AD-A103 389 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 10/1
NUMERICAL SOLUTION OF NATURAL CONVECTION IN AN INCLINED RECTANG--ETC(U)
DEC 80 T K TOLTZIEN
UNCLASSIFIED AFIT/GAE/AA/80D-23 NL

1 OF 2
AD A
103389

AIR FORCE INSTITUTE OF TECHNOLOGY

AIR UNIVERSITY

UNITED STATES AIR FORCE

1 LEVEL II

AFIT/GAE/AA/80D-23

NUMERICAL SOLUTION OF NATURAL
CONVECTION IN AN INCLINED RECTANGULAR
CAVITY WITH PARTITIONS

THESIS

AFIT/GAE/AA/80D-23 Thomas K. Toltzien
Captain USAF

(14)

AFIT/GAE/AA/80D-23

NUMERICAL SOLUTION OF NATURAL CONVECTION

IN AN INCLINED RECTANGULAR CAVITY

WITH PARTITIONS

Master's THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air Training Command

in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

by

Thomas K. Toltzien, B.S.M.E.

Captain USAF

Graduate Aeronautical Engineering

~~December 1980~~

Approved for public release; distribution unlimited.

12225

Acknowledgement

When faced with the task of conducting an independent study, I could not have hoped to bring it to a successful conclusion without the aid and guidance of my thesis advisor and numerous others involved during the duration of the study. It is with this thought in mind that I would like to express my sincere appreciation to Dr. James Hitchcock, my thesis advisor, for his time and effort spent throughout this study. I would also like to thank Dr. Harold Wright and Capt. Robert Roach for their meaningful suggestions, and Mrs. Maripat Meer, who typed this manuscript.

Finally, I include a very special and personal thanks to my wife, Choopit, and daughters, Amy and Ann, for their patience and moral support throughout this study.

Thomas K. Toltzien

Contents

List of Figures

List of Tables

Notation

Symbol

A	Aspect ratio, = L/D
C_p	Constant pressure specific heat
D	Width of enclosure
g	Acceleration due to gravity
g_c	Dimensional constant in Newton's second law
G_R	Grashof number, = $g\beta(\theta_H-\theta_C)D^3/\nu^2$
h	Local heat transfer coefficient, = $q/\theta_C-\theta_H$
k	Thermal conductivity
L	Height of enclosure
Nu	Local Nusselt number, = $\frac{hD}{k}$
$\overline{Nu}$	Mean value of Nusselt number
p	Local dynamic pressure
P	Dimensionless pressure, = $\frac{p_0 D^2 g_c}{\nu^2 \rho}$
P_r	Prandtl number, = $g_c \mu Cp/k$
q	Heat flux at hot wall
S	Inclination angle
t	Time
u	Velocity in x-direction
U	Dimensionless velocity in X-direction, = uD/ν
v	Velocity in y-direction
V	Dimensionless velocity in Y-direction, = vD/ν
x	Distance along hot wall
X	Dimensionless distance along hot wall, = x/D

Symbol

y Distance from hot wall

Y Dimensionless distance from hot wall, $= y/D$

Greek Letters

α Molecular thermal diffusivity, $= \frac{k}{\rho C_p}$

β Volume coefficient of thermal expansion

ΔX Grid spacing in X direction

ΔY Grid spacing in Y direction

$\Delta\tau$ Time increment

ζ Dimensionless vorticity, $= -(\frac{\partial\Psi}{\partial X} + \frac{\partial\Psi}{\partial Y})$

θ Temperature (θ_H, and θ_C refer to the temperatures at the hot and cold walls respectively).

Θ Dimensionless temperature, $= (\theta-\theta_H)/(\theta_C-\theta_H)$

ν Kinematic viscosity

μ Viscosity

ρ Density

τ Dimensionless time, $= t\nu/D^2$

Ψ Dimensionless stream function, such that $U=\partial\Psi/\partial Y$ and $V= -\partial\Psi/\partial X$

ω Relaxation parameter

ϕ Slat angle

Subscripts

i,j Space grid point indices in X and Y directions

opt Value of relaxation parameter giving fastest convergence.

w Wall grid point

Superscripts

n Time index

m Iternation number

Abstract

A numerical investigation was conducted on two-dimensional natural convection within inclined rectangular enclosures partitioned into 45 degree triangular cells. The time dependent governing equations, vorticity, energy, and stream function, were solved by an ADI method and a Gauss-Seidel SOR technique. The numerical procedure was validated for rectangular enclosures, then modified for triangular cells. Heat transfer coefficients were determined for an inclined square enclosure with a diagonal partition for Grashof numbers less than $2.x10^5$ and inclination angles between 10 degrees and 90 degrees. These results show a diagonal partition reduces the heat transferred by natural convection across an inclined square enclosure by more than 50%.

NUMERICAL SOLUTION OF NATURAL CONVECTION IN AN INCLINED RECTANGULAR CAVITY WITH PARTITIONS

I. Introduction

Background

A major factor in the design of solar collectors is the reduction of natural convection heat losses within the collector enclosure. Traditional designs have suppressed natural convection by developing large aspect ratio enclosures or designing a small aspect ratio honeycomb structure to separate the collector plates (Ref.1). However, recent experimental studies by Meyer, el.al. (Ref.2), and Holland (Ref.3) have demonstrated the effectiveness of inclined slats (partitions) in reducing natural convection heat losses. The slats in these studies were positioned to form a series of moderate aspect ratio parallelogram enclosures. Meyer, et.al., also concluded that slats oriented downward from the hot plate resulted in a reduction of convective heat transfer with the minimum heat loss occurring at a slat angle of 45 degrees.

Objective

In the present study, the influence of slats in reducing natural convection will be determined for non-parallel slat arrangements. Specifically, the slats will be oriented so as to partition a moderate aspect ratio rectangular enclosure into a series of triangular regions (see Figure 1). The temperature distribution and the cell heat transfer coefficients will then be determined theoretically by a numerical procedure which will also be developed for this investigation. The heat transfer coefficient for the triangular cells will then be compared to results obtained for similar rectangular enclosures with and without slats. From these results the effectiveness of triangular slat arrangements will be determined.

The first step in this investigation, however, will be to adapt a numerical procedure which is capable of solving the nonlinear partial differential equations which describe natural convection in an inclined enclosure. The implicit finite difference computational scheme developed by Wilkes and Churchill (Ref.4) is just such a procedure. Since the Wilkes and Churchill numerical procedure was developed for a rectangular cavity with one vertical wall hot and the other cold, the method will be first modified to describe an inclined rectangular cavity, then modified again to account for the addition of partitions in the rectangular enclosure.

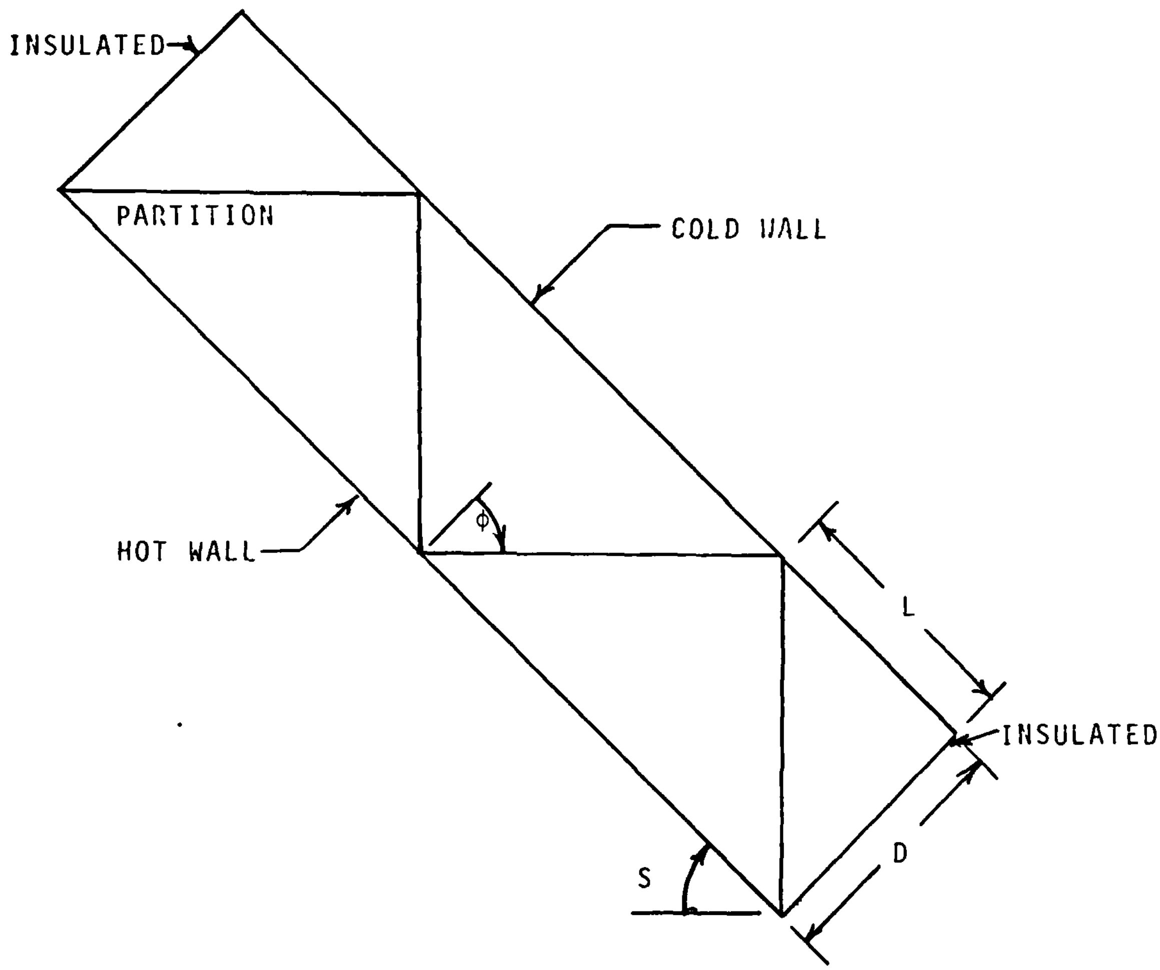


Figure 1. Triangular Partitioned Enclosure.

Scope

In this study, natural convection heat transfer coefficients were evaluated for moderate aspect ratio (A=1.0 to 3.0) rectangular enclosures with thin partitions. The partitions were oriented at ±45 degress to the hot wall. Thermal conduction was assumed to be infinite across the partitions and zero along the partitions. Additionally, the steady state temperature distribution along a partition was considered linear. Finally, since only heat transferred due to natural convection was to be evaluated, thermal radiation heat transfer within the enclosure was neglected.

To follow the development of the present investigation, this report is organized into five additional chapters. The time dependent governing equations which describe natural convection in inclined rectangular cavities are developed in Chapter II. In Chapter III, a numerical procedure is developed to evaluate the governing equations. A discussion of the numerical results is presented in Chapter IV. Finally, the significant conclusions are restated, and recommendations are made for further studies in Chapter V and VI, respectively.

II. Mathematical Model

In this section the governing equations describing natural convection in a 2-D inclined rectangular cavity will be developed. The equations and boundary conditions will then be simplified and rewritten in nondimensionalized form. Finally, an expression for local and average heat transfer coefficient will be derived.

Rectangular Cavity

Consider the inclined, rectangular cavity in Figure 2. The left and right walls are at constant temperatures θ_H, and θ_C, respectively, and the other side walls are either insulated or have a linear temperature distribution. Initially, the fluid in the cavity is motionless, and at a constant temperature equal to the average temperature of the hot and cold walls. The fluid thermodynamic and transport properties except density are considered constant and independent of temperature. Under the Boussinesq Approximation, density changes are expressed only as a function of temperature differences, and density is assumed constant except in the buoyant force terms.

Under these conditions, the fluid in an inclined rectangular cavity is described by the following equations:

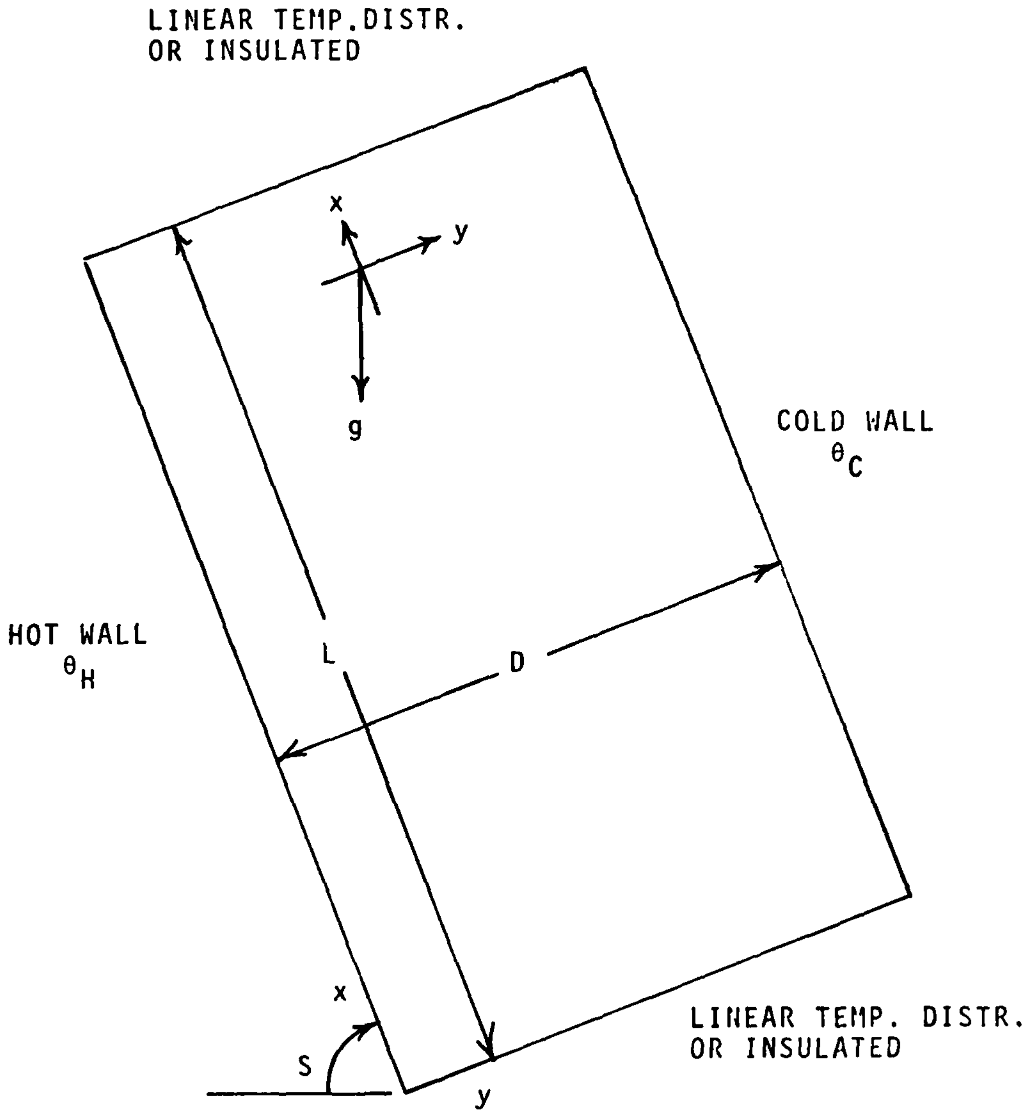


Figure 2. Inclined Rectangular Enclosure Nomenclature.

Conservation of Mass

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \qquad (1)$$

Conservation of Momentum

$$\frac{\rho}{g_c}\frac{\partial u}{\partial t} + \frac{\rho}{g_c}\frac{u\partial u}{\partial x} + \frac{\rho}{g_c}\frac{v\partial u}{\partial y} = -\frac{\partial p}{\partial x} - \frac{\rho g}{g_c}\sin(S) + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \qquad (2)$$

$$\frac{\rho}{g_c}\frac{\partial v}{\partial t} + \frac{\rho}{g_c}\frac{u\partial v}{\partial x} + \frac{\rho}{g_c}\frac{v\partial v}{\partial y} = -\frac{\partial p}{\partial y} - \frac{\rho g}{g_c}\cos(S) + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \qquad (3)$$

Conservation of Energy

$$\frac{\partial \theta}{\partial t} + u\frac{\partial \theta}{\partial x} + v\frac{\partial \theta}{\partial y} = \alpha\left(\frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2}\right) \qquad (4)$$

Coefficient of Volumetric Expansion

$$\beta = -\frac{1}{\rho}\left.\frac{\partial \rho}{\partial \theta}\right|_p = \frac{\rho_0 - \rho}{\rho(\theta - \theta_0)} \qquad (5)$$

The associated boundary and initial conditions are as follows:

$$\theta(x,0,t) = \theta_H \qquad (6)$$

$$\theta(x,D,t) = \theta_C \qquad (7)$$

Linear Temperature Side Walls

$$\theta(0,y,t) = \theta_H + (\theta_H - \theta_C)y/D$$

$$\theta(L,y,t) = \theta_H + (\theta_H - \theta_C)y/D \tag{8}$$

Adiabatic Side Walls

$$\frac{\partial\theta}{\partial x}(0,y,t) = 0$$

$$\frac{\partial\theta}{\partial x}(L,y,t) = 0 \tag{9}$$

Cavity Wall No-Slip Condition

$$u(x,0,t) = u(x,D,t) = 0$$

$$v(0,y,t) = v(L,y,t) = 0 \tag{10}$$

Initial Conditions

$$u(x,y,0) = 0$$

$$v(x,y,0) = 0 \tag{11}$$

$$\theta(x,y,0) = \frac{\theta_H + \theta_C}{2}$$

Since the pressure gradient in the cavity results in part from the change in elevation of the fluid, it is convenient to derive an expression for the total pressure gradient in terms of the dynamic pressure gradient and the change in pressure due to the weight of the fluid.

$$p = p_0 - \frac{g\rho_0}{g_c}\left[(\sin S)x+(\cos S)y\right] \tag{12}$$

Differentiating this expression yields

$$\frac{\partial p}{\partial x} = \frac{\partial p_0}{\partial x} - \frac{g}{g_c}\,\rho_0\,\sin(S) \tag{13}$$

$$\frac{\partial p}{\partial y} = \frac{\partial p_0}{\partial y} - \frac{g}{g_c}\,\rho_0\,\cos(S) \tag{14}$$

Substituting from Equations (13),(14) and (5) into equations (2) and (3) and simplifying

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{g_c}{\rho}\frac{\partial p_0}{\partial x} - \beta(\theta_0-\theta)g\,\sin(S)+\nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{15}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{g_c}{\rho}\frac{\partial p_0}{\partial y} - \beta(\theta_0-\theta)g\,\cos(S)+\nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{16}$$

Equations (1), (15), (16) and (4), and the associated boundary and initial conditions can be restated in nondimensional form. The nondimensional parameters are defined in the List of Symbols.

