فیزیکی را وارد نمود.

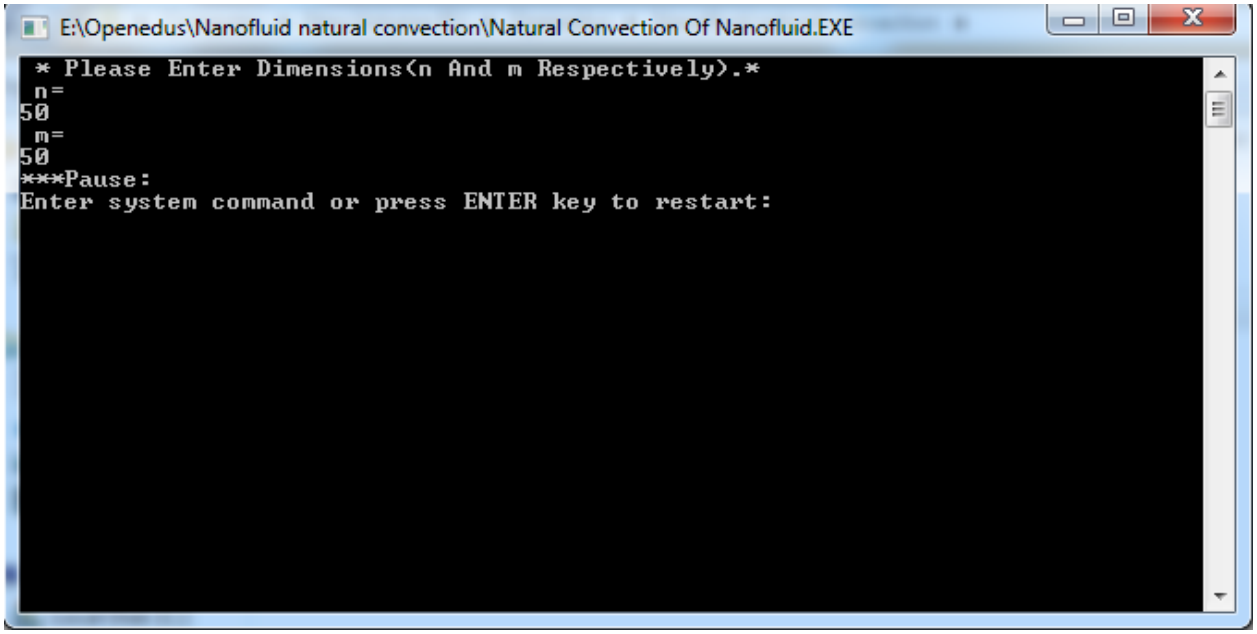


شکل ۲-۱: وارد کردن پارامترهای فیزیکی

در اینجا برای امکان اعتبارسنجی پرانتل 6.2 و رایلی 6.2×10^4 و غلظت نانوذرات را 0.1 قرار داده شده است.

گام دوم:

در مرحله آخر روی فایل Natural Convection.EXE راست کلیک کرده و گزینه Run as administrator را انتخاب می‌نماییم. سپس نرم‌افزار اجرا شده و تعداد شبکه را می‌خواهد، پس از وارد کردن به ترتیب هرکدام از مقادیر n و m، کلید Enter را فشار می‌دهیم.



شکل ۲-۲: وارد کردن ابعاد شبکه

در اینجا ابعاد 50×50 انتخاب شده‌است.

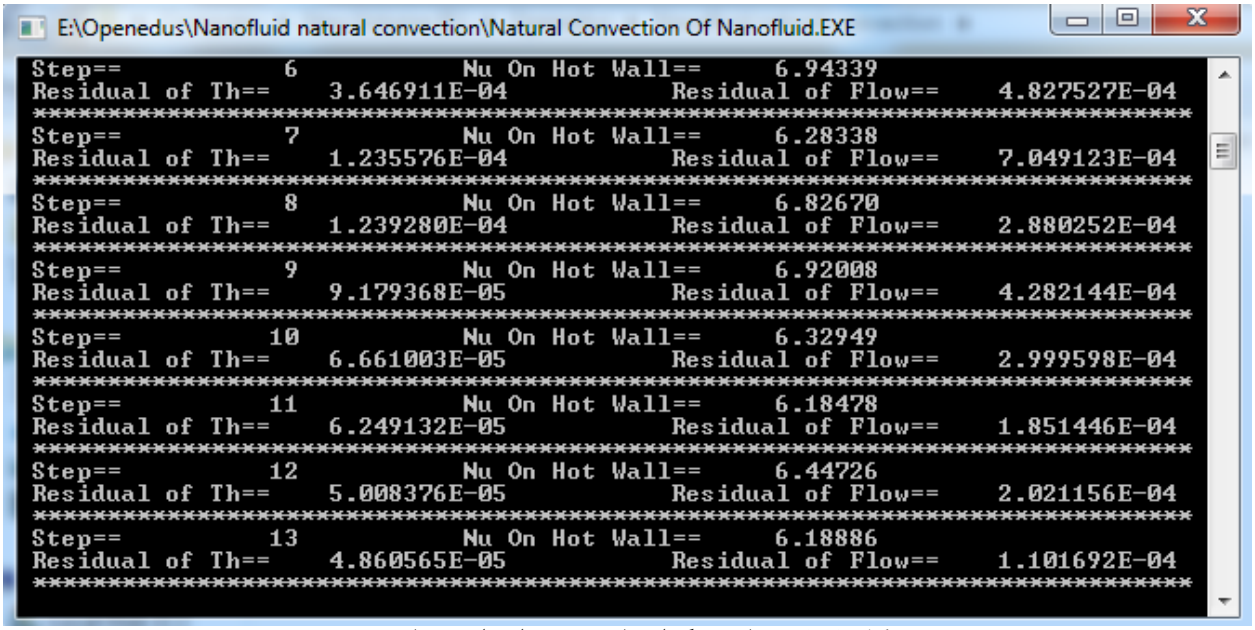
گام چهارم:

در این مرحله پس از فشار دوباره کلید Enter، فرایند اجرای برنامه شروع شده و گزارشاتی شامل از روند حل نمایش داده می‌شود که توضیحات آن در جدول ۲-۵ آورده شده است.

جدول ۲-۲: توضیح گزارشات حین اجرای برنامه

متغیر داخل برنامه	تعریف متغیر
Step	شمارنده گام زمانی است
Nu on Hot Wall	عدد ناسلت روی دیواره گرم است
Residual Of Th	معیار همگرایی توزیع دما است و کوچک شدن آن نشانه پیشروی به سمت همگرایی است.
Residual Of Flow	معیار همگرایی جریان است و کوچک شدن آن نشانه پیشروی به سمت همگرایی است.

نمایش گزارشات اجرای برنامه در شکل زیر نشان داده شده است.



شکل ۲-۳: نمایش گزارشات حین اجرای برنامه

گام پنجم:

پس از ارضا شدن شرایط همگرایی، پنجره اجرا خود به خود بسته شده و یا در صورت اجرا برنامه در نرم‌افزار Fortran گزارشی به صورت زیر به نمایش در خواهد آمد:

Solution Has Finished!

Thanks For Using This Code.

اجرا برنامه توسط یک رایانه (Laptop) با پردازشگر Intel(R) Core(TM) i5-2430M CPU @ 2.40GHz انجام شده و در حدود ۶۵ ثانیه به طول انجامید.