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0 \tag{17}$$

$$\frac{\partial U}{\partial \tau} + U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} = - \frac{\partial P}{\partial X} - G_R\ \Theta\ \sin(S) + \frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2} \tag{18}$$

$$\frac{\partial V}{\partial \tau} + U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y} = - \frac{\partial P}{\partial Y} - G_R\ \Theta\ \sin(S) + \frac{\partial^2 V}{\partial X^2} + \frac{\partial^2 V}{\partial Y^2} \tag{19}$$

$$\frac{\partial \Theta}{\partial \tau} + U\frac{\partial \Theta}{\partial X} + V\frac{\partial \Theta}{\partial Y} = \frac{1}{P_r}\left[\frac{\partial^2 \Theta}{\partial X^2} + \frac{\partial^2 \Theta}{\partial Y^2}\right] \tag{20}$$

$$\begin{gathered}
\Theta(X,0,\tau) = 0 \\
\Theta(X,1,\tau) = 1.0 \\
\Theta(0,Y,\tau) = \Theta(A,Y,\tau) = Y \\
\frac{\partial \Theta}{\partial X}(0,Y,\tau) = \frac{\partial \Theta}{\partial X}(A,Y,\tau) = 0 \\
U(X,0,\tau) = U(X,1,\tau) = V(A,Y,\tau) = V(0,Y,\tau) = 0 \\
U(X,Y,0) = V(X,Y,0) = 0 \\
\Theta(X,Y,0) = 0.5
\end{gathered} \tag{21}$$

The pressure terms in Equations (18) and (19) can now be eliminated by differentiating Equation (18) and (19) with respect to Y and X, respectively, subtracting and simplifying the resulting expression with Equation (17). This process yields Equation (22).

$$\frac{\partial}{\partial \tau}\left[\frac{\partial U}{\partial Y} - \frac{\partial V}{\partial X}\right] + U\left[\frac{\partial U}{\partial Y} - \frac{\partial V}{\partial X}\right] + V\left[\frac{\partial U}{\partial Y} - \frac{\partial V}{\partial X}\right]$$

$$= G_R\left[\cos(S)\frac{\partial \Theta}{\partial X} - \sin(S)\frac{\partial \Theta}{\partial Y}\right] + \frac{\partial^2}{\partial X^2}\left[\frac{\partial U}{\partial Y} - \frac{\partial V}{\partial X}\right]$$

$$+ \frac{\partial^2}{\partial Y^2}\left[\frac{\partial U}{\partial Y} - \frac{\partial V}{\partial X}\right] \tag{22}$$

If a vorticity function (ζ) is now defined as

$$-\zeta = \frac{\partial U}{\partial Y} - \frac{\partial V}{\partial X} \tag{23}$$

and the stream function (Ψ) is introduced, where

$$U = \frac{\partial \Psi}{\partial Y} \tag{24}$$

$$V = -\frac{\partial \Psi}{\partial X} \tag{25}$$

Equation (22) can be rewritten in terms of vorticity, and Equation (23) in terms of the stream function.

$$\frac{\partial \zeta}{\partial \tau} + U\frac{\partial \zeta}{\partial X} + V\frac{\partial \zeta}{\partial Y} = G_R\left[\sin(S)\frac{\partial \Theta}{\partial Y} - \cos(S)\frac{\partial \Theta}{\partial X}\right]$$

$$+ \frac{\partial^2 \zeta}{\partial X^2} + \frac{\partial^2 \zeta}{\partial Y^2} \tag{26}$$

$$-\zeta = \frac{\partial \Psi}{\partial X} + \frac{\partial \Psi}{\partial Y} \tag{27}$$

The governing equations and boundary conditions which describe natural convection in an inclined rectangular cavity can now written in their final form.

Vorticity Equation

$$\frac{\partial \zeta}{\partial \tau} + U\frac{\partial \zeta}{\partial X} + V\frac{\partial \zeta}{\partial Y} = G_R\left[\sin(S)\frac{\partial \Theta}{\partial Y} - \cos(S)\frac{\partial \Theta}{\partial X}\right]$$

$$+ \frac{\partial^2 \zeta}{\partial X^2} + \frac{\partial^2 \zeta}{\partial Y^2} \qquad (28)$$

Energy Equation

$$\frac{\partial \Theta}{\partial \tau} + U\frac{\partial \Theta}{\partial X} + V\frac{\partial \Theta}{\partial Y} = \frac{1}{P_R}\left[\frac{\partial^2 \Theta}{\partial X^2} + \frac{\partial^2 \Theta}{\partial Y^2}\right] \qquad (29)$$

Stream Function Equation

$$-\zeta = \frac{\partial^2 \Psi}{\partial X^2} + \frac{\partial^2 \Psi}{\partial Y^2} \qquad (30)$$

Velocity Component Equations

$$U = \frac{\partial \Psi}{\partial Y} \qquad (31)$$

$$V = -\frac{\partial \Psi}{\partial X} \qquad (32)$$

Boundary Conditions

$$U(0,Y,\tau) = U(A,Y,\tau) = 0$$

$$V(X,0,\tau) = V(X,1,\tau) = 0$$

$$\Psi(0,Y,\tau) = \Psi(A,Y,\tau) = \Psi(X,0,\tau) = \Psi(X,1,\tau) = 0 \tag{33}$$

$$\Theta(X,0,\tau) = 0$$

$$\Theta(X,1,\tau) = 1.0$$

Adiabatic Side Walls

$$\frac{\partial\Theta}{\partial X}(0,Y,\tau) = \frac{\partial\Theta}{\partial X}(A,Y,\tau) = 0 \tag{34}$$

Linear Temperature Side Walls

$$\Theta(0,Y,\tau) = \Theta(A,Y,\tau) = Y \tag{35}$$

Initial Conditions:

$$U(X,Y,0) = V(X,Y,0) = 0$$

$$\Psi(X,Y,0) = \zeta(X,Y,0) = 0 \tag{36}$$

$$\Theta(X,Y,0) = 0.5$$

<u>Heat Transfer Coefficients</u>

Local and average heat transfer coefficients are expressed as non-dimensional Nusselt numbers.

$$Nu = hD/k \tag{37}$$

Local Nusselt Number

The convection heat transfer coefficient (h) at the hot wall is defined as follows:

$$h = \frac{-k(\partial\theta/\partial y)_{wall}}{\theta_H - \theta_C} \qquad (38)$$

Introducing the definition of Nu, Equation (37), Θ , and Y , Equation (38) becomes

$$Nu = \left(\frac{\partial\Theta}{\partial Y}\right)_{wall} \qquad (39)$$

Average Nusselt Number

To obtain an average Nusselt Number, $\overline{Nu}$, along the hot wall, the local Nusselt Numbers, Nu, are integrated over the length of the hot wall.

$$\overline{Nu} = \frac{\int_0^L Nu\,dx}{\int_0^L dx} \qquad (40)$$

Partitioned Cavity

A review of the mathematical model of natural convection in an inclined rectangular enclosure shows the governing Equations (28), (29), (30), (31) and (32) are independent of the enclosure geometry. Therefore, these equations can be applied to triangular enclosures, and only the boundary conditions need to be changed.

The partitions will be placed in the enclosure at $\phi=\pm45$ degrees with respect to the hot wall, and also all partition steady state temperature distributions will be linear. Therefore, applying these assumptions, the boundary conditions for the partitions can now be written.

$$\phi = 45^0$$

$$\Theta(1.0-Y,Y,\tau) = Y \tag{41}$$

$$\Psi(1.0-Y,Y,\tau) = 0$$

$$U(1.0-Y,Y,\tau)\sin\phi+V(1.0-Y,Y,\tau)\sin\phi = 0$$

$$\phi = -45$$

$$\Theta(Y,Y,\tau) = Y$$

$$\Psi(Y,Y,\tau) = 0 \tag{42}$$

$$U(Y,Y,\tau)\sin\phi+V(Y,Y,\tau)\sin\phi = 0$$

Now the partition boundary conditions can be used in combination with those for the rectangular cavity to completely describe the boundaries of an inclined rectangular enclosure with partitions.

III. Numerical Procedure

Rectangular Enclosure

This section contains a description of the procedure that was used to solve the system of equations developed in the previous section. This procedure was adapted from a finite difference computational procedure developed by Wilkes and Churchill (Ref.4) to study natural convection in a rectangular enclosure with one vertical wall heated and the other cooled. This procedure is illustrated by a flow chart in Figure 3. Because the procedures are so similar, tests of stability and convergence were not rigorously carried out in this study.

The geometry of the rectangular enclosure and the finite difference nomenclature are shown in Figure 4. The mesh spacing in the X-direction is ΔX and the Y-direction is ΔY . Subscripts (i,j) are associated with each mesh point, so that the space variables may be expressed as $X_i=(i-1)\Delta X$, for $i=1,2,\ldots,I$ and $Y_j=(j-1)\Delta Y$, for $j=1,2,3,\ldots,J$. Time is segmented into equal intervals $\Delta\tau$ so that the nondimensional time τ is $\tau=n\Delta\tau$ for $n=0,1,2$. The notation $\phi^n_{i,j}$ denotes the values of the variable at mesh point (i,j) at time level n .

The first equation to be solved from $n\Delta\tau$ to $(n+1)\Delta\tau$ is the energy equation (29), which is parabolic in time,

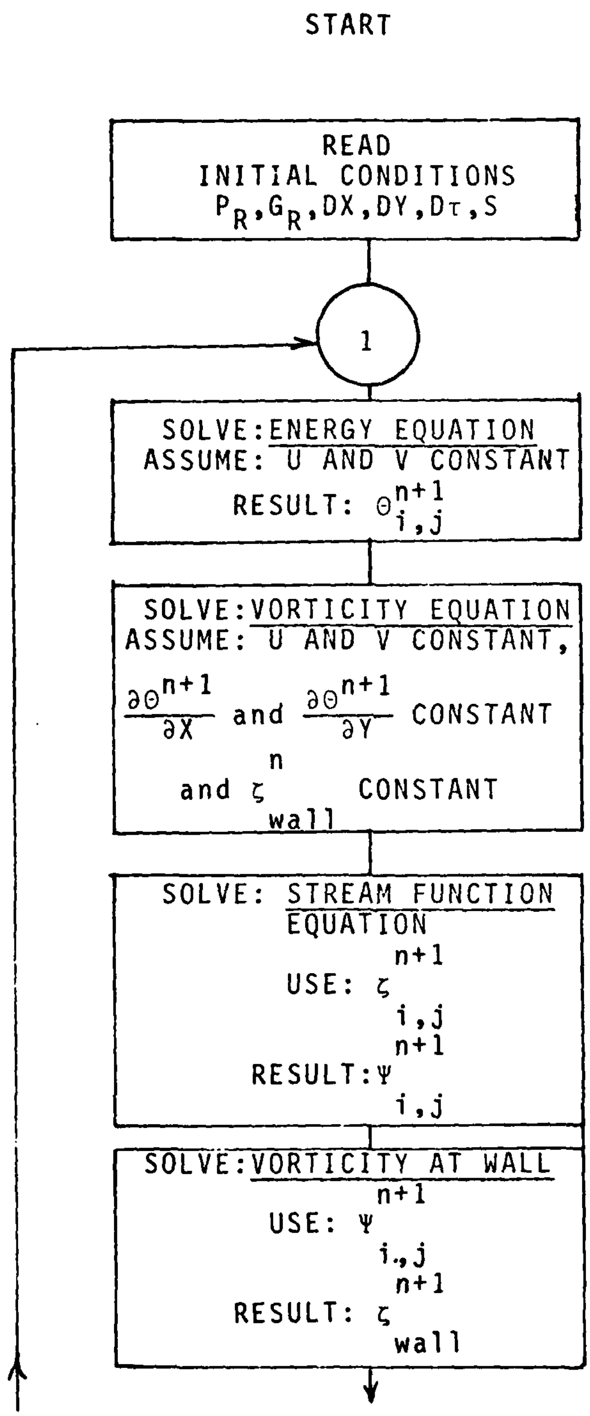


Figure 3. Numerical Procedure Sequence of Solutions of the Governing Equations.

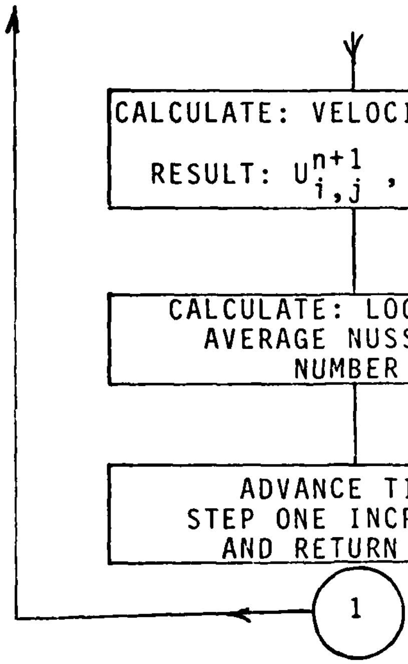


Figure 3. Continuation of Numerical Procedure Sequence of Solution of the Governing Equations.

nonlinear and of second order. The solution to this equation is advanced one time level by using an implicit alternating direction (ADI) technique developed by Peaceman-Rachford (Ref.7).

To apply the ADI method to the energy equation (29) requires the coefficient velocities U and V to be held constant at any grid point over a time step. Now the partial derivatives in Equation (29) can be approximated by finite-differences. (Note: All space derivatives are central differences.)

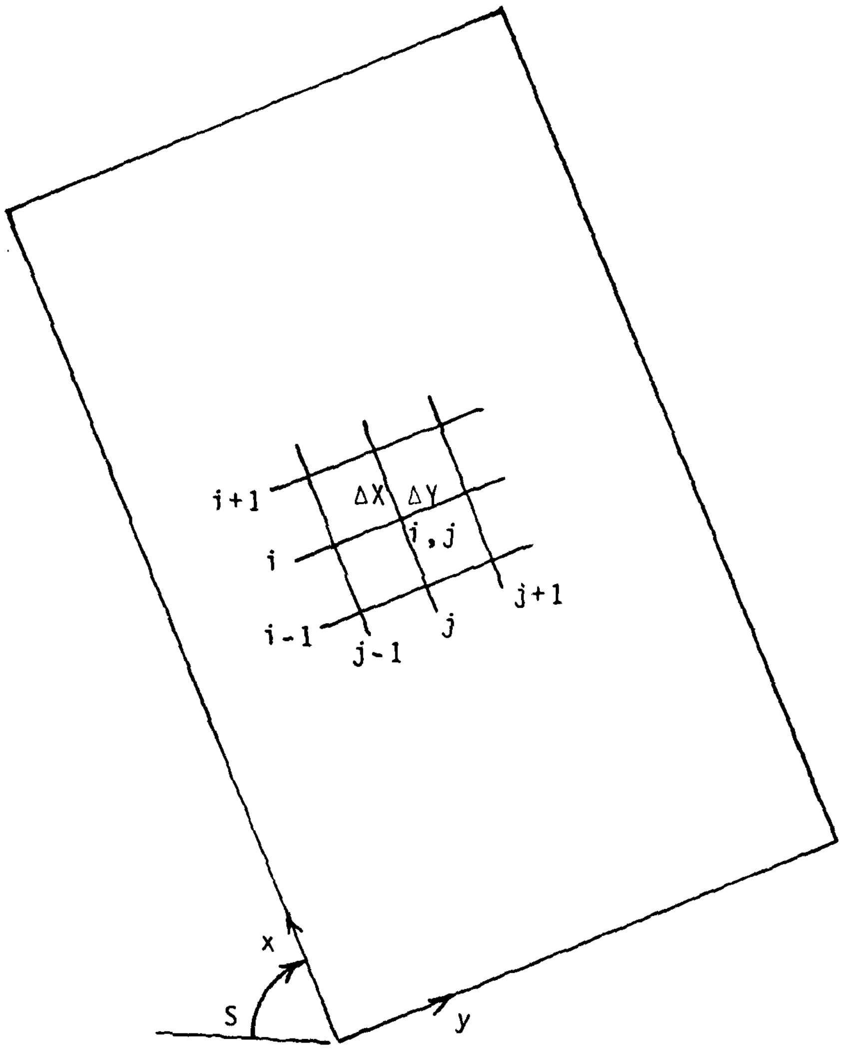


Figure 4. Inclined Rectangular Enclosure Finite-Difference Grid Representation

$\underline{\tau=(n+\frac{1}{2})\Delta\tau}$

$$\frac{\Theta^{n+\frac{1}{2}}_{i,j} - \Theta^{n}_{i,j}}{\Delta \frac{\tau}{2}} + U^{n}_{i,j}\left[\frac{\Theta^{n+\frac{1}{2}}_{i+1,j} - \Theta^{n+\frac{1}{2}}_{i-1,j}}{2\Delta X}\right] + V^{n}_{i,j}\left[\frac{\Theta^{n}_{i,j+1} - \Theta^{n}_{i,j-1}}{2\Delta Y}\right]$$

$$= \frac{1}{P_r}\left[\frac{\Theta^{n+\frac{1}{2}}_{i+1,j} - 2\,\Theta^{n+\frac{1}{2}}_{i,j} + \Theta^{n+\frac{1}{2}}_{i-1,j}}{\Delta X^2}\right] + \frac{1}{P_r}\left[\frac{\Theta^{n}_{i,j+1} - 2\Theta^{n}_{i,g} + \Theta^{n}_{i,j-1}}{\Delta Y^2}\right] \quad (43)$$

$\underline{\tau=(n+1)\Delta\tau}$

$$\frac{\Theta^{n+1}_{i,j} - \Theta^{n+\frac{1}{2}}_{i,j}}{\Delta\tau/2} + U^{n}_{i,j}\left[\frac{\Theta^{n+\frac{1}{2}}_{i+1,j} - \Theta^{n+\frac{1}{2}}_{i-1,j}}{2\Delta X}\right] + V^{n}_{i,j}\left[\frac{\Theta^{n+1}_{i,j+1} - \Theta^{n+1}_{i,j-1}}{2\Delta Y}\right]$$

$$= \frac{1}{P_r}\left[\frac{\Theta^{n+\frac{1}{2}}_{i+1,j} - 2\Theta^{n+\frac{1}{2}}_{i,j} + \Theta^{n+\frac{1}{2}}_{i-1,j}}{\Delta X^2}\right] + \frac{1}{P_r}\left[\frac{\Theta^{n+1}_{i,j+1} - 2\Theta^{n+1}_{i,j} + \Theta^{n+1}_{i,j-1}}{\Delta Y^2}\right] \quad (44)$$

Equations (43) and (44) are implicit in X and Y directions, respectively, and when applied to every point in a row or column, as the case may be, yield a tridiagonal matrix in the unknown temperatures $\Theta^{n+\frac{1}{2}}_{i,j}$ or $\Theta^{n+1}_{i,j}$. In this study, a tridiagonal algorithm adapted by Roache (Ref.7) was used to solve the tridiagonal matrices.

The vorticity equation (28), which has characteristics similar to the energy equation, can now be solved by the ADI method for new values of vorticity at the n inner grid points. The vorticity at the boundary points will remain at their old $n\Delta\tau$ values during this calculation, and will be advanced later. Because there is a temperature gradient in the vorticity equation (28), the new $(n+1)\Delta\tau$ temperature field will be used to evaluate the temperature gradient at each grid point. The temperature gradient will then be held constant throughout the calculation. The finite difference representations of the vorticity are

$\underline{\tau=(n+\frac{1}{2})\Delta\tau}$

$$\frac{\zeta_{i,j}^{n+\frac{1}{2}} - \zeta_{i,j}^{n}}{\Delta\tau/2} + U_{i,j}^{n}\left[\frac{\zeta_{i+1,j}^{n+\frac{1}{2}} - \zeta_{i-1,j}^{n+\frac{1}{2}}}{2\Delta X}\right] + V_{i,j}^{n}\left[\frac{\zeta_{i,j+1}^{n} - \zeta_{i,j-1}^{n}}{2\Delta Y}\right]$$

$$= G_R\left[\sin(S)\left(\frac{\Theta_{i,j+1}^{n+1} - \Theta_{i,j-1}^{n+1}}{2\Delta Y}\right) - \cos(S)\left(\frac{\Theta_{i+1,j}^{n+1} - \Theta_{i-1,j}^{n+1}}{2\Delta X}\right)\right]$$

$$+\left[\frac{\zeta_{i+1,j}^{n+\frac{1}{2}} - 2\zeta_{i,j}^{n+\frac{1}{2}} + \zeta_{i-1,j}^{n+\frac{1}{2}}}{\Delta X^2}\right] + \left[\frac{\zeta_{i,j+1}^{n} - 2\zeta_{i,j}^{n} + \zeta_{i,j-1}^{n}}{\Delta Y^2}\right] \qquad (45)$$

$\underline{\tau=(n+1)\Delta\tau}$

$$\frac{\zeta_{i,j}^{n+1}-\zeta_{i,j}^{n+\frac{1}{2}}}{\Delta\tau/2}+U_{i,j}^{n}\left[\frac{\zeta_{i+1,j}^{n+\frac{1}{2}}-\zeta_{i-1,j}^{n+\frac{1}{2}}}{2\Delta X}\right]+V_{i,j}^{n}\left[\frac{\zeta_{i,j+1}^{n+1}-\zeta_{i,j-1}^{n+1}}{2\Delta Y}\right]$$

$$=G_R\left[\sin(S)\left(\frac{\Theta_{i,j+1}^{n+1}-\Theta_{i,j-1}^{n+1}}{2\Delta Y}\right)-\cos(S)\left(\frac{\Theta_{i+1,j}^{n+1}-\Theta_{i-1,j}^{n+1}}{2\Delta X}\right)\right]$$

$$+\left[\frac{\zeta_{i+1,j}^{n+\frac{1}{2}}-2\zeta_{i,j}^{n+\frac{1}{2}}+\zeta_{i-1,j}^{n+\frac{1}{2}}}{\Delta X^2}\right]+\left[\frac{\zeta_{i,j+1}^{n+1}-2\zeta_{i,j}^{n+1}+\zeta_{i,j-1}^{n+1}}{\Delta Y^2}\right] \qquad (46)$$

Equations (45) and (46) are also implicit in X and Y directions, respectively, and when applied to every point in a row or column, as the case may be, also yield a tridiagonal matrix. These matrices are solved by the same algorithm that was used for solving the energy equation.

With the new interior vorticity field calculated, the method of successive over-relaxation is used to solve the stream function equation for the new stream function field. The expression for the $m+1$ iteration of the stream function at a point is as follows:

$$\Psi_{i,j}^{(m+1)}=\Psi_{i,j}^{(m)}+\frac{\omega}{4}\left[(\Delta X)^2\zeta_{i,j}^{n+1}+\Psi_{i-1,j}^{(m)}+\Psi_{i+1,j}^{(m)}+\Psi_{i,j-1}^{(m)}+\Psi_{i,j+1}^{(m)}-4\Psi_{i,j}^{(m)}\right] \qquad (47)$$

TABLE I			
OPTIMUM VALUES OF RELAXATION PARAMETER FOR USE IN EQ.(47) (Ref.4)			
No. of grid spacings:			
X Direction	Y Direction	$\Delta X(=\Delta Y)$	ω_{OPT}
10	10	0.2	1.58
20	20	0.05	1.83
20	10	0.1	1.70
30	10	0.1	1.73

Wilkes and Churchill (Ref.4) determined optimum values of relaxation parameters for some representative grid sizes as seen in Table I. Using these values, Wilkes and Churchill experienced good convergence at each grid point with about twenty five iterations. In this study twenty five iterations were performed for each grid point using the optimum value of the relaxation parameters in Table I.

The new wall vorticities can now be determined from the new stream function field. In this study, two wall vorticity expressions were used depending upon the Grashof Number of the fluid. Wilkes and Churchill results indicated the first expression will cause instabilities above G_R=200,000.

$$\zeta_{i,0}^{n+1} = -\frac{8\Psi_{i,2}^{n+1} - \Psi_{i,2}^{n+1}}{2(\Delta Y)^2} \quad \text{when } G_R \le 200{,}000 \tag{48}$$

$$2) \quad \zeta_{i,0}^{n+1} = -\frac{2\Psi_{i,1}^{n+1}}{(\Delta Y)^2} \quad \text{when } G_R > 200{,}000 \tag{49}$$

Both expressions were derived by Wilkes and Churchill (Ref.4)

The new velocity fields U and V can now be calculated from the new stream function field. In this study second order space central finite difference approximations were used for the new velocities.

$$U_{i,j}^{n+1} = \frac{\Psi_{i,j+1} - 2\Psi_{i,j} + \Psi_{i,j-1}}{2\Delta Y} \tag{50}$$

$$V_{i,j}^{n+1} = -\frac{\Psi_{i+1,j} - 2\Psi_{i,j} + \Psi_{i-1,j}}{2\Delta X} \tag{51}$$

These expressions differ from those of Wilkes and Churchill who used a fourth order finite difference approximation.

Now using the newest temperature field, the local and average Nusselt Numbers can also be calculated. The derivative in Equation (39) is expressed by a second order forward finite difference relation.

$$Nu = \frac{\Theta_{i,3} - 4\Theta_{i,2} + 3\Theta_{i,1}}{2\Delta Y} \tag{52}$$

The average Nusselt Number is obtained by integration using the Trapezoidal rule.

Partitioned Cavity

In the previous chapter, the mathematical model of natural convection in an inclined rectangular enclosure with partitions was developed. The governing equations were determined to be unchanged. However, new boundary conditions were necessary to describe the partitioned enclosure. In this section, the numerical procedure previously developed will be modified to account for 45 degree partitions. And finally, a method of solution of a partitioned rectangular enclosure will be discussed.

Numerical Procedure Modification. The two significant modifications made to the inclined rectangular enclosure numerical procedure were; 1) a new wall vorticity model for the 45 degree partition, and 2) local heat transfer coefficients for the partition. While the wall vorticity models used in the rectangular enclosure are accurate for grid points along the mesh, the following expression (see Ref.7 page 144) provides a more accurate evaluation of the wall vorticity of a sloping wall (ϕ=45 degrees) with $\Delta X=\Delta Y$.

$$\zeta_{iw,jw} = \frac{2(\Psi_{iw-1,jw}+\Psi_{iw,jw+1}-2\Psi_{iw,jw})}{\Delta X} \qquad (53)$$

This expression was used to evaluate the wall vorticity of all partitions.

Finally, the local heat transfer equation was modified to determine the heat transfer coefficient across a partition. First and second order finite difference schemes were sufficient for all grid points along the partition except the end points. At the end points, the heat transfer coefficient across the partition was determined to be equal to the heat transferred along the partition. Details of the heat transfer coefficients expressions can be found in Appendix B.

Solution Method. Since the natural convection governing equations did not change with the addition of partitions, each triangular cell can be independently solved by assuming common boundary conditions for the neighboring triangles. However, it is then necessary to compute an energy balance at each partition grid point. If the partition boundary conditions do not balance, it is necessary to iterate the triangular cell partition boundary conditions until the heat transfer coefficients and temperatures match. This iterative procedure, however, was simplified in this study by the thermal characteristic of the partition model.

In this study, the partition model steady state temperature distribution was linear, and the partition was a pure conductor across the partition. Because of these thermal characteristics, the temperature and heat transfer coefficients at each partition grid point always match. Therefore, it was decided to build a composite model of the rectangular enclosure with partitions.

The composite model was composed of individual triangular cells which were each solved independently. Then the solutions were added together to obtain the final natural convection heat transfer results. A complete computer program listing is in Appendix B.

IV. Discussion of Results

The discussion of results centers on two areas: determining the validity and accuracy of the numerical procedure and computer code; and determining the effect of partitioning an inclined rectangular enclosure into triangular regions on natural convection heat transfer.

The computations were conducted (using a CDC 6600 computer) for air cavities (P_r=.7333) with aspect ratios from 1.0 to 3.0 at heat fluxes yielding G_R values up to $2x10^5$.

Rectangular Enclosure Numerical Results

To determine the validity and accuracy of the numerical procedure, transient and steady state heat transfer results were obtained for rectangular enclosures at various inclination angles, aspect ratios, side wall boundary conditions and Grashof numbers. These results were then compared with previous numerical results obtained by Wilkes and Churchill (Ref.4), Koutsoheras (Ref.5), and Ozoe, et.al. (Ref.6). A complete summary of the computer runs is presented in Appendix A.

Vertical Rectangular Enclosure. The accuracy of the heat transfer solution was determined by comparing the steady state numerical solution of the rectangular enclosure in the vertical position ($S=90^0$)

TABLE II					
Comparison of Steady State Mean Nusselt Numbers					
Grashof Number (G_R)	Aspect Ratio (A)	Side wall Boundary Condition	Mean Nusselt Number ($\overline{Nu}$)	Wilkes & Churchill Mean Nusselt no. $\overline{Nu}$	Percent Diff.
20000	2.0	Insulated	3.036	2.874	5.60
20000	2.0	Insulated	3.141	2.992	4.98
20000	3.0	Insulated	2.954	2.825	4.56
60000	1.0	Insulated	4.905	4.793	2.34
100000	1.0	Insulated	6.136	5.512	11.30
20000	1.0	Linear	2.061	2.068	0.34

with heat transfer results attained by Wilkes and Churchill (Ref.4). A comparison of the two results is shown in Table II.

The results in Table II show that in each comparison, the numerical procedure developed in this study overestimates by five to 12 percent the mean Nusselt numbers achieved by Wilkes and Churchill. This difference in results could be caused by one or both of two differences in the numerical procedures.

In this investigation, the velocity field was calculated from the new stream functions with a second order finite difference representation. On the other hand, Wilkes and Churchill used a fourth order finite difference model to calculate the velocity field. The higher order model yields more accurate velocities per time step. But, since this investigation was primarily interested in the steady state solution, the improved accuracy of the transient solution velocity field was not necessary. So it was decided to use the less complicated second order finite difference representation.

A second difference between the numerical methods could be the finite difference representation used to determine the local Nusselt numbers at the hot wall. The wall heat flux approximation used by Wilkes and Churchill (Ref.4) was not specified in their report. So, it was decided to calculate the local heat flux with a second order, forward difference approximation in this study. The second order approximation will provide sufficient accuracy for comparison and trend information, but may represent the difference observed in the heat transfer results.

One further comparison of the two numerical procedures was to compare the transient heat transfer results. Figure 5 is a plot of this study's mean Nusselt numbers vs. nondimensional time, starting from the initial condition to steady state. The shape and indicated trends of the transient solution curve is identical to the plot achieved by Wilkes and Churchill (Ref.4). A conclusion that can be deduced from this result is that eventhough the two numerical

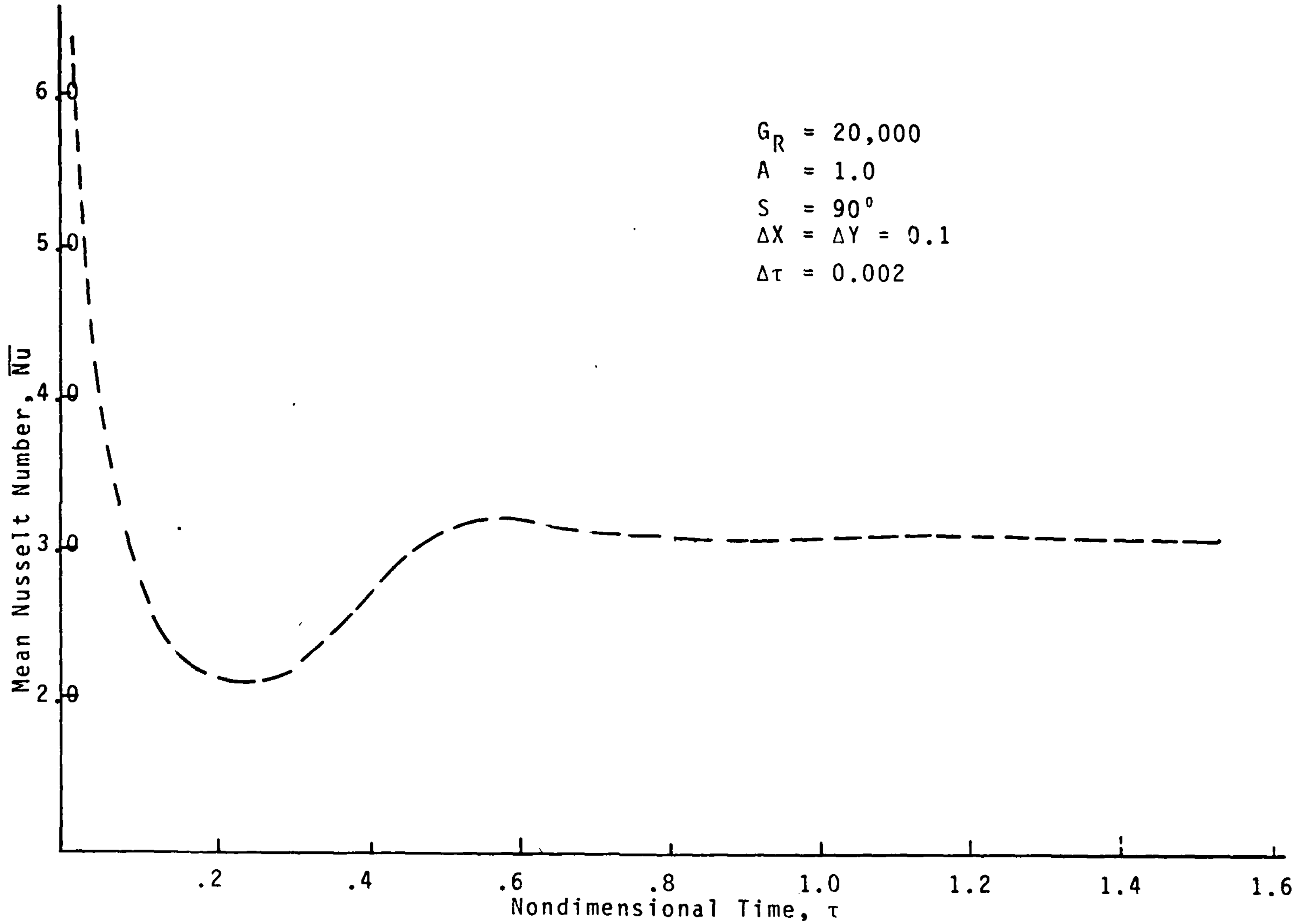


Figure 5. Transient Solution for the Vertical Square Enclosure

procedures produce slightly different values for mean Nusselt numbers, the difference is constant throughout the solution. Therefore, the numerical procedure developed in this study is an adequate numerical model of transient, as well as steady-state, natural convection in a vertical rectangular enclosure.

Inclined Rectangular Enclosure. The previous discussion was limited to determining the adequacy of this study's numerical procedure for the vertical rectangular enclosure. Now it is necessary to analyze the results of the numerical procedure and computer code over the total planned operating range of each of the variables. Figures (6), (7), and (8) illustrate the effects of inclination angle, aspect ratio, Grashof number, and side wall boundary conditions on heat transfer. The following are conclusions which can be drawn from the figures.

I. Effect of Side Wall Boundary Conditions. Figures (6) and (7) show that the heat transfer at the hot wall is a strong function of the side wall boundary conditions. The heat transferred at the hot wall decreases approximately 35% when the side walls are changed from an insulated boundary to a pure conductor across the boundary (linear temperature distribution along the side wall). The effect of the side wall boundary condition also appears to be independent of Grashof number and inclination angle. The results of this

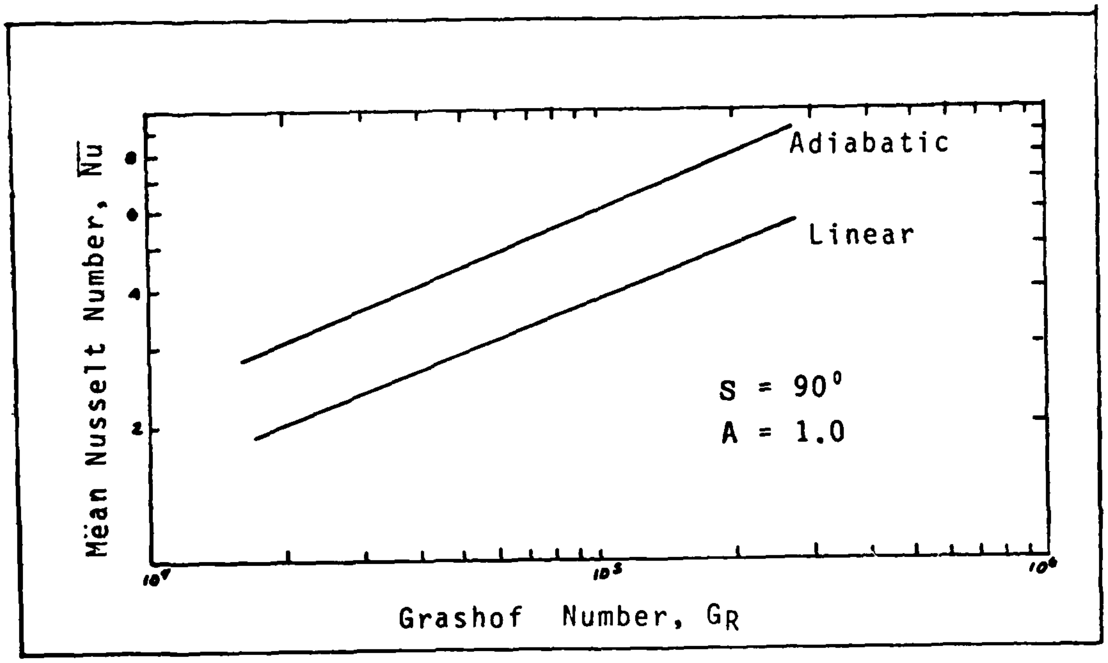


Figure 6. Effect of Side Wall Boundary Conditions on Heat Transfer.

effect are in excellent agreement with those reported by Koutsoheras (Ref.5).

II. Effect of Inclination Angle. The effect of inclination angle is best illustrated by Figure 7. The highest heat transfer takes place when the enclosure is inclined at 60^0 from the horizontal. However, the total change in heat transfer due to a change in inclination angle is insignificant when compared to side wall boundary conditions and aspect ratio changes. This result was previously observed by Koutsoheras (Ref.5) and Ozoe (Ref.6).

III. Effect of Aspect Ratio. The effect of aspect ratio on heat transfer for the case of moderate aspect ratio enclosures with insulated side walls is illustrated in Figure 8.

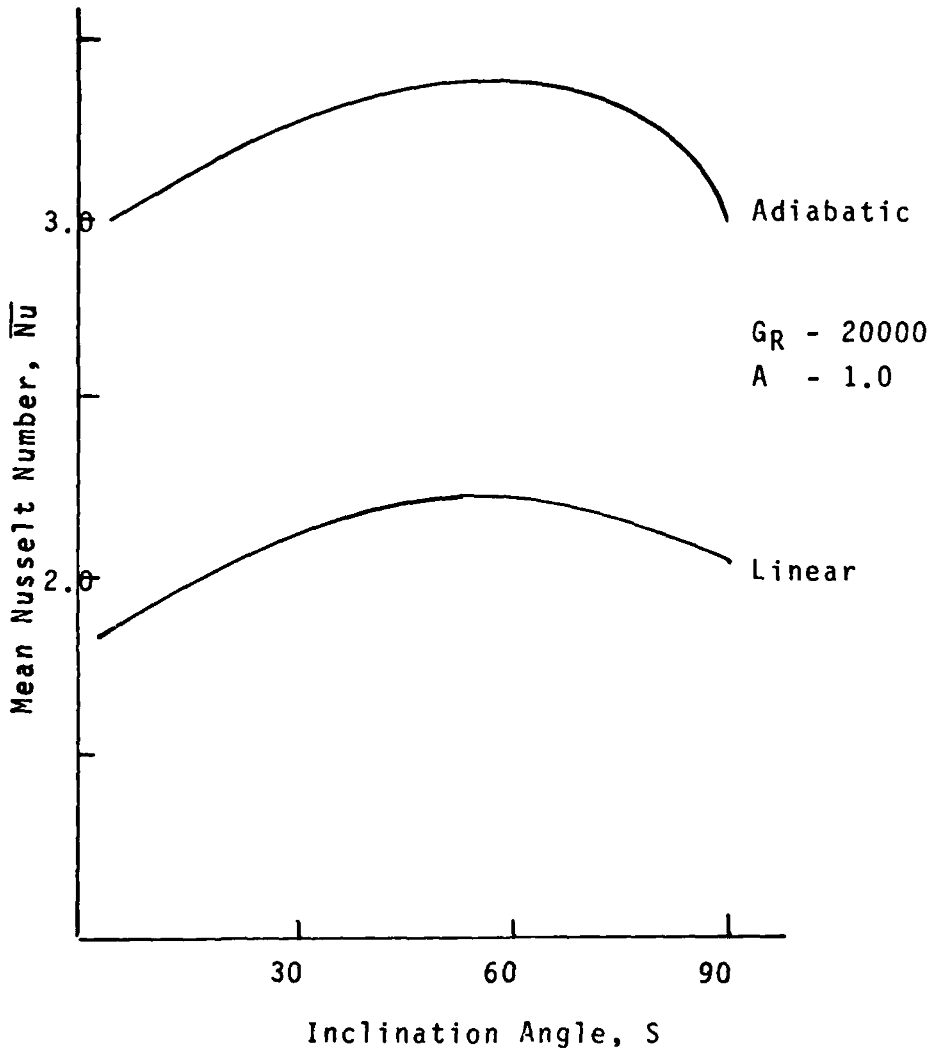


Figure 7. Effect of Inclination Angle on Heat Transfer for Two Side Wall Boundary Conditions