گام ششم:

در نهایت فایل‌های نتایج آماده خواهد شد که در جدول ۲-۶ توضیحات آنها بیان شده است:

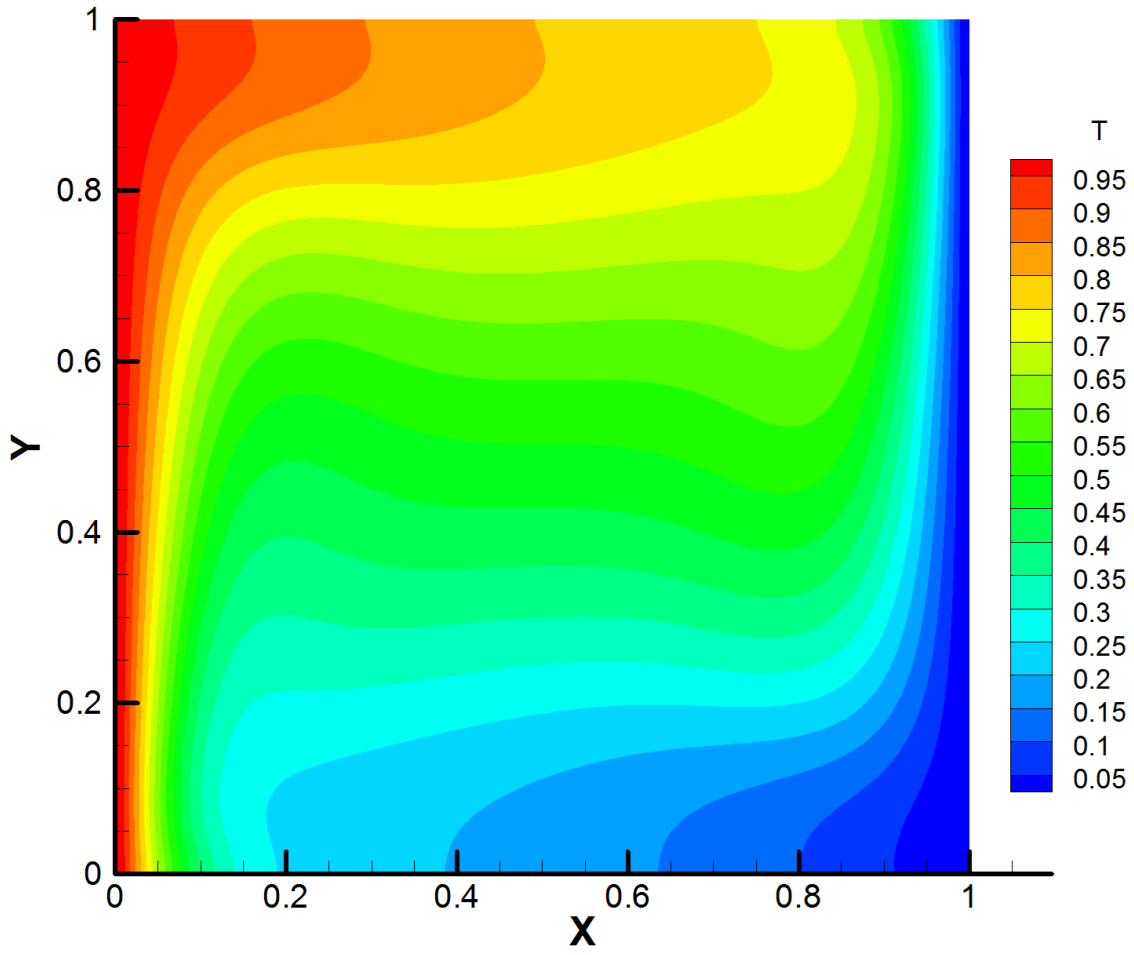
جدول ۲-۳: توضیح فایل‌های خروجی برنامه

متغیر داخل برنامه	تعریف متغیر
Result.dat	این فایل شامل کانتورهای سرعت عمودی، سرعت افقی، فشار، دما و خطوط جریان است.
TEMPERATURE_MidLine.dat	این فایل دما را در امتداد خط مرکزی افقی نمایش می‌دهد.
U_Velocity_MidLine.dat	این فایل سرعت عمودی را در امتداد خط مرکزی افقی نمایش می‌دهد.
V_Velocity_Vertical_MidLine.dat	این فایل سرعت افقی را در امتداد خط مرکزی عمودی نمایش می‌دهد.
Nu.dat	این فایل عدد ناسلت نهایی روی دیواره گرم را نمایش می‌دهد.

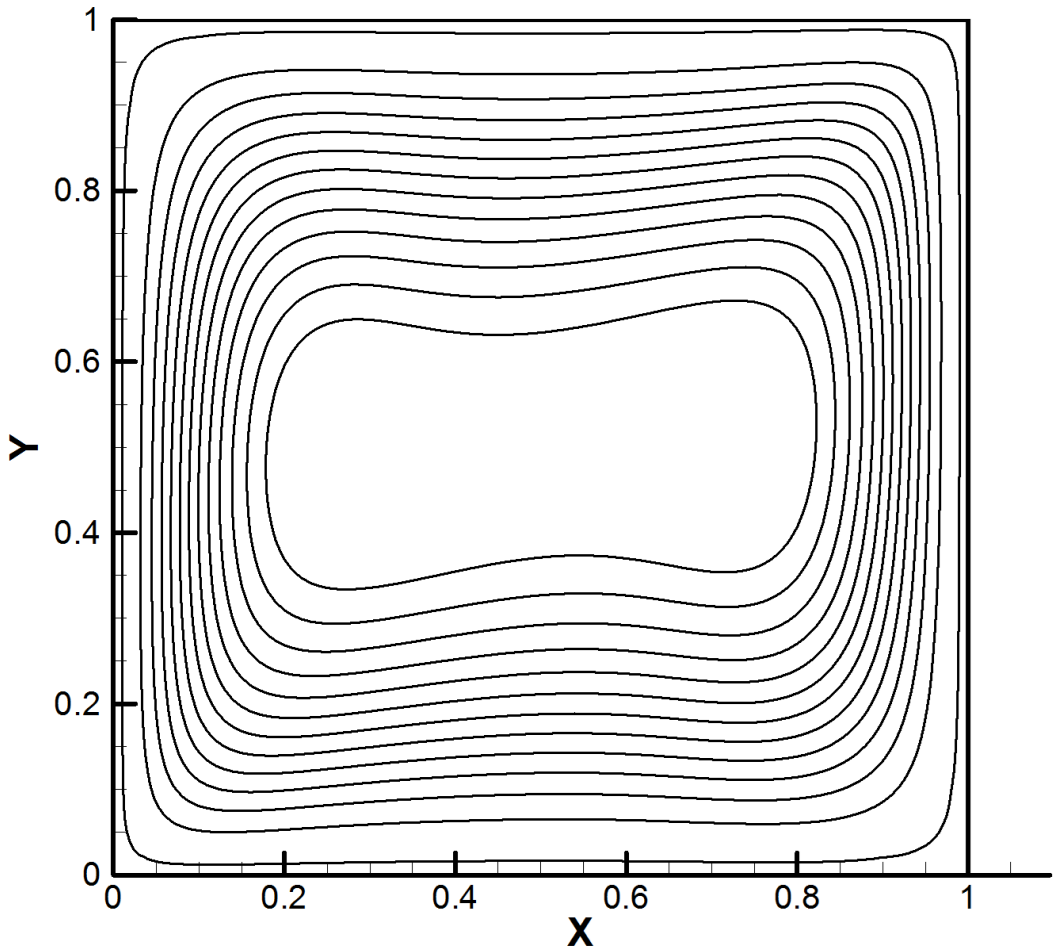
در انتها کانتورها و نمودارهای نمونه اجرا شده نشان داده شده و نمودارهای سرعت و دما با مقاله‌ای معتبر مقایسه شده است.

نتایج نمونه اجرا شده و مقایسه با مقالات معتبر

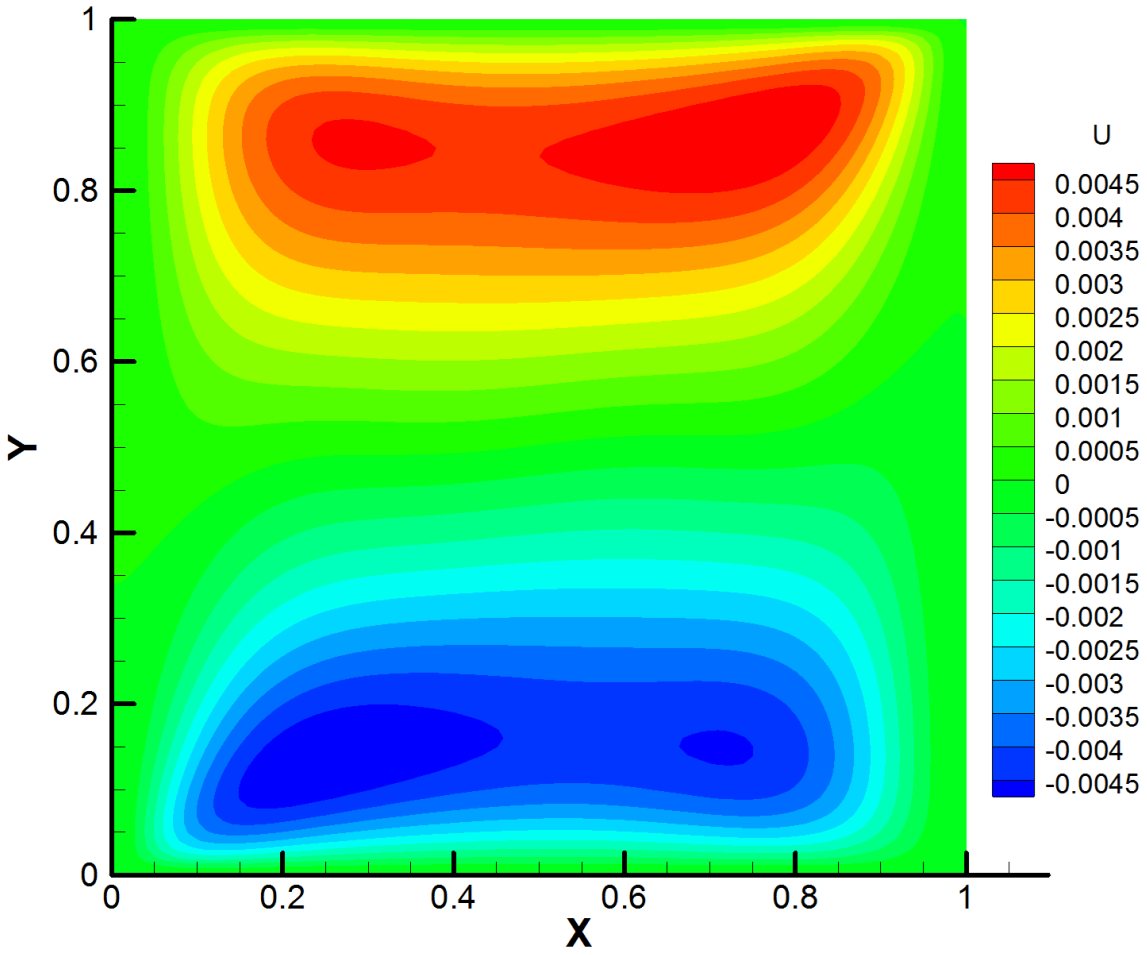
در این قسمت کانتورها و نمودارهای نمونه اجرا شده نشان داده شده و نمودارهای سرعت و دما با مقاله‌ای معتبر مقایسه شده است.



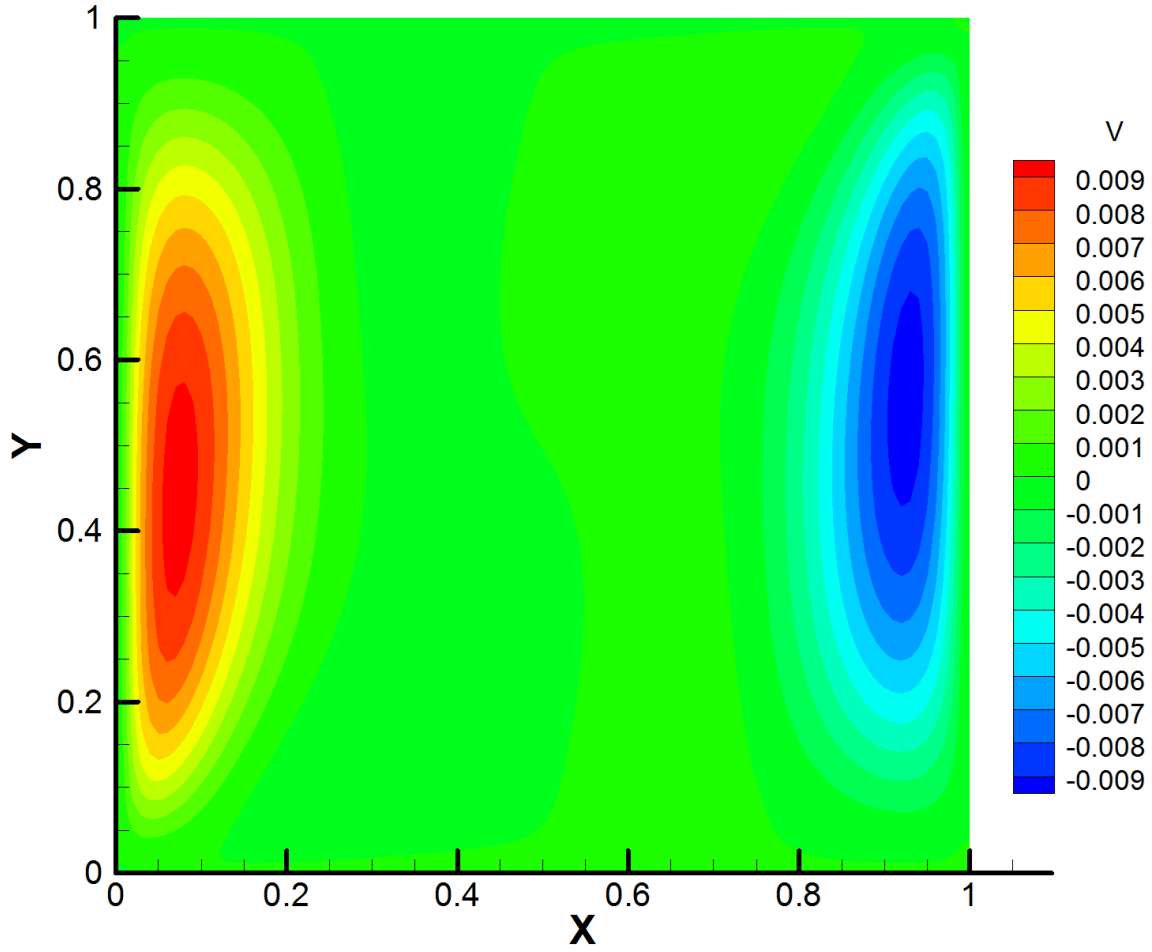
شکل ۲-۴: کانتور دما نمونه اجرا شده $\left(\mathrm{Ra}=6.2\times10^{6},\mathrm{Pr}=6.2\right)$