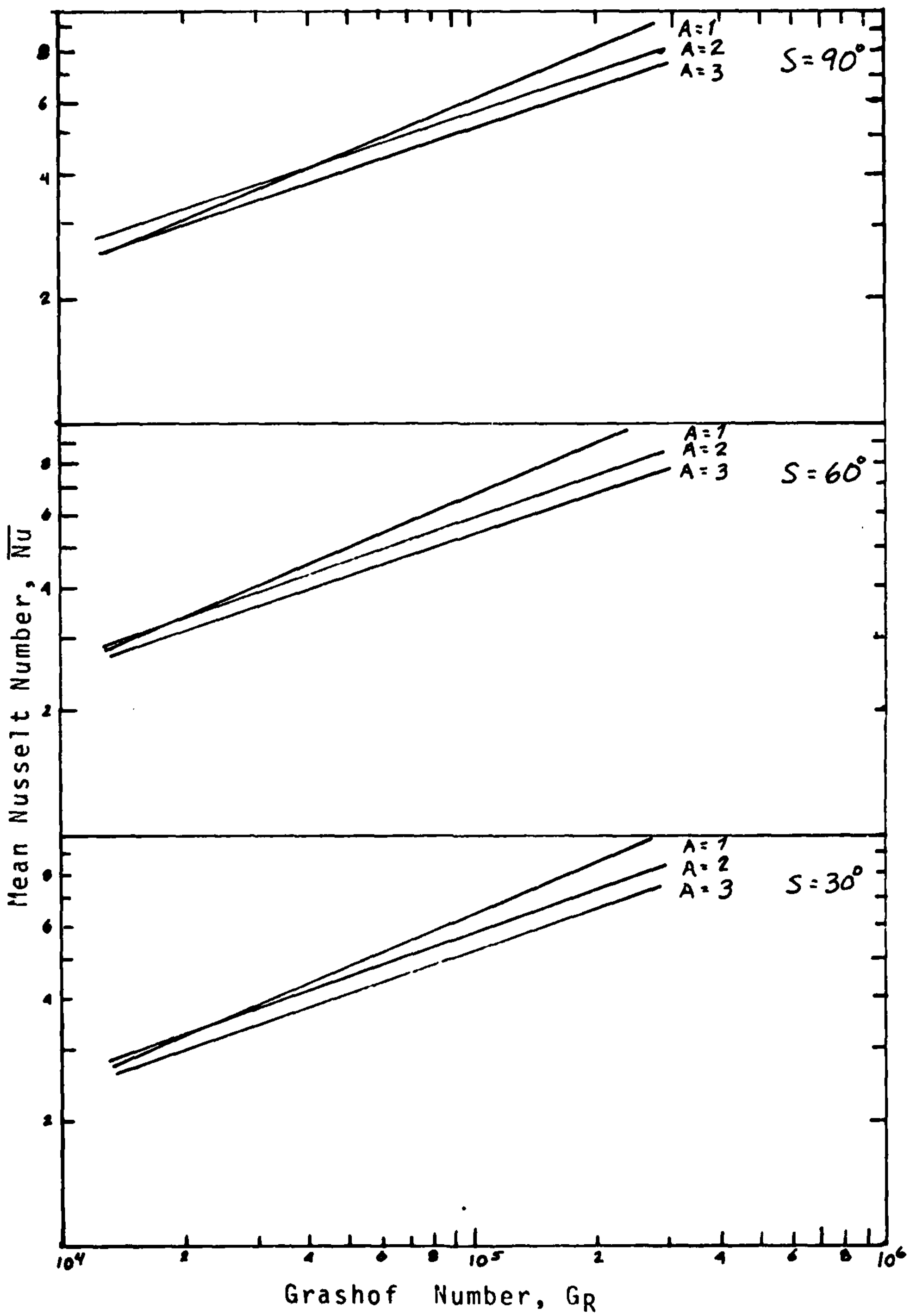


Figure 8. Effect of Aspect Ratio on Heat Transfer at Various Inclination Angles

This figure shows that changes in Grashof number and inclination angle have a stronger influence on heat transfer in a rectangular enclosure where A=1.0 than in enclosures with A=2.0 or 3.0. This is due to the side wall disturbances in a square enclosure occupying a larger proportion of the enclosure than in higher aspect ratio cavities.

In summary, the numerical results for the inclined moderate aspect ratio rectangular enclosure indicate the validity of the numerical procedure and computer code.

Effect of Partitions

In this section, the validity of natural convection heat transfer results for composite rectangular enclosures will be determined. Then, the heat transfer results for the partitioned enclosure will be compared to heat transfer results of rectangular enclosures without partitions. From these comparisons the effectiveness of partitions in reducing natural convection heat transfer across and enclosure will be resolved.

Composite Enclosures. To determine the validity of composite heat transfer solutions, two partitioned rectangular enclosures were investigated. These enclosures are shown in Figure (9). The square enclosure is composed of two triangular cells while the rectangular enclosure (A=2) is composed of three triangular cells. Natural convection heat transfer coefficients were evaluated for each of

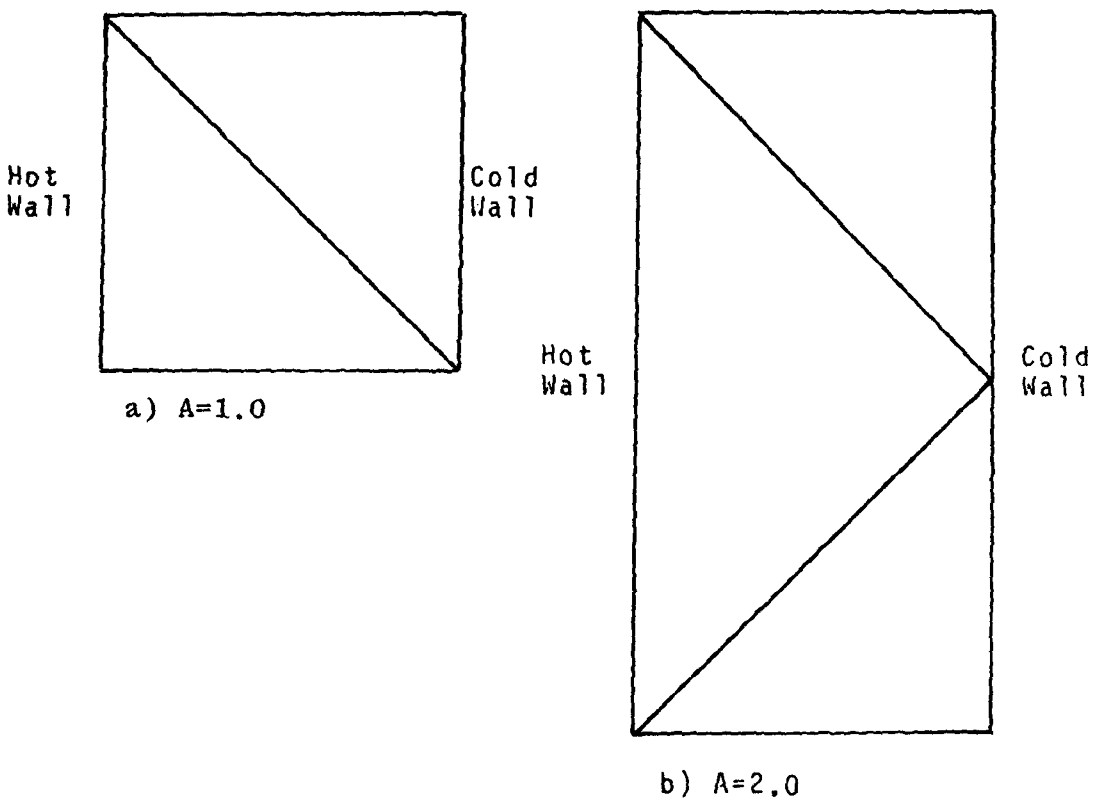


Figure 9. Partitioned Rectangular Enclosures

these triangular cavities. The heat transfer coefficients of adjoining triangular cells were then compared at the common partition boundary.

In the square enclosures, the total heat transfer across the partition was balanced, however, heat conduction along the partition was required to maintain the partition boundary linear temperature distribution. On the other hand, the total heat transfer across the partitions in the rectangular enclosure (A=2.0) did not balance, eventhough the total heat transfer at the hot and cold walls were balanced. These results indicate valid composite heat transfer solutions exist only for square enclosures with a diagonal partition. An iterative numerical procedure is required for other than square enclosure.

<u>Partitioned Square Enclosures</u>. To determine the effectiveness of diagonal partitions in suppressing natural convection, heat transfer coefficients were evaluated for three square enclosures with adiabatic side walls at various inclination angles, and Grashof numbers. The results are shown in Figure 10.

From this figure, it was concluded that a diagonal partition in a square enclosure reduces natural convection heat transfer by more than 50%. An explanation for the substantial decrease in natural convection heat transfer across the square enclosure is that the diagonal partition increases

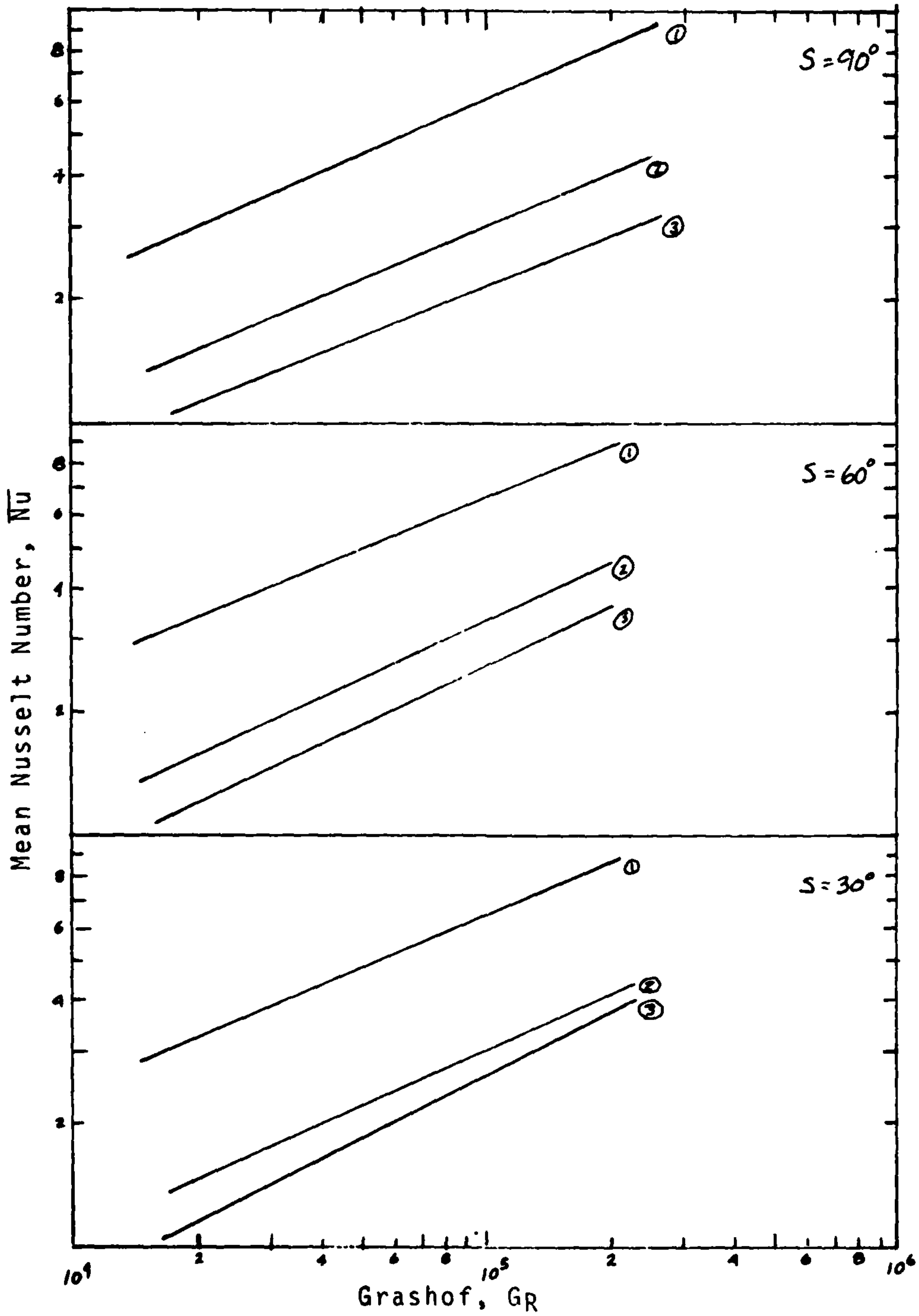


Figure 10. The Effect of a 45° Diagonal Partition in an Inclined Square Enclosure on the Rate of Heat Transfer

the wetted surface area by 70%. And, when air flows over this surface, additional friction forces are generated which suppress the natural convection in the enclosure.

Figure 10 also illustrates the effect of partition angle on natural convection heat transfer. The partition oriented at ϕ=-45 degrees to the hot wall provided an additional 15% decrease in natural convection heat transfer when compared to the partition at ϕ=45 degrees . To provide an expanation for this further decrease in heat transfer, the steady state temperature distributions for the three square enclosures were examined in Figures 11, 12 and 13.

From these figures, it was concluded that the partition angle affects the thickness of the thermal boundary layer at the hot wall. The partition oriented at ϕ=-45 degrees resulted in the greater reduction in heat transfer because it produced the thicker thermal boundary layer on the lower half of the hot wall (the region of highest heat transfer), as evidenced by the increased spacing between isotherms.

In summary, the composite heat transfer solution was validated for square enclosures with partitions. Then, a diagonal partition in an inclined enclosure was determined to reduce natural convection heat transfer by more than 50%.

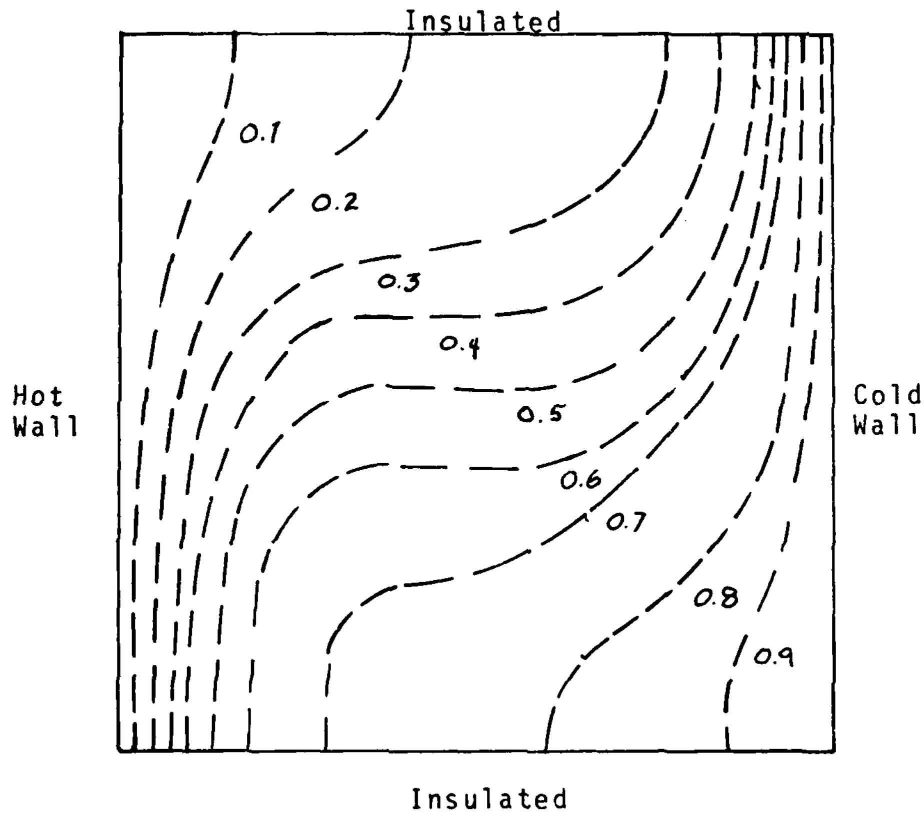


Figure 11. Steady State Isotherms in a Vertical Square Enclosure.

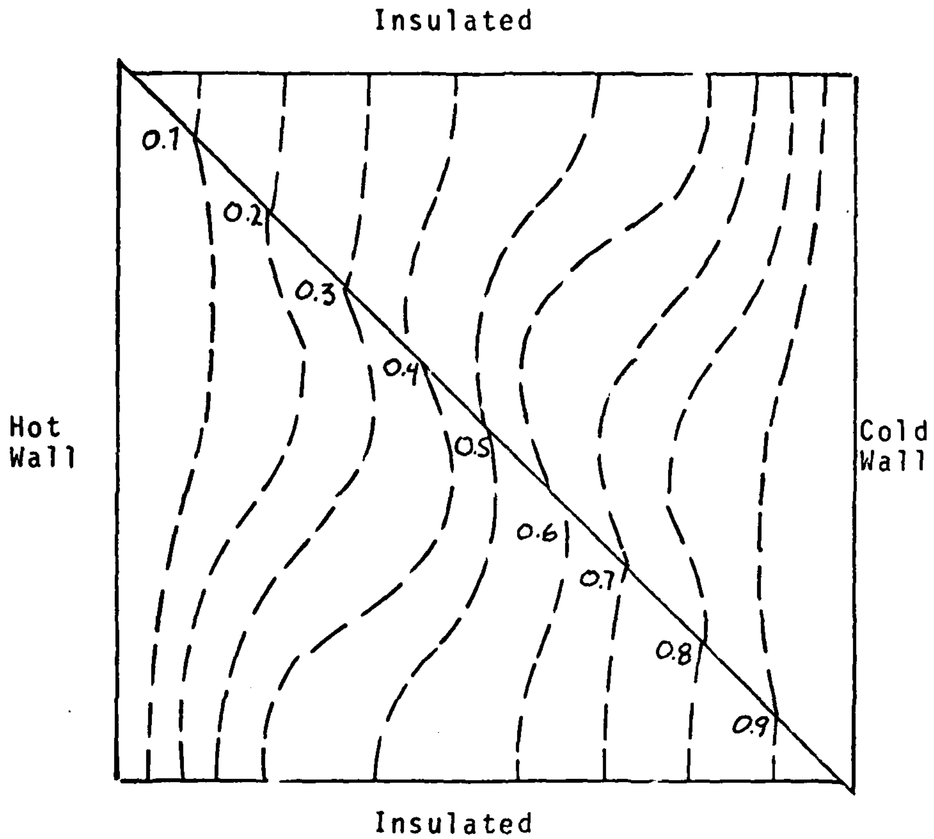


Figure 12. Steady State Isotherms in a Vertical Square Enclosure with a Diagonal Partition ϕ=45 degrees.

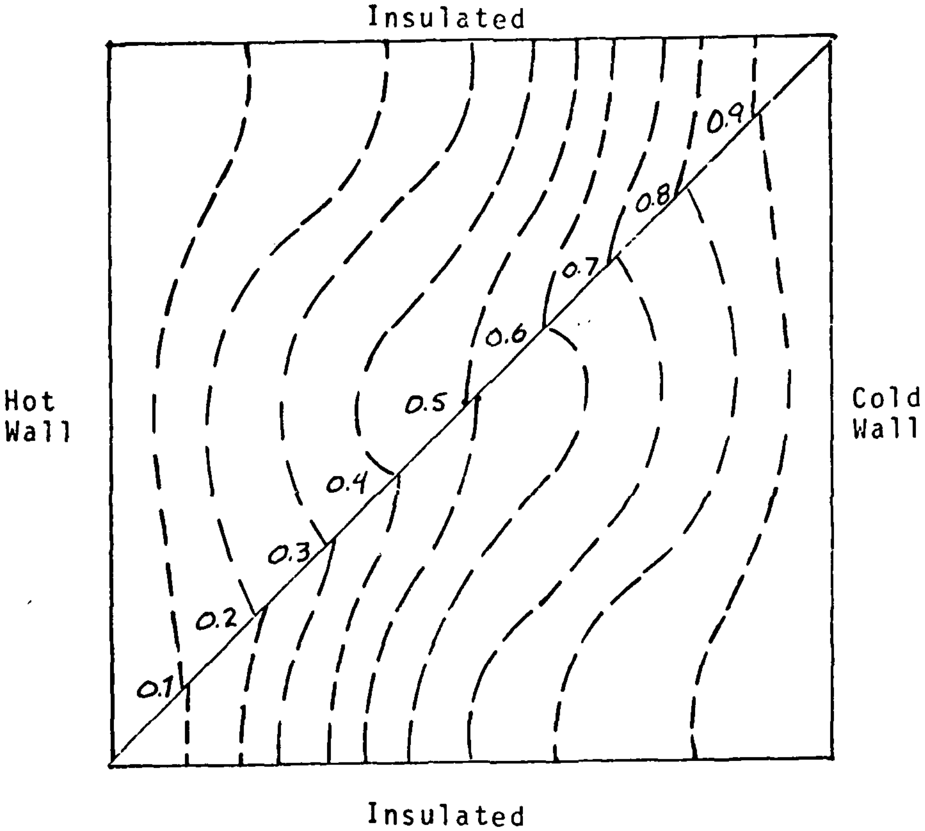


Figure 13. Steady State Isotherms in a Vertical Square Enclosure with a Diagonal Partition $\phi=-45$ degrees.