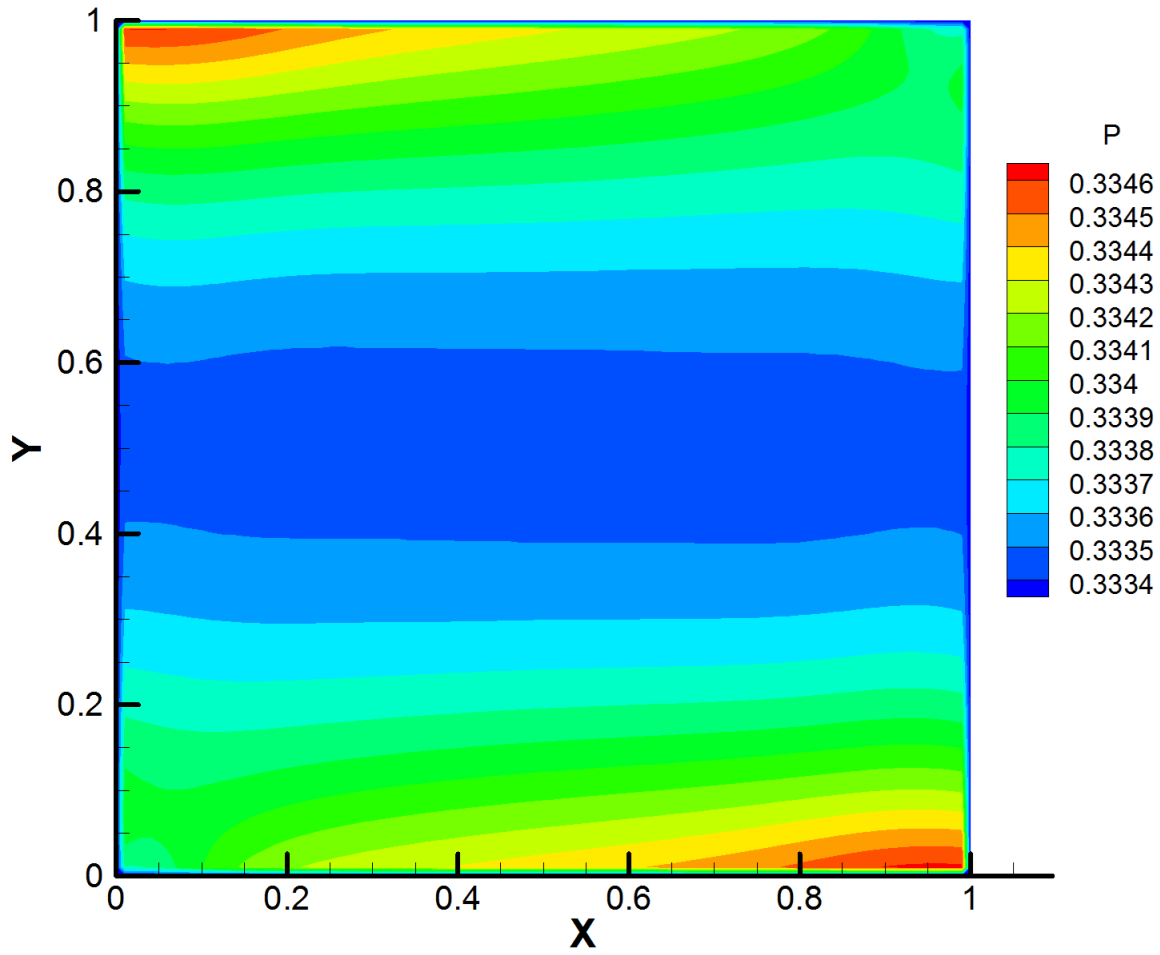
شکل ۲-۵: خطوط جریان نمونه اجرا شده $\left(\mathrm{Ra}=6.2\times10^{6},\mathrm{Pr}=6.2\right)$



شکل ۲-۶: کانتور سرعت افقی نمونه اجرا شده $\left(\mathrm{Ra}=6.2\times10^{6},\mathrm{Pr}=6.2\right)$



شکل ۲-۷: کانتور سرعت قائم نمونه اجرا شده $\left(\mathrm{Ra}=6.2\times10^{6},\mathrm{Pr}=6.2\right)$



شکل ۲-۸: کانتور فشار نمونه اجرا شده $\left(\mathrm{Ra}=6.2\times10^{6},\mathrm{Pr}=6.2\right)$

۳- متن اصلی برنامه Main

متن اصلی برنامه در واقع دارای دو وظیفه مهم به شرح زیر است:

۱- گرفتن ابعاد شبکه از کاربر (n,m) و تعیین ابعاد متغییرهای ماتریسی(Allucate) بکار رفته در تمام سابروتین‌ها

۲- کنترل فرایند حل از طریق فراخوانی(Call) سابروتین‌ها، سنجش شرایط همگرایی و نمایش گزارشاتی از شرایط محاسبات

در این قسمت به تعریف متغیرهای کنترل فرایند حل اکتفا نموده و سایر متغییرها در سابروتین مربوطه به تفصیل بررسی خواهند شد. پس از آن به صورت اجمالی کارایی و نحوه عملکرد سابروتین‌ها را توضیح داده خواهد شد.

در جدول ۳-۱ متغیرهای کنترل فرایند حل معرفی شده‌اند.

جدول ۳-۱: تعریف متغیرهای کنترل فرایند حل

متغیر داخل برنامه	تعریف متغیر	نوع متغیر
iTStep	شمارنده گام زمانی	عدد طبیعی
mstep	حداکثر گام زمانی است که در کد حاضر در ۱۰۰۰۰۰۰ ثابت شده است و در موارد خاص مثلاً در استفاده از شبکه‌های بسیار ریز یا پدیده‌های با فیزیک کند می‌توان آن را افزایش داد و هیچگاه نیازی به کاهش آن نیست.	عدد طبیعی
StepCR	گام زمانی بحرانی است و در واقع پس از هر تعداد گام زمانی بحرانی سابروتین خروجی فراخوانی می‌شود. این مقدار بحرانی هرچه بزرگتر باشد، زمان کمتری برای محاسبات خروجی و نمایش گزارشات حین اجرای برنامه صرف می‌شود. این مقدار در اینجا برابر ۵۰ است.	عدد طبیعی
ResT	باقیمانده دما(Residual) است که معیاری برای سنجش همرایی حوزه دما می‌باشد.	عدد حقیقی
ResF	باقیمانده جریان(Residual) است که معیاری برای همگرایی حوزه جریان است.	عدد حقیقی
ResConvergT	حداقل باقیمانده دما برای همگرایی	عدد حقیقی
ResConvergF	حداقل باقیمانده جریان برای همگرایی	عدد حقیقی

تمام سابروتین‌های کد حاضر که در برنامه اصلی فراخوانی می‌شوند در جدول ۳-۲ به اختصار توضیح داده شده اند.