V. Conclusions

Conclusions

The results of this study provide the following conclusions:

1. The finite difference numerical procedure yields valid transient and steady state, natural convection heat transfer coefficients for air filled, inclined, moderate aspect ratio, rectangular enclosures. The procedure is limited to $G_R<2.0\times10^5$, and $10\le S\le90$.

2. A thin diagonal partition in an inclined square enclosure with insulated side walls reduces natural convection across the enclosure by more than 50%.

VI. Recommendations

Recommendations

The following recommendations are suggested for follow on investigations:

1. Numerical techniques should be developed which would extend the present procedure's Grashof number limitation of 2.0×10^5 .

2. A numerical procedure should be developed to evaluate natural convection heat transfer in an inclined rectangular enclosure with more than one partition. One approach to this procedure would be to assume an initial partition temperature distribution, compute the flow field in each partition cell, then compare the resulting heat transfer coefficients at each partition grid point. If the heat transfer coefficients do not balance, select a new partition temperature distribution, and repeat the process until the heat transfer coefficients balance at each partition grid point. This procedure preserves the computational efficiency obtained with tridiagonal matrices.

3. The effects of radiation heat transfer within the enclosure on partition, and side wall thermal boundary conditions should be investigated.

Bibliography

1. Buchberg, H., Catton, I., and Edwards, D.K., "Natural Convection in Enclosed Spaces-A Review of Application to Solar Energy Collection," Journal of Heat Transfer, May 1976, pp.182-188.

2. Meyer, B.A., Mitchell, J.W., and El-Wakil, M.M., "Natural Convection Heat Transfer in Moderate Aspect Ratio Enclosures," Journal of Heat Transfer, November 1979, pp.655-659.

3. Hollands, K.G.T., "Studies of Methods of Reducing Heat Losses from Flat-Plate Solar Collectors," ERDA Annual Progress Report, COO-2597-3, University of Waterloo, Waterloo, Onterio, Canada, 1977.

4. Wilkes, J.O. and Churchill, S.W., "The Finite Difference Computation of Natural Convection in a Rectangular Enclosure," A.I.Ch.E.J. Vol.12,Nov 1,1966,pp161-166.

5. Koutsoheras, W., Charters, W.W.S. "Natural Convection Phenomena in Inclined Cells with Finite Side Walls-Numerical Solution," Solar Energy,Vol.19,May 1977, pp.433-438.

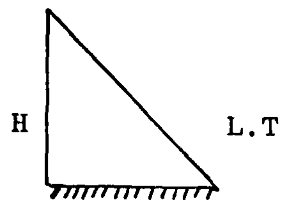
6. Ozoe, H., Sayama, H. and Churchill, S.W., "Natural Convection in an Inclined Square Channel," Inter. J. of Heat and Mass Transfer, Vol.17,Feb. 1974,pp.401-406.

7. Roache, P.J., Computational Fluid Dynamics, Albuquerque; Hermosa Publishers, 1976.

Appendix A

Numerical Data

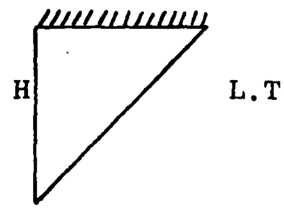
Triangular Cell with $\phi = 45^0$



DX = DY = 0.1 $D\tau$ = 0.002

G_R	AR	S	$\overline{Nu}$
20000	1.0	90	1.5088
		60	1.5885
	1.0	30	1.4610
60000	1.0	90	
		60	2.7108
		30	2.5536
100000	1.0	90	
		60	3.389
		30	2.5536
200000	1.0	90	
		60	4.5353
		30	3.7664

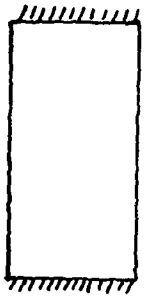
Triangular Cell with $\phi = 45^{0}$



DX = DY = 0.1 DZ = 0.002

G_R	AR	S	$\overline{Nu}$
20000	1.0	90	1.1546
		60	1.21657
		30	1.17399
60000	1.0	90	1.6820
		60	1.96207
		30	1.9933
100000	1.0	90	2.1197
		60	
		30	2.61767
200000	1.0	90	2.9630
		60	3.6636
		30	3.7764

Rectangular Cavity



DX = DY = 0.1 DT = .002

G_R	AR	S	$\overline{Nu}$
20000	1.0	90	3.036
		60	3.395
		30	3.271
		90	2.874
20000	2.0	90	3.141
		60	3.367
		30	3.244
		90	2.992
20000	3.0	90	2.954
		60	3.081
		30	
		90	2.825
60000	1.0	90	4.905
		60	5.349
		30	5.0615
		90	4.793
60000	2.0	90	4.720
		60	5.010
		30	4.8642
60000	3.0	90	4.390
		60	4.6197
		30	4.3779
100000	1.0	90	6.136
		60	6.699
		30	6.2863
		90	5.512

G_R	AR	S	$\overline{Nu}$
100000	2.0	90	6.136
		60	6.0683
		30	5.8112
100000	3.0	90	5.271
		60	5.5385
		30	5.2912
200000	1.0	90	7.9696
	2.0	90	7.0052
	3.0	90	6.4187
200000	1.0	30	8.6576
	2.0	30	7.3584
	3.0	30	6.6134
200000	1.0	60	8.8755
	2.0	60	7.5066
	3.0	60	6.7237

Appendix B

Computer Code Listing

I. Inclined Rectangular Enclosure with Insulated Sidewalls.

II. Inclined Triangular Enclosure ϕ=45 degrees, Insulated Side Wall.

III. Inclined Triangular Enclosure ϕ=45 degrees, Insulated Side Wall.

IV. Inclined Triangular Enclosure Two Partition Walls.

V. Tridiagonal Algorithm Subroutine.

Computer Code Notation

AR	Aspect Ratio
DT	Time Increment
DX	Grid Spacing X-direction
DY	Grid Spacing Y-direction
GR	Grashof Number
IX	Number Grid Points in X-direction
IY	Number Grid Points in Y-direction
PR	Prandil Number
S	Inclination Angle (Degrees)
S1	Inclination Angle (Radians)
ST	Stream Function
T	Dimensionless Temperature
U	Dimensionless Velocity in X-direction