جدول ۳۳-۲: سابروتین‌های فراخوانی شده در متن اصلی برنامه

سابروتین	شرح وظایف
Pre_Solution	این سابروتین پارامترهای فیزیکی ورودی را گرفته و پس از تحلیل آنها پارانترهای محاسباتی را مقدار دهی اولیه می‌کند. ضمناً روابط نانو در این سابروتین اعمال می‌شود.
Fluid_Flow	این سابروتین حوزه جریان را حل نموده و سرعت‌های افقی و عمودی را بدست می‌آورد.
Temperature_Distribution	این سابروتین حوزه دما را حل نموده و دما را محاسبه می‌کند.
Output	این سابروتین پارامترهای خروجی را محاسبه کرده و در فایل‌های مربوطه چاپ می‌کند.
Residual	این سابروتین باقیمانده‌های دما و جریان را جهت سنجش همگرایی حل محاسبه می‌کند.
After_Solution	در این سابروتین پس از همگرایی حل، بار دیگر سابروتین Output فراخوانی شده و گزارشات اتمام فرایند حل مسئله به نمایش در می‌آیند.

ترتیب فراخوانی توابع و کنترل فرایند حل به شرح زیر است:

```
Call Pre_Solution

Do iTStep=1,mstep
```

```
        Call Fluid_Flow
        Call Temperature_Distribution
        If(iTStep-StepCR*(iTStep/StepCR)<0.01)Then
            Call Output
            Print*,'Step==',iTStep/StepCR,' Nu On Hot Wall==',Nuavg
            Print*,'Residual of Th==',ResT,' Residual of Flow==',ResF
        End If
        Call Residual
        If(ResT<ResConvergT.Or.ResF<ResConvergF)Exit
End Do
Call After_Solution
```

در خطوط برنامه بالا سه قسمت که زیر آنها خط کشیده شده است، نیاز به توضیح بیشتر دارد که در جدول زیر به آن پرداخته شده است.

جدول ۳-۴: توضیح خطوط نکته‌دار برنامه

خط برنامه	توضیح
If(iTStep-StepCR*(iTStep/StepCR)<0.01) Then	با توجه به اینکه iTSte و StepCR هردو عدد طبیعی هستند، عبارت iTStep/StepCR نیز یک عدد طبیعی است و در نتیجه در صورت تقسیم‌پذیر بودن گام زمانی بر گام بحرانی عبارت برابر صفر و کوچکتر از هر عدد گویای کوچکتر از یک(۰/۰۱) می‌باشد.
Step==',iTStep/StepCR'	Step چاپ شده در این قسمت در واقع گام زمانی نیست و منظور تعداد دفعات فراخوانی سابروتین Output است.
If(ResT<ResConvergT.Or.ResF< ResConvergF)Exit	این خط برنامه در واقع شرط همگرایی برنامه را می‌سنجد و عبارت Or طبق نظر مراجع استفاده شده و برتری آن از عبارت And در اینجا از تجربه نیز اثبات شده است.

در بخش‌های آتی باقی سابروتین‌ها شرح داده می‌شوند.

۴- سابروتین Pre_Solution

سابروتین Pre_Solution در واقع وظیفه آماده کردن پیش نیازهای شروع فرایند حل مسئله را دارند. این وظایف به شرح زیر هستند:

۱- خواندن تمام ورودی‌های فیزیکی مسئله

۲- تحلیل پارامترها و بدست آوردن سایر پارامترها اعم از پارامترهای فیزیکی و عددی مربوط به روش شبکه بولتزمن از پارامترهای ورودی

۳- مقداردهی اولیه پارامترهای ماتریسی

وظیفه اول از لحاظ کد نویسی پیچیدگی خاصی ندارد و از توضیح آن برای پرهیز از زیاده‌گویی اجتناب می‌شود.

وظیفه دوم شامل خطوط برنامه‌ای است که پیرو برخی روابط فیزیکی و عددی در روش شبکه بولتزمن هستند. این خطوط و روابط متناظر در جدول ۴-۱ به تفصیل بررسی شده اند.

جدول ۴-۱: خطوط مربوط به تحلیل پارمترها و روابط مربوطه

خط برنامه	توضیح
Rho0=1.	ρ_0 مقدار اولیه چگالی است و با توجه به اینکه در جواب مسئله تأثیر ندارد برابر با یک قرار داده می‌شود.
Alpha=(ThowT-.5)/3.	در روش شبکه بولتزمن با مدل سرعت D2Q9 رابطه ضریب نفوذ گرما و زمان آرامش گرمایی (Relaxation time) به صورت زیر است: $\alpha = (\tau_T - 0.5)/3.0$
Visco=Alpha*Pr	$\Pr = \frac{\nu}{\alpha}$
Thow=3.*Visco+.5	در روش شبکه بولتزمن با مدل سرعت D2Q9 رابطه ویسکوزیته سینماتیکی و زمان آرامش جریان (Relaxation time) به صورت زیر است: $\tau = 3.0\nu + 0.5$
gBeta=Ra*Visco*Alpha/ ((TbH-TbC)*Float(m*m*m))	gBeta در واقع حاصلضرب شتاب گرانش در ضریب انبساط گرمایی $(g \times \beta)$ است و از آنجا که این دو پارانتر همواره با هم هستند به صورت یک پارامتر در کد وارد شده‌اند و از رایلی با رابطه زیر بدست می‌آیند: $g \times \beta = \mathrm{Ra}\,\nu\alpha (T_H - T_C) H^2 (T_H - T_C)$ گفتنی است که $(T_H - T_C)$ اختلاف دمای دو دیواره گرم و سرد است.
VScale=Visco*sqrt(Ra*Alpha/Visco) /Float(m)	Vscale در واقع Scale سرعت است و از رابطه زیر بدست می‌آید: $\text{scale of velocity} = \sqrt{\frac{Ra\alpha\nu}{H^2}}$
TRef=.5*(TbC+TbH)	دمای مرجع که البته تأثیری بر پاسخ مسئله ندارد معمولاً با رابطه $T_{ref} = \frac{T_C + T_H}{2}$ بدست می‌آید.

وظیفه سوم مقداردهی اولیه ماتریس‌ها است که به وضوح در انتهای سابروتین دیده می‌شود و نیازی به توضیح بیشتر احساس نمی‌شود.

۵- سابروتین Fluid_Flow

این سابروتین که وظیفه محاسبه سرعت و پارامترهای جریانی را دارد، مطابق روش شبکه بولتزمن از چهار بخش زیر تشکیل شده است:

۱- مرحله برخورد(Collision):

در این مرحله تابع توزیع چگالی از روی سرعت‌ها، چگالی و نیروهای خارجی در هر نقطه و جهت سرعتی محاسبه می‌شود.

۲- مرحله انتشار(Streamming):

در این مرحله تابع توزیع چگالی هر نقطه در جهت‌های مشخص شبکه و متناسب با آن منتشر شده و بر نقاط کناری اثر خود را منتقل می‌کند.

۳- شرط مرزی:

در این مرحله شرط مرزی دیواره‌های اطراف که در اینجا همگی شرط مرزی بازگشت به عقب(Bounce back) هستند بر گره‌های مرزی اعمال می‌شود.

۳- محاسبه سرعت‌ها و چگالی:

در این مرحله با کمک روابط بین دیدگاهی مزوسکوپیک و ماکروسکوپیک، چگالی و سرعت‌های افقی و عمودی از تابع توزیع چگالی دیدگاه مزوسکوپیک روش شبکه بولتزمن محاسبه می‌شود.

در ادامه پس از معرفی برخی از پارامترهای روش شبکه بولتزمن تک تک مراحل فوق با توجه به خطوط برنامه و روابط مربوطه توضیح داده خواهد شد.

در جدول ۵-۱ پارامترهای روش شبکه بولتزمن توضیح داده شده‌اند.

جدول۵-۱: تعریف متغیرهای روش شبکه بولتزمن

متغیر داخل فایل	تعریف متغیر	نوع متغیر
b	تعداد جهت‌های سرعت است که در مدل سرعت D2Q9 برابر ۹ است ولی با توجه به اینکه در کد حاضر جهت سرعت‌ها از صفر شمرده می-شود، برابر ۸ است.	عدد طبیعی
w	ω_i ضریب وزنی جهت‌های سرعت شبکه بوده و در جهت‌های i از صفر تا ۸ و بر پایه مدل سرعت D2Q9 مقادیر زیر را دارد: $\omega_i = \begin{cases} \omega_0 = 4/9 \\ \omega_{1-4} = 1/9 \\ \omega_{5-8} = 1/36 \end{cases}$	عدد طبیعی
ex و ey	به ترتیب سرعت افقی و عمودی شبکه هستند که به صورت زیر تعریف می‌شوند: $\left(e_x, e_y\right)_{i=0,8} = \begin{bmatrix} (0,0) \\ (1,0) \\ (0,1) \\ (-1,0) \\ (0,-1) \\ (1,1) \\ (-1,1) \\ (-1,-1) \\ (1,-1) \end{bmatrix}$	عدد طبیعی
Cs	سرعت صوت در شبکه است که در مدل سرعت D2Q9 برابر $1/\sqrt{3}$ می‌باشد.	
Δx و Δt	به ترتیب تغییر زمان و تغییر مکان هستند و برابر یک فرض می‌شوند.	
f	تابع توزیع چگالی است که در مرحله برخورد توضیح داده خواهد شد.	عدد حقیقی
feq	تابع توزیع تعادلی چگالی است که در مرحله برخورد توضیح داده خواهد شد.	عدد حقیقی

گفته شد که هدف مرحله برخورد محاسبه تابع توزیع چگالی است ولی لازمه آن محاسبه تابع توزیع تعادلی چگالی است. روابط و خطوط برنامه مربوط به محاسبه این دو متغیر در جدول ۵-۲ آورده شده است.

جدول ۵-۲: خطوط برنامه مربوط به محاسبه تابع توزیع و تابع توزیع تعادلی چگالی در مرحله برخورد

برنامه	هدف	رابطه
Do i=1,n-1 Do j=1,m-1 t1=u(i,j)*u(i,j)+v(i,j)*v(i,j) Do k=0,b t2=u(i,j)*ex(k)+v(i,j)*ey(k) feq(k,i,j)=Rho(i,j)*w(k)*(1.+3.*t2+4.5*t2*t2-1.5*t1) End Do End Do End Do	محاسبه تابع توزیع تعادلی	$f_i^{eq} = \omega_i \rho \left[1 + \frac{\mathbf{e}_i \cdot u}{Cs^2} + \frac{(\mathbf{e}_i \cdot u)^2}{2Cs^4} - \frac{u \cdot u}{2Cs^2} \right]$
Do i=1,n-1 Do j=1,m-1 Do k=0,b GForce=3.*w(k)*gBeta*(Th(i,j)-TRef)*Rho(i,j)*ey(k) f(k,i,j)=f(k,i,j)-(f(k,i,j)-feq(k,i,j))/Thow(i,j)+GForce End Do End Do End Do	محاسبه تابع توزیع	$f_i\left(x+\mathbf{e}_i\Delta t, t+\Delta t\right) = f_i(x,t) - \frac{1}{\tau}\left[f_i(x,t) - f_i^{eq}(x,t) \right] + \Delta t F_i$

مرحله بعدی انتشار است که همانطور که گفته شده است، توابع توزیع را در جهت‌های مختلف از شبکه‌ای به شبکه دیگر منتقل می‌کند. کد مربوط به این قسمت به صورت زیر است. گفتنی است که روش کد نویسی این قسمت انواع مختلفی دارد و این شکل آن که در پایین مشاهده می‌شود روش ابتکاری نگارنده بوده و به نظر می‌رسد که علاوه بر کوتاهی نحوه انتشار تابع توزیع را بهتر نمایش می‌دهد.

```
f(1,1:n,:) =f(1,0:n-1,:)

f(2,:,1:m) =f(2,:,0:m-1)

f(3,0:n-1,:) =f(3,1:n,:)

f(4,:,0:m-1) =f(4,:,1:m)

f(5,1:n,1:m) =f(5,0:n-1,0:m-1)

f(6,0:n-1,1:m) =f(6,1:n,0:m-1)

f(7,0:n-1,0:m-1)= f(7,1:n,1:m)

f(8,1:n,0:m-1) =f(8,0:n-1,1:m)
```

مرحله بعدی مربوط به اعمال شرط مرزی است، در کد حاضر برای هر چهار دیواره محفظه از شرط مرزی بازگشت به عقب(Bounce back) استفاده شده است. کد و روابط مربوط به اعمال شرط مرزی در جدول ۵-۳ آورده شده است.

جدول ۵-۳: دیواره و شرط مرزی اعمال شده برای آن در کد حاضر

دیواره	کد اعمال شرط مرزی	توضیح
دیواره پایینی	f(2,0:n,0)=f(4,0:n,0) f(5,0:n,0)=f(7,0:n,0) f(6,0:n,0)=f(8,0:n,0)	در واقع تابع توزیع در جهت مجهول برابر با مقدار تابع توزیع در جهت مخالف قرار داده می‌شود.
دیواره بالایی	f(7,0:n,m)=f(5,0:n,m) f(4,0:n,m)=f(2,0:n,m) f(8,0:n,m)=f(6,0:n,m)	در واقع تابع توزیع در جهت مجهول برابر با مقدار تابع توزیع در جهت مخالف قرار داده می‌شود.
دیواره سمت راست	f(3,n,0:m)=f(1,n,0:m) f(7,n,0:m)=f(5,n,0:m) f(6,n,0:m)=f(8,n,0:m)	در واقع تابع توزیع در جهت مجهول برابر با مقدار تابع توزیع در جهت مخالف قرار داده می‌شود.
دیواره سمت چپ	f(1,0,0:m)=f(3,0,0:m) f(5,0,0:m)=f(7,0,0:m) f(8,0,0:m)=f(6,0,0:m)	در واقع تابع توزیع در جهت مجهول برابر با مقدار تابع توزیع در جهت مخالف قرار داده می‌شود.

مرحله نهایی سابروتین Fluid_Flow بدست آوردن سرعت‌ها و چگالی ماکروسکوپی از توابع توزیع است که کد و روابط مربوطه در جدول ۵-۴ آورده شده است.

جدول ۵-۴: خطوط برنامه مربوط به محاسبه سرعت و چگالی

خطوط برنامه	رابطه
Rho(i,j)=Sum(f(:,i,j)) u(i,j)=Sum(f(:,i,j)*ex(:))/Rho(i,j) v(i,j)=Sum(f(:,i,j)*ey(:))/Rho(i,j)	$\rho = \sum_{i=0}^{b-1} f_i$ $\rho u = \sum_{i=0}^{b-1} \mathbf{e}_i f_i$

در بخش بعدی سابروتین Temperature_Distribution که حوزه دما را به روش شبکه بولتزمن حل می‌کند، توضیح داده خواهد شد. گفتنی است که سابروتین Temperature_Distribution و سابروتین Fluid_Flow بسیار شبیه به هم هستند ولی مطالب به نوعی ارائه خواهند شد که خواننده با مطالعه مجزا این سابروتین‌ها نیز با روش شبکه بولتزمن آشنا شود.