V	Dimensionless Velocity in Y direction
VOR	Vorticity
NUC	Local Nusselt Number-Cold Wall
NUH	Local Nusselt Number-Hot Wall

```
      PROGRAM BASIC(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DIMENSION ST(31,31),VOR(31,31),T(31,31),U(31,31),V(31,31),
     1WK(31,31),A(31),B(31),C(31),D(31),E(31),F(31),W(31)
      REAL NUC,NUH,L
  900 FORMAT(1X,11(F11.5))
  901 FORMAT(2X,F11.5, 5X, F11.6)
C
C     INPUT INITIAL,BOUNDARY CONDITIONS
C
   10 TIME=0.0
      READ *,PR,GR,DX,DY,DT,AR,S1,Z
      IF(EOF(5LINPUT) .NE. 0) GO TO 999
      S=S1/57.2957795
      AR1 =AR/DX
      IX = AR1 + 1.00001
      AR2 = 1. /DY
      IY = AR2 + 1.00001
C
      PRINT*, "      "
      PRINT*, "      "
      PRINT*,"PRANTL NUMBER = ",PR
      PRINT*, "GRASHOFF NUMBER = ",GR
      PRINT*, " ASPECT RATIO =  ",AR
      PRINT*, "DELTA X = ",DX, "DELTA Y = ",DY
      PRINT*, "NUMBER OF INCREMENTS  = ",IX
      PRINT*, "Y-INCREMENTS =  ",IY
      PRINT*, "TIME INCREMENT = ",DT
      PRINT*, "INCLINATION ANGLE = ", S1
      PRINT*, "INCLINATION ANGLE = ", S
      PRINT*, "      "
      PRINT*, "      "
C
      DO 6 I=1,IX
      DO 7 J=1,IY
      VOR(I,J)=0.0
      ST(I,J)=0.
      U(I,J) = 0.
      V(I,J) = 0.0
      T(I,J) = 0.0
      T(I,1)= 0.
      T(I,IY)=1.0
    7 CONTINUE
    6 CONTINUE
      IF(Z.EQ.1.0) GO TO 105
      PRINT*, "TIME = ",TIME
      PRINT*, "STREAM FUNCTION"
      WRITE(6,900) ((ST(I,J),J=1,IY),I=1,IX)
      PRINT*, "VORTICITY  "
      WRITE(6,900) ((VOR(I,J),J=1,IY),I=1,IX)
      PRINT*, " U-VELOCITY  "
      WRITE(6,900) ((U(I,J),J=1,IY),I=1,IX)
      PRINT*, " V-VELOCITY  "
      WRITE(6,900) ((V(I,J),J=1,IY),I=1,IX)
      PRINT*, " TEMPERATURE  "
      WRITE(6,900) ((T(I,J),J=1,IY),I=1,IX)
C
C     SOLVE ENERGY EQUATION FOR TIME=N+1/2
```

```
C       TEMPERATURE SOLUTION IN X-DIRECTION (INTERIOR POINTS)
C
C
 10     TIMEA= TIME + 0.5*DT
        IXA=IX-1
        IYA=IY-1
        A2 = DT/ (PR*(DY**2.))
        A3  = DT /(2.*DX)
        A4  =DT /(2.*DY)
        A5 = 2.0
        A6 = DT/(PR*(DX**2.))
        DO 2   J=2,IYA
        DO 21  I=2,IXA
        A(I) =A6 -(U(I,J)*A3)
        B(I) =A5 +(2.*A6)
        C(I) =(U(I,J)*A3) +A6
        D(I) =((A2-(V(I,J)*A4))*T(I,J+1)) +((A5-(2.*A2))*T(I,J)) +
       1(((V(I,J)*A4)+A2)*T(I,J-1))
 21     CONTINUE
C       BOUNDARY CONDITIONS        AT X+=0, X+=L/D
        L1=2
        LM=2
        A1=0.
        AM=0.
C       SOLVE TRI DIAGONAL MATRIX
        CALL GTRI(A,B,C,D,IX,E,F,W,L1,A1,D1,LM,AM,DM)
        DO 22 I=1,IX
        WK(I,J) =W(I)
        WK(I,1) =T(I,1)
        WK(I,IY) =T(I,IY)
 22     CONTINUE
 2      CONTINUE
        IF(Z.EQ.1. ) GO TO 11
        PRINT *,"TEMPERATURE DISTRIBUTION, TIME= ",TIMEA
        DO 23 I=1,IX
        K=IX+1 -I
        WRITE(6,100) (WK(K,J),J=1,IY)
 23     CONTINUE
C
C       NOW SOLVE TEMPERATURE DISTRIBUTION ,TIME=N+1
C
C
 11     TIMEB =TIME +DT
        DO 3   I=2,IXA
        DO 31  J=2,IYA
        A(J) =(A2 -(V(I,J)*A4))
        B(J) =(A5  +(2.*A2))
        C(J) =((V(I,J)*A4)+A2)
        D(J) = ((A6- (U(I,J)*A3)) *WK(I+1,J)) +((A5-(2.*A6))* WK(I,J))
       1 + (((U(I,J)*A3)+A6)* WK(I-1,J))
 31     CONTINUE
C       SOLVE TRI-DIAGONAL MATRIX
        L1= 1
        LM=1
        A1= 0.
        AM=1.0
        CALL GTRI (A,B,C,D,IY,E,F,W,L1,A1,D1,LM,AM,DM)
```

```
      DO 32  J=1,IY
      T(I,J) =U(J)
      T(1,J)=WK(1,J)
      T(IX,J)=WK(IX,J)
 32   CONTINUE
 3    CONTINUE
      IF(Z.EQ.1.) GO TO 12
      PRINT*, "TEMPERATURE DISTRIBUTION ,TIME= ",TIMER
      DO 33 I=1,IX
      K=IX+1 -I
      WRITE(6,900)  (T(K,J),J=1,IY)
 33   CONTINUE
C
C   SOLVE EQN OF MOTION (VORTICITY),VELOCITY AT TIME =N
C
C      TEMPERATURE GRADIENT AT TIME=N+1
C
 12   B1 = DT/(DX**2.)
      B2 =DT/(DY**2.)
      B3 =DT/(2.*DX)
      B4  =DT/(2.*DY)
      B5  =?.
      B6  =(GR*SIN(S)*B4)
      B7 =(GR*COS(S)*B3)
      DO 4    J=2,IYA
      DO 41  I=2,IXA
      A(I)=(B1-(U(I,J)*B3))
      B(I)= (B5+(2.*B1))
      C(I)=((U(I,J)*B3)+B1)
      D(I)=(B6*(T(I,J+1)-T(I,J-1)))-(B7*(T(I+1,J)-T(I-1,J)))+
     1((B2-(V(I,J)*B4))*VOR(I,J+1)) +((B5-(2.*B2))*VOR(I,J))+
     2 ((B2+(V(I,J)*B4))*VOR(I,J-1))
 41   CONTINUE
      L1=1
      LM=1
      A1= VOR(1,J)
      AM= VOR(IX,J)
      CALL STRI(A,B,C,D,IX,E,F,W,L1,A1,B1,LM,AM,BM)
      DO 42 I=1,IX
      WK(I,1)=VOR(I,1)
      WK(I,IY)=VOR(I,IY)
      WK(I,J)= W(I)
 42   CONTINUE
 4    CONTINUE
      IF(Z.EQ.1. ) GO TO 13
      PRINT*, "VORTICITY DISTRIBUTION, TIME = ",TIMER
      DO 43 I=1,IX
      K=IX+1 -I
      WRITE(6,900)  (WK(K,J),J=1,IY)
 43   CONTINUE
C
C   SOLVE VORTICITY AT TIME=N+1
C
C
 13   DO 5   I=2,IXA
      DO 51  J=2,IYA
      A(J)=(B2-(V(I,J)*B4))
```

```
      B(J)= (B3+ (2.* B2))
      C(J)= ((V(I,J) B4)+B2)
      D(J)=(B5 (T(I,J+1)-T(I,J-1)))-(B7*(T(I+1,J)-T(I-1,J))) +
     1 ((B1-(U(I,J)*B3))*WK(I+1,J))+((B5-(2.*F1))* WK(I,J)) +
     2 ((B1+(U(I,J)*B3))*WK(I-1,J))
 51   CONTINUE
      L1=1
      LM=1
      A1=VOR(I,1)
      AM=VOR(I,IY)
      CALL GTRI(A,B,C,D,IY,E,F,W,L1,A1,C1,LM,AM,CM)
      DO 52 J=1,IY
      VOR(I,J)=W(J)
 52   CONTINUE
 5    CONTINUE
      IF(Z.EQ.1.) GO TO 14
      PRINT*, "VORTICITY DISTRIBUTION , TIME = ", TIMER
      DO 53 I=1,IX
      K=IX+1-I
      WRITE(6,*) (VOR(K,J),J=1,IY)
 53   CONTINUE
C
C
C     SOLVE STREAM FUNCTION( METHOD OF SUCCESSIVE OVERRELAXATION)
C     VALUES OF WOPT ARE FROM THE RESULTS OF WILKES SCHUCHTLL
C     ASSUME DX = DY
C
C
 14   IF(IYA - 2 ) 141,142,143
 141  WOPT =1.68
      GO TO 14
 142  WOPT =1.7
      GO TO 14
 143  WOPT = 1.73
 14   DO 5 N=1,2
      DO 51 I=2,IXA
      DO 52 J=2,IYA
      ST(I,J)=ST(I,J)+((WOPT/4.0)*(((DY**2.)*VOR(I,J))+ST(I-1,J)+
     1 ST(I+1,J)+ST(I,J-1)+ST(I,J+1) -4. *ST(I,J)))
 52   CONTINUE
 51   CONTINUE
      IF(N.LT.2 .OR.Z .EQ. 1. ) GO TO
      PRINT*, "STREAM FUNCTION, TIME = ", TIMER
      DO 5 I=1,IX
      K=IX+1-I
      WRITE(6,*) (ST(K,J),J=1,IY)
 55   CONTINUE
 5    CONTINUE
C
C
C     CALCULATE VORTICITY AT THE WALL FROM THE STREAM FUNCTION AT T0
C
C
C
      IF(G.GT.2 ... ) GO TO 145
      DO   I=1,IX
      VOR(I,1)= -((8.*ST(I,2)-ST(I,3))/(2.*(DY**2.)))
      IY3=IY-2
      VOR(I,IY)= -((8. *ST(I,IYA)-ST(I,IY3))/(2.*(DY *2.)))
```

```
 7      CONTINUE
        DO 71 J=1,IY
        VOR(1,J)= -((8.*ST(2,J)-ST(3,J))/(2.*(DX**2.)))
        IXP=IX -2
        VOR(IX,J)= -((8.*ST(IXA,J)-ST(IXP,J))/(2.*(DX**2.)))
 71     CONTINUE
        GO TO 147
C
C
C      CALCULATE WALL VORTICITY FROM STREAM FUNCTION AT TIME=N+1
C       FIRST ORDER MODEL
C
C
 145    DO 75 I=1,IX
        VOR(I,1)=(-2.*ST(I,2))/DY**2.
        VOR(I,IY)=(-2.*ST(I,IYA))/DY**2.
 75     CONTINUE
        DO 76 J=1,IY
        VOR(1,J)=(-2.*ST(2,J))/DX**2.
        VOR(IX,J)=(-2.*ST(IXA,J))/DX**2.
 76     CONTINUE
 147    IF(Z.EQ.1. ) GO TO 150
        PRINT*, "VORTICITY AT TIME= ", TIMEB
        DO 72 I=1,IX
        K=IX+1-I
        WRITE(6,30 ) (VOR(K,J),J=1,IY)
 72     CONTINUE
C
C       CALCULATE VELOCITIES AT TIME =N+1
C
 150    DO 30 I=2,IXA
        DO 31 J=2,IYA
        U(I,J)=(ST(I,J+1)-ST(I,J-1))/(2.*DY)
        V(I,J) =-(ST(I+1,J)-ST(I-1,J))/(2.*DX)
 31     CONTINUE
 30     CONTINUE
        IF(Z.EQ.1. ) GO TO 160
        PRINT*, "VELOCITIES AT TIME= ",TIMEB
        PRINT*, "U-VELOCITY "
        DO 32 I=1,IX
        K=IX+1-I
        WRITE(6, 30)  (U(K,J),J=1,IY)
 32     CONTINUE
        PRINT*,  "V-VELOCITY  "
        DO 33 I=1,IX
        K=IX+1-I
        WRITE(6,300)  (V(K,J),J=1,IY)
 33     CONTINUE
C
C    CALCULATE LOCAL AND AVERAGE NUSSELT NUMBER
C
 160    NUH= .
        NUC =0.0
        DO 9  I=1,IX
        DXI =1.0
        IF (I.EQ.1  .OR. I.EQ. IX) DXI =0.5
        A(I) =(-(T(I,3) -4.*T(I,2) + 3.0*T(I,1)))/(2.0*DY)
```

```
      B(I) =(T(I,IY-2) -4.*T(I,IY-1) + 3.*T(I,IY)) /(2.*DY)
      NUH =NUH +DXI*DX A(I)/AR
      NUC =NUC + DXI DX *B(I)/AR
 90   CONTINUE
      PRINT*, "       "
      PRINT*, "       "
      PRINT*, "TIME =   ",TIMEB
      PRINT*, "       "
      DO 91  I=1,IY
      K=IX+1 -I
      WRITE(6,501) A(K), B(K)
 91   CONTINUE
      PRINT*, "       "
      PRINT*, "  AVERAGE NUSSELT NUMBER ( HOT WALL ) =    ",NUH
      PRINT*, "  AVERAGE NUSSELT NUMBER ( COLD WALL )  = ", NUC
      TIME =TIMEB
      IF(TIME.EQ...150) Z=2.
      IF(TIME  .GT. ..10) GO TO 130
      GO TO 1
 130  STOP
      END
```

```
C
      SUBROUTINE STRI(A,B,C,D,MD,E,F,W,L1,A1,D1,LM,AM,DM)
C     GENERAL TRI DIAGONAL MATRIX SOLVER
      DIMENSION A(31), B(31),C(31),D(31),E(31),F(31),W(31)
      IF(L1.EQ.1) E(1)=0.0
      IF(L1.EQ.1) F(1)= A1
C     ABOVE OVERWRITTEN IF LM.NE.2
      IF(L1.EQ.2) E(1)=1.0
      IF(L1.EQ.2) F(1)=-A1
      IF(L1.EQ.3) E(1)=A1/(A1-1.0)
      IF(L1.EQ.3) F(1)=D1/(1.0-A1)
      MM=MD-1
      DO 1 M=2,MM
      DEN= B(M)-C(M)*E(M-1)
      E(M)= A(M)/DEN
    1 F(M)=(D(M)+C(M)*F(M-1))/DEN
      IF(LM.EQ.1) W(MD)=AM
      IF(LM.EQ.2) W(MD)=(F(MM)+AM)/(1.0-E(MM))
      IF(LM.EQ.3) W(MD)=(F(MM)+DM/AM)/((1.0+AM)/AM-E(MM))
      DO 2 MK=1,MM
      M=MD-MK
    2 W(M)= E(M)*W(M+1)+F(M)
      RETURN
      END
```

```
      [illegible] (INPUT,OUTPUT,TAPE[illegible],TAPE6=OUTPUT)
      [illegible] ([illegible],[illegible]),[illegible]([illegible]),[illegible],J([illegible]),V([illegible]),
     1[illegible]([illegible]),[illegible](31),[illegible](31),[illegible](31),[illegible](31),E(31),F(31),W(31)
      [illegible]
 [illegible]  FORMAT([illegible]X,[illegible](F[illegible]))
 [illegible]1 FORMAT([illegible]X,F11.[illegible], [illegible]X, F11.[illegible])
C
C      INPUT INITIAL,BOUNDARY CONDITIONS
C
 1    [illegible]
      [illegible],[illegible],[illegible],[illegible],DY,[illegible],[illegible],[illegible],7
      IF([illegible]) .[illegible]. [illegible]) GO TO [illegible]
      [illegible]
      [illegible] = [illegible] /[illegible]Y
      [illegible]Y = [illegible] + 1.    [illegible]1
      [illegible] = [illegible]. / Y
      IY = [illegible] + 1.    [illegible]1
C
      [illegible] , "          "
      [illegible] , "          "
      PRINT [illegible],"[illegible] NUMBER = ",[illegible]
      [illegible] , "G[illegible] NUMBER = ",GR
      [illegible] , "[illegible] RATIO = ",[illegible]
      [illegible] , "[illegible] X = ",[illegible]X, "DELTA Y = ",[illegible]Y
      [illegible] , "NUMBER OF INCREMENTS = ",IX
      [illegible] , "Y-[illegible] = ",IY
      [illegible] , "[illegible] = ",DT
      [illegible] , "[illegible] ANGLE = ", S1
      [illegible] , "[illegible] ANGLE = ", [illegible]
      [illegible] , "          "
      [illegible] , "          "
C
      DO [illegible] J=1,[illegible]Y
      [illegible]=[illegible]Y+1-J
      Y[illegible]=[illegible]-1
      Y[illegible]=[illegible]Y-[illegible]
      DO [illegible] I=1,[illegible]
      IF(J.EQ.1) GO TO 1[illegible]
      IF(I.[illegible]EQ.[illegible]) GO TO 17
      [illegible]([illegible],J) = [illegible]
      GO TO [illegible]
 [illegible]  [illegible]([illegible],J) = [illegible].
      GO TO 1[illegible]
 [illegible]  [illegible]([illegible],J)=YJ1 /YJ[illegible]
 [illegible]  [illegible](I,J) = [illegible]
      [illegible](I,J) = [illegible].
      V(I,J)= [illegible].
      [illegible](I,J) = [illegible].
 [illegible]  CONTINUE
 [illegible]  CONTINUE
      IF(I.EQ.1.[illegible]) GO TO 1[illegible]
      PRINT [illegible], "[illegible] = ",TIME
      PRINT [illegible], "[illegible] FUNCTION"
      [illegible] ([illegible],[illegible]) ((S[illegible](I,J),J=1,IY),I=1,IX)
      PRINT [illegible], "V[illegible]CITY  "
      [illegible] ([illegible],[illegible]) ((V[illegible](I,J),J=1,IY),I=1,IX)
      [illegible] , " U-VELOCITY  "
```

```
      WRITE (?,  ) ((U(I,J)),J=1,IY),I=1,IX)
      P   ?   , " V-VELOCITY  "
      WRITE (?,?  ) ((V(I,J)),J=1,IY),I=1,IX)
      PRINT , " TEMPERATURE  "
      WRITE ( ,?  ) ((T(I,J)),J=1,IY),I=1,IX)
C
C     TEMPERATURE SOLUTION IN X_DIRECTION (INTERIOR POINTS)
C     ENERGY EQUATION FOR TIME=N+1/2
C
C
 1    TIME = TIME + . DT
      IX1=IX-1
      IY1= Y-1
      IX2=IX-2
      IY2= Y-2
      ? = DT/ (?  (DY  2.))
      ?  = DT /(2.  DX)
      ?  = DT /(?.  DY)
      ?  = ?.
      ? = DT/(?  (DX  2.))
      DO 2? J=?,IY3
      ?=IY-J
      ? = ?+1
      DO 2? I=?,?
      A(I) =?  -(U(I,J) ?)
      B(I) =?  +(2.  ? )
      C(I) =(U(I,J)  ?) +?
      D( ) =((?2-(V(I,J) ?)) T(?,J+1)) +((? -(2.*A2))*T(I,J)) +
     1(((V(?,J)  )+?2) ?(I,J-1))
 2?   CONTINUE
C     BOUNDARY CONDITIONS        AT X*=?, X*=L/D
      L? =?
      L?=1
      ?  = .
      ?  =?(?,J)
      CALL ?(?,?,C,?,?1,E,F,U,?,?1,D1,?,?M,?)
      DO 22 I=1,?
      WK(I,J) =?(I)
 22   CONTINUE
 2    CONTINUE
      DO 2  I=1,IY
      WK(I,1) = ?(1,1)
 2?   CONTINUE
      WK(1,IY )=?(?,IY )
      WK(?,IY )= (?, Y )
      WK(?,IY) = ?(?,IY)
      IF(?.??.?. ) GO TO 11
      PRINT  ," TEMPERATURE DISTRIBUTION, TIME= ",TIME?
      DO 2  I=1,IX
      K= IY+1 -I
      WRITE(?,  )(WK(K,J),J=1,I)
 2    CONTINUE
C
C     NOW SOLVE TEMPERATURE DISTRIBUTION ,TIME=N+1
C
C
 11   TIME? =TIME +DT
```

```
      DO 3  I=2,IXP
      I = IX-I
      M1 = I+1
      DO 2  J=2,
      A(J)  =(A  -(V(I,J)  A ))
      B(J)  =(   +(2.  B))
      C(J)  =((V(I,J) A  )+ 2)
      D(J)  = ((  -  (U(I,J) A3))  WK(I+1,J))  +((A -(2. AE))  WK(I,J)
     1 + (((U(I,J) A3)+  )  WK(I-1,J))
 21   CONTINUE
C     SOLVE TRI-DIAGONAL MATRIX
      L=1
      L =1
      
      
      CALL  TRI (A,B,C,D, 1,E,F,W,L1, 1,D1,LM,A4,ON)
      DO 22  J=1,M1
      T(I,J) =W(J)
       (I,J)=WK(I,J)
 22   CONTINUE
 3    CONTINUE
      IF(T. T .  )  ) 12
      PRINT  , "TEMPERATURE DISTRIBUTION ,TIME= ",TIME
      DO 77  I=1,IX
      K= IX+1 -I
      PRINT  (  ,   ) (T(K,J), J=1,I)
 77   CONTINUE
C
C   SOLVE FOR  (VORTICITY),VELOCITY AT TIME =N
C       T  GRADIENT AT TIME=N+1
C
C
 12   B1 = DT/(DY  2.)
      B2 =DT/(DY  2.)
      B3 =DT/(2. DY)
         =DT/(2. DY)
            
      B  =(G  ST (S)  )
      B  =(G  COS(S)  B3)
      DO      I=2,IXP
       IX- I
         =  +1
      DO      I=1,
      A( )=(B1-(U( ,J)  B3))
       ( )  (   +(2.   1))
      C( )-((U(I,J)  3)+ 1)
      D( )=(    ( (T,J+1)-T(I,J-1)))-(3  (T( +1,J)-T( -1,J)))+
     1(( 2-(V(I,J)   ))  VOR(I,J+1))  +((B -(2.  ))  VOR(I,J))+
     2  ((3 +(V(I,J)   ))  VOR(I,J-1))
 14   CONTINUE
      L =1
      L =1
         =VO (I,J)
         =V   ( I,J)
      CALL  TRI (A,B,C,D,M1,E,F,W, 1,A1,D1,LM,A4,ON)
      DO  2  I=1,M1
      WK(I,1)  =VOR(I,1)
```

```
      W[illegible](I,J) =W(J)
      [illegible](1,IY) = [illegible] (1,IY)
      WK(1,IY1)=V[illegible] (1,IY1)
      WK(2,IY1)=[illegible](2,IY1)
    [illegible] CONTINUE
      CONTINUE
      IF(T.[illegible].1. ) GO TO 13
      PRINT , "V[illegible]ICITY DISTRIBUTION, TIME = ",TIMEA
      DO [illegible] I=1,IX
      K= IX+1 -I
      WRITE ([illegible], [illegible] )(WK(K,J),J=1,I)
  [illegible] CONTINUE
C
C   COMPUTE VORTICITY AT TIME=N+1
C
C
 1[illegible] DO [illegible] I=2,IX[illegible]
      N= IY-[illegible]
      [illegible] = [illegible]+1
      DO [illegible] J=[illegible],[illegible]
      A(J)=(B[illegible]-(V(I,J)[illegible]B[illegible]))
      B(J)=(B[illegible]+ (2. [illegible]B[illegible]))
      C(J)= ((V([illegible],J) B[illegible] )+[illegible])
      D(J)=(B[illegible] ([illegible](I,J+1)-[illegible](I,J-1)))-(B[illegible](T(I+1,J)-T(I-1,J))) +
     1 ((B[illegible]-(U(I,J) [illegible]))[illegible]WK(I+1,J))+((B[illegible]-(2. B1))[illegible] WK(I,J)) +
     2 [illegible](B[illegible]+(U(I,J) [illegible]))[illegible]WK(I-1,J))
  [illegible] CONTINUE
      L[illegible]=1
      L[illegible]=[illegible]
      [illegible] =V[illegible](I,1)
      [illegible]=V[illegible](I,[illegible])
      CALL [illegible](L,A,B,C,D,[illegible],E,F,N,[illegible],A1,B1,[illegible],AM,BM)
      DO [illegible] J=1,[illegible]Y
      V[illegible](I,J) = W(J)
  [illegible] CONTINUE
  [illegible] CONTINUE
      IF(T.[illegible].1. ) GO TO [illegible]
      PRINT , "VORTICITY DISTRIBUTION, TIME = ", TIMEB
      DO [illegible] I=1,IX
      K= IX+1 -I
      WRITE ( ,[illegible] )(V[illegible](K,J)),J=1,I)
  [illegible] CONTINUE
C
C     SOLVE STREAM FUNCTION( METHOD OF SUCCESSIVE OVER-RELAXATION)
C     VALUES OF WOPT ARE BASED ON THE RESULTS OF WILKES FOR HUGHILL
C     [illegible] DX = DY
C
C
C
 1[illegible] IF(IYA - 2 ) 1[illegible]1,1[illegible]2,1[illegible]3
 1[illegible] WOPT =1.[illegible]
      GO TO [illegible]
 1[illegible] WOPT =1.
      GO TO [illegible]
 1[illegible] WOPT = 1.[illegible]
 1[illegible] DO [illegible] I=1,[illegible]
      DO [illegible] I=2,IX[illegible]
      N= IY-[illegible]
```

```
      DO [illegible] J=2,[illegible]
      [illegible](I,J)=[illegible](I,J)+((NSRT/[illegible])*(((DX* 2.) VOR(I,J))+ST(I-1,J)+
     1 [illegible](I+1,J)+ST(I,J-1)+ST(I,J+1) -4. *ST(I,J))))
[illegible] DONE [illegible]
[illegible] CONTINUE
      IF([illegible].LT.[illegible]) GO TO [illegible]
      WRITE[illegible], "STREAM FUNCTION, TIME = ", TIMER
      DO [illegible] J=1,IX
      K= IY+1 -J
      WRITE([illegible]) (ST(K,J),J=1,I)
[illegible] CONTINUE
      CONTINUE
C
C     CALCULATE VORTICITY AT THE WALL FROM THE STREAM FUNCTION AT [illegible]
C
C
      DO [illegible] I=2,IXA
      M=IY+1-I
      VOR(I,1) = (-2.*ST(I,2))/(DY**2.)
      VOR(1,I) = (-2.*ST(2,I))/(DX**2.)
      VOR(I,M)=-2.*(ST(I,M-1)+ST(I-1,M))/DX** 2.
[illegible] CONTINUE
      IF([illegible].EQ.1.) GO TO [illegible]
      PRINT[illegible], "VORTICITY AT TIME= ", TIMER
      DO [illegible] J=1,IX
      K= IY+1 -J
      WRITE([illegible]) (VOR(K,J),J=1,I)
[illegible] CONTINUE
C
C     CALCULATE VELOCITIES AT TIME =N+1
C
[illegible] DO [illegible] J=2,IY2
      M= IY-J
      DO [illegible] I=2,[illegible]
      U(I,J) =(ST(I,J+1) - ST(I,J-1))/(2.*DY)
      V(I,J) =-(ST(I+1,J)-ST(I-1,J))/(2.*DX)
[illegible] CONTINUE
[illegible] CONTINUE
      IF([illegible].EQ.[illegible]) GO TO [illegible]
      PRINT[illegible], "VELOCITIES AT TIME= ",TIMER
      PRINT[illegible], "U-VELOCITY "
      DO [illegible] J=1,IX
      K= IY+1 -J
      WRITE([illegible]) (U(K,J),J=1,I)
[illegible] CONTINUE
      PRINT[illegible], "V-VELOCITY "
      DO [illegible] I=1,IX
      K =IY+1-I
      WRITE([illegible]) (V(K,J),J=1,I)
[illegible] CONTINUE
C
C   CALCULATE LOCAL AND AVERAGE NUSSELT NUMBER
C
[illegible] [illegible]= .
      [illegible] = .
      DO [illegible] I=1,IX
      M=IY+1-I
```

```
      DXY=SQRT (DX**2.+DY**2.)
      DXI =1.
      IF(I.EQ.1 .OR. I.EQ.IX)DXI= .5
      IF(I.EQ.IX )GO TO 1
      IF(I.EQ.IX) GO TO 1
      Q(I) =(-(T(I,3) -4.*T(I,2) + 3.*T(I,1)))/(2.*DY)
      GO TO 1
1     Q(I)=-(T(I,1)-T(I,2))/DY
      GO TO 1
1     Q(I)=-(T(I,1)-T(IX4,2))/DXY/ .
      Q(T)=Q(I)
1     NUH =NUH+DXI DY Q(I)
      IF(I.EQ.1) GO TO 1
      IF(I.EQ.I. . . .EQ.IXA) GO TO 103
      IF(I.EQ.IX)GO TO 1
      Q(I) =(T(I-2,M-2)-4.*T(I-1,M-1)+3.*T(1,M))/(2. SQRT(DX**2.+DY**
     1))
      GO TO
4     Q(I)=(T(I,IY)-T(I+1,IY))/DXY
      GO TO
2     Q(I) =(T(I,M)-(T-1,M-1))/(SQRT(DX**2.+DY**2.))
4     NUC= NUC+ XI  XY Q(I)/1.-1/A
      CONTINUE
      PRINT , "       "
      PRINT , "       "
      PRINT , "TIME = ",TIME
      DO   =1,IX
      K=IX+1 -I
      PRINT ( ,   ) Q(K), R(K)
      CONTINUE
      PRINT , "  AVERAGE NUSSELT NUMBER ( HOT WALL ) =    ", NUH
      PRINT , "  AVERAGE NUSSELT NUMBER ( COLD WALL )  = ", NUC
      =TIMER
      IF(I. NE.EQ. .0 ) GO TO 1
      GO TO 1
      STOP
      END
```

```
      PROGRAM DTG2(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DIMENSION ST(31,31),VOR(31,31),T(31,31),U(31,31),V(31,31),
     1WK(31,31),A(31),B(31),C(31),D(31),E(31),F(31),W(31)
      REAL NUC,NUP,L
    3 FORMAT(1X,11(F11.5))
  300 FORMAT(2X,F11.5, 3X, F11.5)
C
C     INPUT INITIAL,BOUNDARY CONDITIONS
C
    1 TIME=0.
      READ *,PR,GR,DX,DY,DT,AR,S1,Z
      IF(EOF(5LINPUT) .NE. 0) GO TO 999
      S=S1/57.2957795
      DXY=SQRT (DX**2.+DY**2.)
      XX1 =AR/DX
      IX = XX1 + 1. + 0.1
      XX2 = 1. /DY
      IY = XX2 + 1. + 0.1
C
      PRINT*, "      "
      PRINT*, "      "
      PRINT*,"PRANDTL NUMBER = ",PR
      PRINT*, "GRASHOFF NUMBER = ",GR
      PRINT*, "ASPECT RATIO =  ",AR
      PRINT*, "DELTA X = ",DX, "DELTA Y = ",DY
      PRINT*,  "NUMBER OF INCREMENTS  = ",IX
      PRINT*, "Y-INCREMENTS =  ",IY
      PRINT*, "TIME INCREMENT = ",DT
      PRINT*, "INCLINATION ANGLE = ", S1
      PRINT*, "INCLINATION ANGLE = ", S
      PRINT*, "      "
      PRINT*, "      "
C
      DO 10 I=1,IX
      DO 14 J=1,IY
      T(I,J)= 0.
      E(I,J)=0.
      XJ1 = J-1
      XJ2 =IY-1
      T(I,J)=XJ1/XJ2
   15 ST(I,J) = 0.
      VOR(I,J) = 0.
      U(I,J) = 0.
      V(I,J)= 0.
   14 CONTINUE
   10 CONTINUE
      IF(Z.EQ.1. ) GO TO 105
      PRINT*, "TIME = ",TIME
      PRINT*, "STREAM FUNCTION"
      WRITE (6,3  ) ((ST(I,J),J=1,IY),I=1,IX)
      PRINT*, "VORTICITY  "
      WRITE (6,3  ) ((VOR(I,J),J=1,IY),I=1,IX)
      PRINT*, " U-VELOCITY  "
      WRITE (6,3  ) ((U(I,J),J=1,IY),I=1,IX)
      PRINT*, " V-VELOCITY  "
      WRITE(6,3  ) ((V(I,J),J=1,IY),I=1,IX)
      PRINT*, " TEMPERATURE  "
```

```
      WRITE (6,[illegible]) ((T(I,J),J=1,IY),I=1,IX)
C
C     TEMPERATURE SOLUTION IN X_DIRECTION (INTERIOR POINTS)
C     SOLVE ENERGY EQUATION FOR TIME=N+1/2
C
C
C
 1    TIMEH= TIME + [illegible]*DT
      IX1=IX-1
      IY1=IY-1
      IX3=IY-2
      IY3=IY-2
      A2 = DT/ (PE*(DY**2.))
      A3  = DT /(2.*DX)
      A4  =DT /(2.*DY)
      A5 = 2.
      A6 = DT/(PE*(DX**2.))
      DO 2  J=2,IY3
      N1=IY+1-J
      J1= J+1
      DO 21 I=J1,IY4
      K=I+1-J
      A(K) =A[illegible] -(U(I,J)*A3)
      B(K) =A[illegible] +(2.*A6)
      C(K) =(U(I,J)*A3) +A6
      D(K) =(((A2-(V(I,J)*A4))*T(I,J+1))+((A[illegible]-(2.*A2))*T(I,J)) +
     1(((V(I,J)*A4)+A2) T(I,J-1))
 21   CONTINUE
C     BOUNDARY CONDITIONS        AT X+=0, X+=L/D
      L1=1
      LM=2
      A1=T(J,J)
      B1= [illegible]
      CALL GTRI(A,B,C,D,B1,E,F,W,L1,A1,D1,LM,AM,DM)
      DO 22 I=J,IX
      K=I+1-J
      WK(I,1)=T(I,1)
      WK(1,I)=T(1,I)
      WK(IX,IY1)=T(IX1,IY1)
      WK(IX1,IY1)=T(IX1,IY1)
      WK(IX,IY)=T(IX,IY)
      WK(I,J)=W(K)
 22   CONTINUE
 2    CONTINUE
      IF(Z.EQ.1.[illegible]) GO TO 11
      PRINT[illegible],"TEMPERATURE DISTRIBUTION, TIME= ",TIME[illegible]
      DO 23 I=1,IX
      K=IX+1-I
      WRITE (6,[illegible])(WK(K,J),J=1,K)
 23   CONTINUE
C
C     NOW SOLVE TEMPERATURE DISTRIBUTION ,TIME=N+1
C
C
C
 11   TIMEF =TIME +DT
      DO 3  I=3,IX4
      I1=I-1
      DO 31 J=2,I1
```

```
      .(J)  =(A2  -(V(I,J)*AW))
      .(J)  =(/   +(2.  A2))
      .(J)  =((V(I,J) A  )+A2)
      .(J)  = ((A.- (U(I,J)*A7)) *WK(I+1,J))  +((A.-(2.*A5))* WK(I,J))
     1 + (((U(I,J) A3)+A.)* WK(I-1,J))
 .1   CONTINUE
C     SOLV  TRI-DIAGONAL MATRIX
      L1=1
      L.=1
      M1=.
      A. =. (1,1)
      CALL GTRI(A,.,C,D,I,E,F,W,L1,A1,D1,LM,AM,DM)
      DO 32  J=1,.
      T(I,J)=W(J)
      .(IX,J)=WK(.X,J)
 32   CONTINUE
 3    CONTINUE
      IF(7.EQ.1. ) GO TO 12
      PRIN .,  "T..PERATUR. DISTRIBUTION ,TIME= ",TIMER
      DO .. .=1,.X
      K=IX+1-1
      WRITE(.,.  )(T(K,J),J=1,K)
 3.   CONTINUE
C
C   SOLVE EQN OF MOTION (VORTICITY),VELOCITY AT TIME =N
C      TEMPERATURE GRADIENT AT TIME=N+1
C
C
 12   B1  = DT/(DX* 2.)
      B2 =DT/(DY *2.)
      B3 =DT/(2. DX)
      B.   =DT/(2.*DY)
      B.   =2.
      B.   =(G. S.N(S)*B.)
      B. =(G.*COS(S) B3)
      DO .  J=2,IYB
      M.=IX+1-J
      J1=J+1
      DO .1 I=J1,IX.
      K=I+1-J
      A(K)=(B1-(U(I,J) B3))
      B(K)= (B. +(2.  B1))
      C(K)=((U(I,J) B3)+B1)
      D(K)=(B.  (T(I,J+1)-T(I,J-1)))-(B.*(T(I+1,J)-T(I-1,J)))+
     1((B2-(V(I,J)*B.))*VOR(I,J+1)) +((B.-(2.*B2))*VOR(I,J))+
     2 ((B2+(V(I,J) B4)) VOR(I,J-1))
 .1   CONTINUE
      L1=1
      LM=1
      A1=VOR(J,J)
      .M=VOR(IX,J)
      CALL GTRI(A,B,C,D,M1,E,F,W,L1,A1,D1,LM,AM,DM)
      DO .. I=J,IX
      K=I+1-J
      WK(1,1)=VOR(1,1)
      WK(I,1)=VO. (I,1)
      WK(IXA,IYA)=VO.(IXA,IYA)
```

```
      WK(I,J)=W(K)
      WK(IX,IY)=VOR(IX,IY)
      WK(IX,IY1)=VOR(IX,IY4)
 42   CONTINUE
  4   CONTINUE
      IF(Z.EQ.1.) GO TO 15
      PRINT*, "VORTICITY DISTRIBUTION, TIME = ",TIMEA
      DO 43 I=1,IX
      K= IY+1 -I
      WRITE(6,**)(WK(K,J),J=1,K)
 43   CONTINUE
C
C     SOLVE VORTICITY AT TIME=N+1
C
C
C
 15   DO 3  I=3,IXA
      I1=I-1
      DO 31 J=2,I1
      (J)=(B2-(V(I,J)*B3))
      C(J)=(B + (2.*B2))
      D(J)= ((V(I,J)*B3)+B2)
      E(J)=(B1*(T(I,J+1)-T(I,J-1)))-(B4*(T(I+1,J)-T(I-1,J))) +
     1 ((B1-(U(I,J)*B3))*WK(I+1,J))+((B3-(2.*B1))*WK(I,J)) +
     2 ((B1+(U(I,J)*B3))*WK(I-1,J))
 31   CONTINUE
      L1=1
      LM=1
      A1 =VOR(I,1)
      AM =VOR(I,I)
      CALL STR (A,B,C,D,1,E,F,W,L1,A1,B1,LM,AM,BM)
      DO 32 J=1,I
      VOR(I,J) = W(J)
 32   CONTINUE
  3   CONTINUE
      IF(Z.EQ.1.) GO TO 1..
      PRINT*, "VORTICITY DISTRIBUTION , TIME = ", TIMEB
      DO 33 I=1,IX
      K= IY+1 -I
      WRITE(6,**)(VOR(K,J),J=1,K)
 33   CONTINUE
C
C     SOLVE STREAM FUNCTION( METHOD OF SUCCESSIVE OVERRELAXATION)
C     VALUES OF WOPT ARE FROM THE RESULTS OF WILKES SCHUCHILL
C     ASSUME DX = DY
C
C
 1.   IF(IXA - 2 ) 101,102,103
 101  WOPT =1.18
      GO TO 1..
 102  WOPT =1.
      GO TO 1..
 103  WOPT = 1.2.
 1.   DO 5   I=1,2.
      DO 51 I=3,IXA
      I1=I-1
      DO 52 J=2,I1
      ST(I,J)=ST(I,J)+((WOPT/4.)*(((DX**2.)*VOR(I,J))+ST(I-1,J)+
```

```
      1 ST( +1,J)+ST(I,J-1)+ST(I,J+1) -4. *ST(I,J)))
  32  CONTINUE
  34  CONTINUE
      IF(N.LT.2  .OR.Z .EQ. 1.0) GO TO 37
      PRINT*, "STREAM FUNCTION, TIME = ", TIMER
      DO 3  I=1,IX
      K= IX+1 -I
      WRITE(6,3  )(ST(K,J),J=1,K)
  33  CONTINUE
  37  CONTINUE
C
C     CALCULATE VORTICITY AT THE WALL FROM THE STREAM FUNCTION AT T
C
C
      DO 7  I=2,IXA
      VOR(I,1)=-(2.*ST(I,2))/DY**2.
      VOR(IX,I)=-(2.*ST(IXA,I))/DX**2.
      VOR(I,I)=-2.*(ST(I+1,I)+ST(I,I-1))/DX**2.
   7  CONTINUE
      IF(Z.EQ.1. ) GO TO 10
      PRINT*, "VORTICITY AT TIME= ", TIMER
      DO 12 I=1,IX
      K= IX+1 -I
      WRITE(6,  )(VOR(K,J),J=1,K)
  12  CONTINUE
C
C     CALCULATE VELOCITIES AT TIME =N+1
C
  10  DO 8  I=3,IXA
      I1=I-1
      DO 31 J=2,I1
      U(I,J) =(ST(I,J+1) - ST(I,J-1))/(2.*DY)
      V(I,J) =-(ST(I+1,J)-ST(I-1,J))/(2.*DX)
  31  CONTINUE
   8  CONTINUE
      IF(Z.EQ.1. ) GO TO 14
      PRINT*, "VELOCITIES AT TIME= ",TIMER
      PRINT*, "U-VELOCITY "
      DO 32 I=1,IX
      K= IX+1 -I
      WRITE(6,  )(U(K,J),J=1,K)
  32  CONTINUE
      PRINT*,  "V-VELOCITY  "
      DO 3  I=1,IX
      K =IX+1-I
      WRITE(6,  )(V(K,J),J=1,K)
  3   CONTINUE
C
C   CALCULATE LOCAL AND AVERAGE NUSSELT NUMBER
C
  13  NUH= .
      NUC = .
      DO 1  I=1,IX
      DXI =1.
      IF (I.EQ.1  .OR. I.EQ. IX) DXI =.5
      IF(I.EQ.1) GO TO 1
      IF(I.EQ.2) GO TO 15
```

```
      (I) =(-(T(I,3) -4. *T(I,2) + 3.*T(I,1)))/(2.*DY)
      GO TO 1
 1    A(I)= -(T(1,I)-T(I+1,I+1))/DXY/.7 7
      B(1)=A(1)
      GO TO 1
 1    A(I)=-(T(I,1)-T(I,2))/DY
 1    NUH=NUH+DXY*DY A(I)/A
      IF(I.EQ.IX) GO TO 1 7
      IF(I.EQ.IX . .OR. I.EQ.2) GO TO 153
      IF(I.EQ.1) GO TO 1
      B(I)=(T(I+2,I-2)- .*T(I+1,I-1)+3.*T(I,1))/2./DXY
      GO TO 1
 1    B(I)=(T(I,1)-T(I-1,I-1))/DXY
      GO TO 1
 1    B(T)= -(T(I+1,I-1)-T(I,I))/DXY
 1    NUC= NUC+DXY*DXY*B(I)/1. 14/
 3    CONTINUE
      PRINT*, "        "
      PRINT*, "        "
      PRINT*, "TIME =  ",TIMEB
      PRINT*, "        "
      DO 21 I=1,IY
      K=IX+1 -I
      WRITE(6, 1) A(K), B(K)
 21   CONTINUE
      PRINT*, "        "
      PRINT*, "  AVERAGE NUSSELT NUMBER ( HOT WALL ) =     ", NUH
      PRINT*, "  AVERAGE NUSSELT NUMBER ( COLD WALL )  = ", NUC
      TIME =TIME
      IF(TIME.GT. .1 ) GO TO 1
      GO TO 1
 9    STOP
      END
```

```
      PROGRAM PARAL(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DIMENSION ST(31,31),VO (31,31),T(31,31),U(31,31),V(31,31),
     1WK(31,31),A(31),B(31),C(31),D(31),E(31),F(31),W(31)
      REAL NUC,NUH,NUHT,NUCA
 30   FORMAT(1X,11(F11.5))
 31   FORMAT(2X,F11.5,5X, F11.5)
 32   FORMAT(1X,11(F11.5))
C
C     INPUT INITIAL,BOUNDARY CONDITIONS
C
 1    TIME= .0
      READ *,PR,GR,DX,DY,DT,AR,S1,Z
      IF(EOF(5LINPUT) .NE. 0) GO TO 333
      S=S1/57.29578
      AR1 =AR/DX
      IX = AR1 + [illegible]1
      AR2 = 1. /DY
      IY = AR2 + 1.[illegible]1
      IYY=IY*2.-1
C
      PRINT *, "       "
      PRINT *, "       "
      PRINT *,"PRANTL NUMBER = ",PR
      PRINT *, "GRASHOFF NUMBER = ",GR
      PRINT *, "ASPECT RATIO = ",AR
      PRINT *, "DELTA X = ",DX, "DELTA Y = ",DY
      PRINT *, "NUMBER OF INCREMENTS = ",IX
      PRINT *, "Y-INCREMENTS = ",IY
      PRINT *, "TIME INCREMENT = ",DT
      PRINT *, "INCLINATION ANGLE = ", S1
      PRINT *, "INCLINATION ANGLE = ", S
      PRINT *, "       "
      PRINT *, "       "
C
      DO10 J=1,IY
      M=IX+1-J
      M2=IYY+1-J
      XJ1=J-1
      XJ2=IY-1
      DO11 I=M,M2
      IF(J.EQ.IY) GO TO 15
      IF(J.EQ.1) GO TO 16
      IF(I.EQ.M.OR.I.EQ.M2) GO TO 17
      T(I,J) = .0
      GO TO 18
 15   T(I,IY)=XJ1/XJ2
      GO TO 18
 16   T(I,J)= 0
      GO TO 18
 17   T(I,J)=XJ1/XJ2
 18   ST(I,J)= .0
      VO (I,J)= .
      U(I,J) = .
      V(I,J) = .
 11   CONTINUE
 10   CONTINUE
      IF(Z.EQ.1. ) GO TO 100
```

```
      PRINT*, "TIME = ",TIME
      PRINT*, "STREAM FUNCTION"
      WRITE(6,   ) ((SI(I,J),J=1,IY),I=1,IXX)
      PRINT*, "VORTICITY  "
      WRITE(6,   ) ((VOR(I,J),J=1,IY),I=1,IXX)
      PRINT*, " U-VELOCITY  "
      WRITE(6,   ) ((U(I,J),J=1,IY),I=1,IXX)
      PRINT*, " V-VELOCITY  "
      WRITE(6,   ) ((V(I,J),J=1,IY),I=1,IXX)
      PRINT*, " TEMPERATURE  "
      WRITE(6,   ) ((T(I,J),J=1,IY),I=1,IXX)
C
C     SOLVE ENERGY EQUATION FOR TIME=N+1/2
C     TEMPERATURE SOLUTION IN X_DIRECTION (INTERIOR POINTS)
C
C
    1 TIMEN= TIME + .  DT
      IXA=IX-1
      IYA=IY-1
      IXB=IX-2
      IYB=IY-2
      IXXA=IXX-1
      IXXB=IXX-2
      DXY=SQRT(DX  2.+DY  2.)
      A2 = DT/ (PR (DY  2.))
      A3  = DT /(2.  DX)
      A   =DT /(2.  DY)
      A  = 2.
      A  = DT/(P  (DX   2.))
      DO 2   J=2,IYA
      M=IX+2-J
      M1=IXX-J
      M2= IX+1-J
      M3=IXX+1-J
      DO 21 I= ,M1
      K=I-IX+J
      A(K)  =A  -(U(I,J)  A3)
      B(K)  =A  +(2.  A )
      C(K)  =(U(I,J) A3)  +A
      D(K)  =((A2-(V(I,J) A  )) T(I,J+1)) +((A -(2.  A2))*T(I,J)) +
     1(((V(I,J) A )+A2) T(I,J-1))
   21 CONTINUE
C     BOUNDARY CONDITIONS         AT  X+=  , X+=L/D
      L1=1
      L =1
      A1=T(M2,J)
      A4=T(M3,J)
C     SOLVE TRI DIAGONAL MATRIX
      CALL  GTRI(A,B,C,D,IX,E,F,W,L1,A1,D1,LM,AM,CM)
      DO 22  I=M2,M3
      K=I-IX+J
      WK(I, J)=W(K)
      IF(J.EQ.2)  WK(I+1,1)=T(I+1,1)
      IF(J.EQ.IYA)  WK(I-1,IY)=T(I-1,IY)
   22 CONTINUE
    2 CONTINUE
      IF(Z.EQ.1. ) GO TO 11
```

```
      PRINT*,"TE PERATURE DISTRIBUTION, TIME= ",TI E
      DO 2  I=1,IXY
      K=IXY+1-I
      IF(I.GT.IX) GO TO 1 0
      WRITE(6,F  ) (WK(K,J),J=1,I )
      GO TO 2
 1    M =1
      WRITE(6,F 2)(WK(K,J),J=M ,IY)
 2    CONTINUE
C
C     NOW SOLVE TEMPERATURE DISTRIBUTION ,TIME=N+1
C
C
 11   TIMEB =TIME +DT
      DO 3  I=3,IXYT
      IF(I.GT.IX) GO TO 111
      M=I
      M1=IY+2-I
      M2=IYA
      M3=IY
      M =IY+1-I
      M =IY-I
      GO TO 112
 111  M=IXY+1-I
      M1=2
      M2=M-1
      M3=M
      M =1
      M =
 112  DO 31 J=M1,M2
      K=J-M
      A(K) =(A2 -(V(I,J)*A ))
      B(K) =(A  +(2.*A2))
      C(K) =((V(I,J)*A )+A2)
      D(K) = ((A - (U(I,J)*A3)) *WK(I+1,J)) +((A -(2.*A ))*WK(I,J))
     1 + (((U(I,J)*A3)+A )*WK(I-1,J))
 31   CONTINUE
C     SOLVE TRI-DIAGONAL MATRIX
      L1= 1
      LM=1
      A1=T(I,M )
      AM=T(I,M3)
      CALL GTRI (A,B,C,D, M,E,F,W,L1,A1,B1,LM,AM,BM)
      DO 32 J=M ,M3
      K=J-M
      T(I,J)=W(K)
      T(2,IYA)=WK(2,IYA)
 32   CONTINUE
 3    CONTINUE
      IF(Z.EQ.1. ) GO TO 12
      PRINT*, "TEMPERATURE DISTRIBUTION ,TIME= ",TIMEB
      DO 33 I=1,IXY
      K=IXY+1-I
      IF(I.GT.IX) GO TO 11
      WRITE(6,F  )  (T(K,J),J=1,I )
      GO TO 33
 11   M =1
```

```
      WRITE(6,8 2)(T(K,J),J=M ,IY)
 37   CONTINUE
C
C  SOLVE EQN OF MOTION (VORTICITY),VELOCITY AT TIME =N
C     TEMPERATURE GRADIENT AT TIME=N+1
C
C
C
 12   B1 = DT/(DX**2.)
      B2 =DT/(DY**2.)
      B3 =DT/(2.*DX)
      B4  =DT/(2.*DY)
      PR  =2.
      B6  =(GR*SIN(S)*B4)
      B7 =(GR*COS(S)*B3)
      DO 43  J=2,IYA
      M=IX+2-J
      M1=IXX-J
      M2=IX+1-J
      M3=IXX+1-J
      DO 41 I=M,M1
      K=I-IX+J
      A(K)=(B1-(U(I,J)*B3))
      B(K)= (B +(2.*B1))
      C(K)=((U(I,J)*B3)+B1)
      D(K)=(B6*(T(I,J+1)-T(I,J-1)))-(B7*(T(I+1,J)-T(I-1,J)))+
     1((B2-(V(I,J)*B4))*VOR(I,J+1)) +((PR-(2.*B2))*VOR(I,J))+
     2 ((B2+(V(I,J)*B4))*VOR(I,J-1))
 41   CONTINUE
      L1=1
      LM=1
      A1=VOR(M2,J)
      AM=VOR(M3,J)
      CALL GTRI(A,B,C,D,IX,E,F,W,L1,A1,B1,LM,AM,CM)
      DO 42 I=M2,M3
      K=I-IX+J
      IF(J.EQ.2) WK(I+1,1)=VOR(I+1,1)
      WK(I,J)=W(K)
      IF(J.EQ.IYA) WK(I-1,IY)=VOR(I-1,IY)
 42   CONTINUE
 43   CONTINUE
      IF(Z.EQ.1. ) GO TO 13
      PRINT*, "VORTICITY DISTRIBUTION, TIME = ",TIME
      DO 47 I=1,IXX
      K=IXX+1-I
      IF(I.GT.IX) GO TO 121
      WRITE(6,8  )(WK(K,J),J=1,I)
      GO TO 47
 121  MM=1
      WRITE(6,8 2)(WK(K,J),J=MM,IY)
 47   CONTINUE
C
C   SOLVE VORTICITY AT TIME=N+1
C
C
C
 13   DO 5  I=2,IXXP
      IF(I.GT.IX) GO TO 131
      M=I
```

```
      M1=IY+2-I
      M2=IY
      M3=IY
      M4=IY+1-I
      M5=IY-I
      GO TO 132
131   M=IXX+1-I
      M1=2
      M2=M-1
      M3=M
      M4=1
      M5=
132   DO 51  J=M1,M2
      K=J-M
      A(K)=(P2-(V(I,J)*P3))
      B(K)=(P1+ (2.  B3))
      C(K)= ((V(I,J) B3 )+B2)
      D(K)=(P (U(I,J+1)-U(I,J-1)))-(P (U(I+1,J)-U(I-1,J))) +
     1 ((P -(U(I,J) B3)) VK(I+1,J))+((  -(2.  P1))  VK(I,J)) +
     2 ((P +(U(I,J) P3)) VK(I-1,J))
51    CONTINUE
      L1=1
      LM=1
      A1=VOR(I,M )
      AM=VOR(I,M )
      CALL GT  (A,B,C,D, M,E,F,W,L1,A1,B1,LM,AM,BM)
      DO 52  J=M ,M3
      K=J-M
      VOR( ,J)=W(K)
52    CONTINUE
5     CONTINUE
      IF(Z.EQ.1. ) GO TO 1
      PRINT  , "VORTICITY DISTRIBUTION , TIME = ", TI
      DO 53  I=1,IXX
      K=IXX+1-I
      IF(I.GT.IX) GO TO 1
      WRITE( ,   )(VOR(K,J),J=1,I)
      GO TO  3
1     M =1
      WRITE( ,  2)(VOR(K,J),J=M ,IY)
53    CONTINUE
C
C     SOLVE STREAM FUNCTION( METHOD OF SUCCESSIVE OVERRELAXATION)
C     VALUES OF WOPT A  FROM THE RESULTS OF WILKES FORMULA
C     ASSUME DX = DY
C
C
1     IF(IXA - 2 ) 1 1,1 2,1 3
1     WOPT =1.
      GO TO 1
1     WOPT =1.
      GO TO 1
1     WOPT = 1.
1     DO    J=1,2
      DO 61  I=I,IXX
      IF(I.GT.IX) GO TO 1
      M1=IY+2-I
```

```
      M2=IYA
      GO TO 1
1     M1=2
      M2=IXX-I
1     DO 52 J=M1,M2
      ST(I,J)=ST(I,J)+((WOPT/4. )*(((DX* 2.) VOR(I,J))+ST(I-1,J)+
     1 ST(I+1,J)+ST(I,J-1)+ST(I,J+1) -4. *ST(I,J)))
52    CONTINUE
51    CONTINUE
      IF(N.LT.2 .OR.T .EQ. 1.0) GO TO 5
      PRINT*, "STREAM FUNCTION, TIME = ", TIMER
      DO 5  I=1,IXX
      K=IXX+1-I
      IF(I.GT.IX) GO TO 1
      WRITE(6,  )(ST(K,J),J=1,I)
      GO TO 5
1     M4=1
      WRITE(6, 2)(ST(K,J),J=M ,IY)
5     CONTINUE
5     CONTINUE
C
C     CALCULATE VORTICITY AT THE WALL FROM THE STREAM FUNCTION AT T
C
C
      DO 7  I=2,IX
      K=IY+1-I
      IF(I.EQ.2) GO TO 1031
      VOR(I,IY)= -(8. *ST(I,IYA)-ST(I,IYB))/(2.*(DY* 2.))
1031  IF(I.EQ.IX) GO TO 7
      VOR( ,K)=-2. (ST(I,K+1)+ST( +1,K))/DX  2.
7     CONTINUE
      DO 71 I=IX,IXXA
      K=IXX+1-I
      IF(I.EQ.IXXA) GO TO 1032
      VOR(I,1)= -(8. *ST (I,2)-ST(I,3))/(2.*(DY* 2.))
1032  IF(I.EQ.IX)GO TO 71
      VOR(I,K)= -2.*(ST(I,K-1)+ST(I-1,K))/DX 2.
71    CONTINUE
      IF(T.EQ.1. ) GO TO 1
      PRINT*, "VORTICITY AT TIME= ", TIMER
      DO 72 I=1,IXX
      K=IXX+1-I
      IF(I.GT.IX) GO TO 109
      WRITE(6,  )(VOR(K,J),J=1,I)
      GO TO 72
109   M4=1
      WRITE(6, .2)(VOR(K,J),J=M4,IY)
72    CONTINUE
C
C      CALCULATE VELOCITIES AT TIME =N+1
C
1     DO   I=3,IXXB
      IF(I.GT.IX) GO TO1 1
      M=IY+2-I
      M1= IYA
      GO TO 1
1     M=2
```

```
      M1=IYY-I
 1    DO 31 J=M,M1
      U(I,J)=(ST(I,J+1)-ST(I,J-1))/(2.*DY)
      V(I,J) =-(ST(I+1,J)-ST(I-1,J))/(2.*DX)
 31   CONTINUE
 30   CONTINUE
      IF(Z.EQ.1. ) GO TO 10
      PRINT *, "VELOCITIES AT TIME= ",TIMER
      PRINT *, "U-VELOCITY "
      DO 32 I=1,IXX
      K=IXX+1-I
      IF(I.GT.IX) GO TO 1 3
      WRITE(6,9 )(U(K,J),J=1,I)
      GO TO 32
 1 3  M =1
      WRITE(6,9 2)(U(K,J),J=M ,IY)
 32   CONTINUE
      PRINT *,  "V-VELOCITY  "
      DO 3  I=1,IXX
      K=IXX+1-I
      IF(I.GT.IX) GO TO 134
      WRITE(6,9  )(V(K,J),J=1,I)
      GO TO 3
 1    M4=1
      WRITE(6,9.2)(V(K,J),J=M4,IY)
 3    CONTINUE
C
C
C
C    CALCULATE LOCAL AND AVERAGE NUSSELT NUMBER
C
C
 1    NUH= .
      NUH1= .
      NUC= .
      NUC1= .
      DO   I=1,IXX
      K=IY+1-I
      DXI=1.
      IF(I.EQ.1.  .OR.I.EQ.IX .OR.I.EQ.IXX) DXI= .
      A(I)= .
      B(I)= .
      C(I)= .
      D(I)= .
      IF(I.EQ.1. ) GO TO 161
      IF(I.EQ.2) GO TO 162
      IF(I.EQ.IXX ) GO TO 163
      IF(I.EQ.IXY) GO TO 164
      IF(I.EQ.IX)GO TO 16
      IF(I.GT.IY) GO TO 166
      A(I)=  (T(I,IY-2)-  . (I,IY-1)+3. T(I,IY))/ . /DY
      D(I)=-(T(I+2,K+2)- . T(I+1,K+1)+3. T(I,K))/2./DXY
      GO TO 16
 16   A(I)=(T(I,IY)-T(I+1,IYA))/DXY/ .
      D(I)=A(I)  .
      GO TO 16
 16   A(I)=-(T(I,IYA)-T(I,IY))/DY
```

```
      D(I)=  (T(I+1,K+1)-T(I,K))/DXY
       GO TO 150
145   A(I)=-(T(I,1)-T(I,2))/DX
      B(I)=-(T(I-1,1)-T(I,2))/DXY
        GO TO 150
14[illegible]   A(I)= -(T(I,1)-T(I-1,2))/DXY/ [illegible]
      C(I)=A(I) [illegible]
      GO TO 150
14[illegible]   A(I)=-(T(I,3)-4.*T(I,2)+3.*T(I,1))/2./DY
      B(I)= (T(I,IY-2)-4.*T(I,IY-1)+3.*T(I,IY))/2./DY
      C(I)=A(I)* [illegible]
      D(I)=B(I)* [illegible]
      GO TO 150
13[illegible]   K=IXY+1-I
      D(I)= (T(I-2,K-2)-4.*T(I-1,K-1)+3.*T(I,K))/2./DXY
      A(I)=-(T(I,3)-4.*T(I,2)+3.*T(I,1))/2./DY
150   NUHA=NUHA+DXI*DXY*D(I)/1. [illegible]
      NUCA=NUCA+DXI*DXY*C(I)/1. [illegible]
      NUH = NUH +DXI*DX*A(I)/AR
      NUC = NUC + DXI*DY*B(I)/AR
90    CONTINUE
      PRINT*, "       "
      PRINT*, "       "
      PRINT*, "TIME =  ",TIMEB
      PRINT*, "       "
      DO 91 I=1,IX
      K=IX+1-I
      J=IXY+1-I
      WRITE(6,[illegible]) A(J),C(J),D(K),B(K)
91    CONTINUE
      PRINT*, "       "
      PRINT*, "  AVERAGE NUSSELT NUMBER ( HOT WALL ) =    ",NUH
      PRINT*, "  AVERAGE NUSSELT NUMBER ( COLD WALL ) =  ",NUC
      PRINT*, "  AVERAGE NUSSELT NUMBER ( HOT WALL A) =  ",NUHA
      PRINT*, "  AVERAGE NUSSELT NUMBER ( COLD WALL A) =  ", NUCA
      TIME =TIMEB
      IF(TIME.EQ. [illegible]) GO TO 1
      GO TO 1[illegible]
200   STOP
      END
```

```
      PROGRAM DATAL(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DIMENSION ST(31,31),VOR(31,31),T(31,31),U(31,31),V(31,31),
     1UK(31,31),A(31),B(31),C(31),D(31),(31),E(31),W(31)
      REAL NUC,NUH,NUHS,NUCA
  9   FORMAT(1X,11(F11.5))
  901 FORMAT(2X,F11.5, 5X, F11.5)
C
C     INPUT INITIAL,BOUNDARY CONDITIONS
C
  1   TIME=0.
      READ ,PR,GR,DX,DY,DT,AR,S1,Z
      IF(EOF(5LINPUT) .NE. 0) GO TO 459
      S=S1/57.29578
      AR1 =AR/DX
      IX = AR1 + 1.
      AR2 = 1. /DY
      IY = AR2 + 1.
      IXX=IY 2.-1
C
      PRINT*, "       "
      PRINT*, "       "
      PRINT*,"PRANTL NUMBER = ",PR
      PRINT*, "GRASHOFF NUMBER = ",GR
      PRINT*, " ASPECT RATIO =  ",AR
      PRINT*, "DELTA X = ",DX, "DELTA Y = ",DY
      PRINT*,  "NUMBER OF INCREMENTS  = ",IX
      PRINT*, "Y-INCREMENTS =  ",IY
      PRINT*, "TIME INCREMENT = ",DT
      PRINT*, "INCLINATION ANGLE = ", S1
      PRINT*, "INCLINATION ANGLE = ", S
      PRINT*, "       "
      PRINT*, "       "
C
      DO10 J=1,IY
      M=J
      M2=IYY+1-J
      YJ1=J-1
      YJ2=IY-1
      DO11 I=1,M2
      IF(J.EQ.1) GO TO 16
      IF(I.EQ.M.OR.I.EQ.M2) GO TO 17
      T(I,J) = .
      GO TO 13
  16  T(I,J)= .
      GO TO 13
  17  T(I,J)=YJ1/YJ2
  13  ST(I,J)= .
      VOR(I,J)= .
      U(I,J) = .
      V(I,J) = .
  11  CONTINUE
  10  CONTINUE
      IF(Z.EQ.1. ) GO TO 1 5
      PRINT , "TIME = ",TIME
      PRINT*, "STREAM FUNCTION"
      WRITE(6,9 ) ((ST(I,J),J=1,IY),I=1,IXX)
      PRINT*, "VORTICITY "
```

```
      WRIT? (?,?  ) ((V??(I,J),J=1,?Y),?=1,IXX)
      ?????, " U-V?LOCITY  "
      WRIT? (?,?  ) ((U(I,J),J=1,IY),I=1,IXX)
      PRIN? ?, " V-V?LOCITY  "
      WRIT? (?,?  ) ((V(I,J),J=1,IY),I=1,IXX)
      PRINT ?, " ??MP??ATU?E  "
      WRIT? (?,?  ) ((T(I,J),J=1,IY),I=1,IXX)
C
C     SOLV? ?N??GY ?QUATION FOR TI??=N+1/2
C     T?MP?RATUR? SOLUTION IN X_DI??CTION (INT?RIO? POI???)
C
C
    1 TIM?? = TIM? + ?.? ?DT
      IX?=?X-1
      IY?=?Y-1
      IX?=?X-2
      IY?=IY-2
      ?YX?=?XX-1
      IXX?=?XX-2
      ?XY=???(?X? ?.+?Y  2.)
      A? = ?T/ (??? (?Y? ?2.))
      A?  = ?T /(?.  ?X)
      A?  =?T /(?. ?DY)
      A? = ?.
      A? = ?T/(?? (?X? ?.))
      DO ?  J=?,?Y?
      ?=J+1
      M1=IYY-J
      M?=J
      M?=IYY+1-J
      M?=IYY-((J-1)??.)
      DO ?? I=?,?1
      K=I+? -J
      A(K) =?  -(U(I,J)?A?)
      B(K) =?  +(?.  A? )
      C(K) =(U(I,J)??3) +??
      D(K) =((A?-(V(I,J)???))?T(?,J+1)) +((A?-(?. ??))??(?,J))) +
     1(((V(?,J)??)+?2)?T(I,J-1))
   2?  CONTINUE
C     BOUNDA?Y CONDITIONS       AT X+=? , X+=L/?
      L?=1
      L?=1
      ?1=T(M?,J)
      A?=T(M?,J)
C     SOLV? T?I?DIAGONAL MAT?IX
      CALL GT??(?,?,C,D,??,E,F,W,L?,?1,?1,L?,??,? )
      DO ?? I=?2,?3
      K=I+? -J
      WK(IXX,1)=T(IXX,1)
      WK(1,1)=T(1,1)
      WK(?,?)=T(?,1)
      WK(IY,IY)=T(IY.IY)
      WK(?, J)=W(K)
      IF(J.E?.?) WK(?+1,1)=T(I+1,1)
   ??  CONTINUE
   ?   CONTINUE
      IF(?.E?.?. ) GO TO 1?
```

```
1     M=2
      M1=IXX-1
17    DO 31 J=M,M1
      U(I,J)=(ST(I,J+1)-ST(I,J-1))/(2.*DY)
      V(I,J) =-(ST(I+1,J)-ST(I-1,J))/(2.*DX)
31    CONTINUE
9     CONTINUE
      IF(Z.EQ.1.) GO TO 10
      PRINT*, "VELOCITIES AT TIME= ",TIMER
      PRINT*, "U-VELOCITY "
      DO 32 I=1,IXX
      K=IXY+1-I
      IF(I.GT.IX) GO TO 103
      WRITE(6,8)(U(K,J),J=1,I)
      GO TO 32
103   M4=IXY+1-I
      WRITE(6,8)(U(K,J),J=1,M4)
32    CONTINUE
      PRINT*, "V-VELOCITY "
      DO 34 I=1,IXX
      K=IXY+1-I
      IF(I.GT.IX) GO TO 104
      WRITE(6,8)(V(K,J),J=1,I)
      GO TO 34
104   M4= IXY+1-I
      WRITE(6,8)(V(K,J),J=1,M4)
34    CONTINUE
C
C  CALCULATE NUSSELT NUMBERS AT WALL
C
10    NUH=0.
      NUC=0.
      DO 3 I=1,IXX
      DXI=1.
      IF(I.EQ.1.OR.I.EQ.IXX) DXI=.5
      IF(I.EQ.1.) GO TO 101
      IF(I.EQ.2) GO TO 102
      IF(I.EQ.IXX1) GO TO 103
      IF(I.EQ.IXX) GO TO 104
      IF(I.GT.IX) GO TO 105
      A(I)=-(T(I,3)-4.*T(I,2)+3.*T(I,1))/2./DY
      B(I)=(T(I+2,I-2)-4.*T(I+1,I-1)+3.*T(I,I))/DXY/2.
      GO TO 106
101   A(I)=-(T(I,I)-T(I+1,I+1))/DXY/.707
      B(I)=A(I)/.707
      GO TO 106
102   A(I)= -(T(I,1)-T(I,2))/DY
      B(I)= (T(I,I)-T(I+1,I-1))/DXY
      GO TO 106
103   A(I)=-(T(I,1)-T(I,2))/DX
      B(I)=-(T(I-1,1)-T(I,2))/DXY
         GO TO 106
104   A(I)= -(T(I,1)-T(I-1,2))/DXY/.707
      B(I)=A(I)   ..
      GO TO 106
105   K=IXY+1-I
      B(I)=(T(I-2,K-2)-4.*T(I-1,K-1)+3.*T(I,K))/2./DXY
```

```
      Q(I)=-(T(I,3)-4.*T(I,2)+3.*T(I,1))/2./DY
 1    NUH =NUH +DX1*DX*Q(I)/... /2.
      NUC=NUC+DX1*DXY*Q(I)/1.*1./A  /2.
 3    CONTINUE
      PRINT*, "        "
      PRINT*, "        "
      PRINT*, "TIME =   ",TIMEB
      PRINT*, "        "
      DO 31 I=1,IXX
      K=IXX+1-I
      WRITE(*,6..)..(K),R(K)
 31   CONTINUE
      PRINT*, "        "
      PRINT*, "  AVERAGE NUSSELT NUMBER ( HOT WALL ) =     ", NUH
      PRINT*, "  AVERAGE NUSSELT NUMBER ( COLD WALL )  = ", NUC
      TIME =TIMEB
      IF(TIME .GT. ..1.) GO TO 1..
      GO TO 1..
 3..  STOP
      END
```