۶- سابروتین Temperature_Distribution

این سابروتین وظیفه محاسبه دما را دارد و مطابق روش شبکه بولتزمن از چهار مرحله زیر تشکیل شده است:

۱- مرحله برخورد(Collision):

در این مرحله به ترتیب تابع توزیع تعادلی دما و تابع توزیع دما از روی سرعت‌ها و دما، و در هر نقطه و جهت محاسبه می‌شوند.

۲- مرحله انتشار(Streamming):

در این مرحله تابع توزیع دما هر نقطه در جهت‌های مشخص شبکه و متناسب با آن منتشر شده و بر نقاط کناری اثر خود را منتقل می‌کند.

۳- شرط مرزی:

در این مرحله شرط مرزی دیواره‌های اطراف با توجه به اینکه دیواره‌های عمودی دما ثابت و دیواره‌های افقی عایق هستند، مطابق انواع شرایط مرزی بازگشت به عقب(Bounce back) بر آنها اعمال می‌شود.

۳- محاسبه دما:

در این مرحله با کمک روابط بین دیدگاهی مزوسکوپیک و ماکروسکوپیک، دمای ماکروسکوپی از تابع توزیع دما که دیدگاه مزوسکوپیک روش شبکه بولتزمن را دارد، محاسبه می‌شود.

در ادامه پس از معرفی برخی از پارامترهای روش شبکه بولتزمن تک تک مراحل فوق با توجه به خطوط برنامه و روابط مربوطه توضیح داده خواهد شد.

در جدول ۶-۱ پارامترهای روش شبکه بولتزمن توضیح داده شده‌اند.

جدول۶-۱: تعریف متغیرهای روش شبکه بولتزمن

متغیر داخل فایل	تعریف متغیر	نوع متغیر
b	تعداد جهت‌های سرعت است که در مدل سرعت D2Q9 برابر ۹ است ولی با توجه به اینکه در کد حاضر جهت سرعت‌ها از صفر شمرده می‌شود، برابر ۸ است.	عدد طبیعی
w	ω_i ضریب وزنی جهت‌های سرعت شبکه بوده و در جهت‌های i از صفر تا ۸ و بر پایه مدل سرعت D2Q9 مقادیر زیر را دارد: $$\omega_i = \begin{cases} \omega_0 = 4/9 \\ \omega_{1-4} = 1/9 \\ \omega_{5-8} = 1/36 \end{cases}$$	عدد طبیعی
ex و ey	به ترتیب سرعت افقی و عمودی شبکه هستند که به صورت زیر تعریف می‌شوند: $$\left(e_x, e_y\right)_{i=0,8} = \begin{bmatrix} (0,0) \\ (1,0) \\ (0,1) \\ (-1,0) \\ (0,-1) \\ (1,1) \\ (-1,1) \\ (-1,-1) \\ (1,-1) \end{bmatrix}$$	عدد طبیعی
Cs	سرعت صوت در شبکه است که در مدل سرعت D2Q9 برابر $1/\sqrt{3}$ می‌باشد.	
Δx و Δt	به ترتیب تغییر زمان و تغییر مکان هستند و برابر یک فرض می‌شوند.	
f	تابع توزیع چگالی است که در مرحله برخورد توضیح داده خواهد شد.	عدد حقیقی
feq	تابع توزیع تعادلی چگالی است که در مرحله برخورد توضیح داده خواهد شد.	عدد حقیقی

گفته شد که هدف مرحله برخورد محاسبه تابع توزیع دما است ولی لازمه آن محاسبه تابع توزیع تعادلی دما می‌باشد. روابط و خطوط برنامه مربوط به محاسبه این دو متغیر در جدول ۶-۲ آورده شده است.

جدول ۶-۲: خطوط برنامه مربوط به محاسبه تابع توزیع و تابع توزیع تعادلی چگالی در مرحله برخورد

برنامه	رابطه
Do i=1,n-1 Do j=0,m t1=u(i,j)*u(i,j)+v(i,j)*v(i,j) Do k=0,b t2=u(i,j)*ex(k)+v(i,j)*ey(k) geq(k,i,j)=Th(i,j)*w(k)*(1.+3.*t2+4.5*t2*t2-1.5*t1) g(k,i,j)=g(k,i,j)+(geq(k,i,j)-g(k,i,j))/ThowT End Do End Do End Do	$g_i^{eq} = \omega_i T\left[1+\frac{\mathbf{e}_i\cdot u}{Cs^2}+\frac{(\mathbf{e}_i\cdot u)^2}{2Cs^4}-\frac{u\cdot u}{2Cs^2}\right]$ $g_i\left(x+\mathbf{e}_i\Delta t,t+\Delta t\right) = g_i(x,t)-\frac{1}{\tau_T}\left[g_i(x,t)-g_i^{eq}(x,t)\right]$

مرحله بعدی انتشار است که همانطور که گفته شده است، توابع توزیع را در جهت‌های مختلف از شبکه‌ای به شبکه دیگر منتقل می‌کند. کد مربوط به این قسمت به صورت زیر است. گفتنی است که روش کد نویسی این قسمت انواع مختلفی دارد و این شکل آن که در پایین مشاهده می‌شود روش ابتکاری نگارنده بوده و به نظر می‌رسد که علاوه بر کوتاه بودن، نحوه انتشار تابع توزیع را بهتر نمایش می‌دهد.

g(1,1:n,:)=g(1,0:n-1,:)

g(2,:,1:m)=g(2,:,0:m-1)

g(3,0:n-1,:)=g(3,1:n,:)

g(4,:,0:m-1)=g(4,:,1:m)

g(5,1:n,1:m)=g(5,0:n-1,0:m-1)

g(6,0:n-1,1:m)=g(6,1:n,0:m-1)

g(7,0:n-1,0:m-1)=g(7,1:n,1:m)

g(8,1:n,0:m-1)=g(8,0:n-1,1:m)

مرحله بعدی مربوط به اعمال شرط مرزی است، در کد حاضر برای دیواره‌های دما ثابت و عایق شکل‌های شرط مرزی بازگشت به عقب(Bounce back) مربوط به آنها استفاده شده است. کد و روابط مربوط به اعمال شرط مرزی در جدول ۶-۳ آورده شده است.

جدول ۶-۳: دیواره و شرط مرزی اعمال شده برای آن در کد حاضر

دیواره	کد اعمال شرط مرزی	توضیح
دیواره سمت راست	g(3,n,0:m)=TbC*(w(3)+w(1))-g(1,n,0:m) g(6,n,0:m)=TbC*(w(6)+w(8))-g(8,n,0:m) g(7,n,0:m)=TbC*(w(7)+w(5))-g(5,n,0:m)	برای اعمال شرط مرزی دما ثابت، تابع توزیع دما در جهت مجهول از دمای معلوم و تابع توزیع در جهت مخالف بدست می‌آید.
دیواره سمت چپ	g(5,0,0:m)=TbH*(w(5)+w(7))-g(7,0,0:m) g(1,0,0:m)=TbH*(w(1)+w(3))-g(3,0,0:m) g(8,0,0:m)=TbH*(w(8)+w(6))-g(6,0,0:m)	برای اعمال شرط مرزی دما ثابت، تابع توزیع دما در جهت مجهول از دمای معلوم و تابع توزیع در جهت مخالف بدست می‌آید.
دیواره بالایی	g(7,0:n,m)=g(7,0:n,m-1) g(4,0:n,m)=g(4,0:n,m-1) g(8,0:n,m)=g(8,0:n,m-1)	برای اعمال شرط مرزی عایق کافیست که گرادیان تابع توزیع دما را در مرز برابر صفر قرار دهیم.
دیواره پایینی	g(2,0:n,0)=g(2,0:n,1) g(6,0:n,0)=g(6,0:n,1) g(5,0:n,0)=g(5,0:n,1)	برای اعمال شرط مرزی عایق کافیست که گرادیان تابع توزیع دما را در مرز برابر صفر قرار دهیم.