VITA

Thomas K. Toltzien was born on May 15, 1949 in Madison, Wisconsin to Paul R. and Jo K. Toltzien. In June 1967, he graduated from Madison West High School. He then attended the University of Wisconsin-Madison, and received a Bachelor of Science degree in Mechanical Engineering in January 1972. He entered the United States Air Force in June 1972, and received a commission through OTS in September 1972. He was then assigned to Mather AFB, Ca. for navigator training. Following graduation in July 1973, he was assigned to the 16th Tactical Reconnaissance Squadron (TRS), Shaw AFB, SC for RF-4C Weapon Systems Officer (WSO) training. Upon completion of training, he was assigned to the 14th TRS, Udorn RTAFB, Thailand as a Mission Ready RF-4C WSO. In April 1975, he was reassigned to the 1st TRS, RAF Alconbury, England. While stationed at RAF Alconbury, he was a RF-4C Instructor Weapon Systems Officer, and a Flight Examiner. Capt. Toltzien was assigned to the United States Air Force Institute of Technology in June 1979 in the Graduate Aeronautical Engineering Program.

Permanent address: 2109 Monroe Street
Madison, Wisconsin 53711

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GAE/AA/80D-23	2. GOVT ACCESSION NO. AD-A103389	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Numerical Solution of Natural Convection in an Inclined Rectangular Cavity with Partitions.		5. TYPE OF REPORT & PERIOD COVERED MS Thesis 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Thomas K. Toltzien, Capt., USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology/AFIT-EN Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE December 1980 13. NUMBER OF PAGES 94
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Approved for public release; distribution unlimited.

Frederick C. Lynch

Frederick Lynch, Major, USAF
Director of Public Affairs 20 AUG 1981

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Natural Convection
Partitioned Cavity
Rectangular Enclosure
Triangular Enclosure
Numerical Solution
Composite Solutions

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A numerical investigation was conducted on two-dimensional natural convection within inclined rectangular enclosures partitioned into 45 degree triangular cells. The time dependent governing equations, vorticity, energy, and stream function, were solved by an ADI method and a Gauss-Seidel SOR technique. The numerical procedure was validated for rectangular enclosure, then modified for triangular cells. Heat transfer coefficients were determined for an inclined

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD-A103 389 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 10/1
NUMERICAL SOLUTION OF NATURAL CONVECTION IN AN INCLINED RECTANG--ETC(U)
DEC 80 T K TOLTZIEN
UNCLASSIFIED AFIT/GAE/AA/80D-23 NL

2 OF 2
AD A
103389

END
DATE
FILMED
10-81
DTIC

square enclosure with a diagonal partition for Grashof numbers less than 2. x 10^5 , and inclination angles between 10 degrees and 90 degrees. These results show a diagonal partition reduces the heat transferred by natural convection across an inclined square enclosure by more than 50%.