مرحله نهایی سابروتین Fluid_Flow بدست آوردن دما ماکروسکوپیک از تابع توزیع است که کد و رابطه مربوطه در جدول ۵-۴ آورده شده است.

جدول ۶-۴: خطوط برنامه مربوط به محاسبه دما

خطوط برنامه	رابطه
Do i=1,n-1 Do j=0,m Th(i,j)=Sum(g(:,i,j)) End Do End Do	$T=\sum_{i=0}^{b-1} g_i$

در بخش قبلی سابروتین Residual که مقادیر باقیمانده دما و جریان را برای سنجش همگرایی محاسبه میکند، بررسی خواهد شد.

۷- سابروتین Residual

این سابروتین در حقیقت معیار همگرایی را برای سنجش در متن اصلی محاسبه می‌کند. گفتی است که معیارهای متفاوتی مانند برابر شدن ناسلت روی دو دیواره سرد و گرم و... برای سنجش همگرایی پیشنهاد شده است ولی تجربه نشان داده که بهترین و سازگارترین معیار برای روش شبکه بولتزمن استفاده از معیار L_2 (L_2 norm) می‌باشد. برتری این معیار و در واقع روش محاسبه با قیمانده این است که باقیمانده‌های دما و جریان و ... همگی دارای یک scale بوده و با پایا شدن جریان به سرعت کاهش می‌یابند و بدین ترتیب ضمن اینکه از محاسبات اضافی پرهیز شده، احتمال رخ دادن همگرایی کاذب که ناشی از برابر نبودن scale باقیمانده‌هاست وجود ندارد.

در جدول ۷-۱ کد و رابطه مربوط به محاسبه باقیمانده جریان و دما آورده شده است.

جدول۷-۱: خطوط برنامه مربوط به محاسبه باقیمانده‌های جریان و دما

خطوط برنامه	رابطه
Do i=0,n Do j=0,m Su=Su+U(i,j)**2+V(i,j)**2 St=St+Th(i,j)**2 ResF=ResF+(U(i,j)-UOld(i,j))**2+(V(i,j)-VOld(i,j))**2 ResT=ResT+(Th(i,j)-ThOld(i,j))**2 UOld(i,j)=U(i,j) VOld(i,j)=V(i,j) ThOld(i,j)=Th(i,j) End Do End Do ResF=Sqrt(ResF)/Sqrt(Su) ResT=Sqrt(ResT)/Sqrt(St)	$\text{Residual of flow}=\frac{\sum\|u(x,t+\Delta t)-u(x,t)\|}{\sum\|u(x,t)\|}$ $\text{Residual of temperature}=\frac{\sum\|T(x,t+\Delta t)-T(x,t)\|}{\sum\|T(x,t)\|}$

در بخش بعدی سابروتین Output که کار محاسبه و چاپ خروجی برنامه را در فایل‌های خروجی دارد، مورد بررسی قرار خواهد گرفت.

۸- سابروتین Output

بخش‌های اصلی این سابروتین به شرح زیر هستند:

۱- محاسبه کانتور خط جریان

۲- چاپ کانتورهای دما، سرعت‌های افقی و عمودی و خط جریان در فایل Result.dat مطابق با فرمت برنامه Tecplot

۳- محاسبه ناسلت روی دیواره گرم

۴- چاپ سایر دیتاهای مورد نیاز که در اینجا سرعت‌های افقی و عمودی و دما در امتداد خطوط میانی افقی یا عمودی

خطوط برنامه و نکات مهم بخش‌های بالا در جدول ۸-۱ آورده شده است.

جدول۸-۱: خطوط برنامه و نکات مهم مربوط به بخشهای مختلف سابروتین Output

بخش	خطوط برنامه	نکات مهم
۱	Strf(0,:)=0. Do i=1,n Rhoav=0.5*(Rho(i-1,0)+Rho(i,0)) Strf(i,0)=Strf(i-1,0)-Rhoav*0.5*(v(i-1,0)+v(i,0)) Do j=1,m Rhom=0.5*(Rho(i,j)+Rho(i,j-1)) Strf(i,j)=Strf(i,j-1)+Rhom*0.5*(u(i,j-1)+u(i,j)) End Do End Do	محاسبه کانتور خطوط جریان عیناً از کتب مرجع روش شبکه بولتزمن برداشته شده و توضیح خاصی ندارد.
۲	Open(20,file='Result.dat') Write(20,*)'VARIABLES =X,Y,U,V,P,T,Stream' Write(20,*)"ZONE ","I=",n+1,"J=",m+1 Do j=0,m Do i=0,n Write(20,*)float(i)/float(m),float(j)/float(m),u(i,j)/VScale,v(i,j)/VScale,Rho(i,j)/.3,Th(i,j),Strf(i,j) End Do End Do Close(20)	طبق فرمت نرم‌افزار Tecplot اگر خط سوم با I شروع شود، خط چهارم حتماً باید با j شروع شود و بالعکس.

ادامه جدول۸-۱: خطوط برنامه و نکات مهم مربوط به بخشهای مختلف سابروتین Output

بخش	خطوط برنامه	نکات مهم
۳	Open(10,file='Nu.dat') Nuavg=0. Do j=0,m Nu=2.*(TbH-Th(1,j))*float(n) Nuavg=Nuavg+Nu End Do Nuavg=Nuavg/float(m) Write(10,*)Nuavg Close(10)	نکته مهم این است که ضریب ۲ در خط چهارم به علت محاسبه گرادیان دما در نصف اندازه شبکه است.
۴	Do i=0,n Write(30,*)Float(i)/Float(n),V(i,m/2)/VScale Write(40,*)Float(i)/Float(n),Th(i,m/2) End Do Do j=0,n Write(50,*)U(n/2,j)/VScale,Float(j)/Float(m) End Do	بنا بر نیاز می‌توان هر دیتایی را در خروجی چاپ نمود.

۹- سابروتین After_Solution

پس از اتمام فرایند حل مسئله، سابروتین After_Solution بار دیگر سابروتین Output را فراخوانی نموده و گزارشات اتمام حل را به نمایش در می‌آورد.

۱۰- منابع و مراجع

[1] A. Mohamad, *Lattice Boltzmann Method*: Springer, 2011.

[2] A. A. Mohamad, M. El-Ganaoui, R. Bennacer, Lattice Boltzmann simulation of natural convection in an open ended cavity, *International Journal of Thermal Sciences,* Vol. 48, No. 10, pp. 1870-1875, 10//, 2009.

[3] G. Kefayati, S. Hosseinizadeh, M. Gorji, H. Sajjadi, Lattice Boltzmann simulation of natural convection in tall enclosures using water/SiO< sub> 2</sub> nanofluid, *International Communications in Heat and Mass Transfer,* Vol. 38, No. 6, pp. 798-805, 2011.

[4] K. Khanafer, K. Vafai, M. Lightstone, Buoyancy-driven heat transfer enhancement in a two-dimensional enclosure utilizing nanofluids, *International Journal of Heat and Mass Transfer,* Vol. 46, No. 19, pp. 3639-3653, 2003.