

A Note on the Stability and Accuracy of Finite Difference Approximations to Differential Equations

L.D. Cloutman

RECEIVED
DEC 09 1996
OSTI

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

LD
September 1996



Lawrence
Livermore
National
Laboratory

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the Laboratory.
Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract W-7405-ENG-48.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This report has been reproduced
directly from the best available copy.

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information
P.O. Box 62, Oak Ridge, TN 37831
Prices available from (615) 576-8401, FTS 626-8401

Available to the public from the
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd.,
Springfield, VA 22161

DISCLAIMER

**Portions of this document may be illegible
in electronic image products. Images are
produced from the best available original
document.**

A NOTE ON THE STABILITY AND ACCURACY OF FINITE
DIFFERENCE APPROXIMATIONS TO DIFFERENTIAL EQUATIONS

Lawrence D. Cloutman
Lawrence Livermore National Laboratory
Livermore, California

ABSTRACT

There are many finite difference approximations to ordinary and partial differential equations, and these vary in their accuracy and stability properties. We examine selected commonly used methods and illustrate their stability and accuracy using both linear stability analysis and numerical examples. We find that the formal order of accuracy alone gives an incomplete picture of the accuracy of the method. Specifically, the Adams-Bashforth and Crank-Nicholson methods are shown to have some undesirable features for both ordinary and partial differential equations.

I. INTRODUCTION

There seems to be a widespread misperception that high order accuracy and implicitness (which often confers unconditional stability) implies that numerical solutions will be accurate approximations to solutions of the original differential equations. In this report we demonstrate that this article of faith is often violated in practice, and that the real situation is far more complex and subtle than is often appreciated. We shall show simple examples where stable solutions can be quite inaccurate, possibly including pathological features. In some examples, the lower order method is shown to be more accurate.

We consider three cases to illustrate the point. The first is a comparison of Euler's method (first order) and the Adams-Bashforth method (second order) for a pair of simple ordinary differential equations. Conventional wisdom would have it that the former method is inferior in accuracy. The second example is a simple diffusion equation solved by an implicit method that is second order in space and either first or second order in time depending on the value of a time centering parameter. A special case is Crank-Nicholson differencing, which is second order. We demonstrate that this method has undesirable features at sufficiently large time step in spite of its formal unconditional stability. The third case is a demonstration of unconditional instability in two approximations to the first order wave equation.

The next section presents a discussion of the Adams-Bashforth method applied to ordinary differential equations. Section III explores the accuracy and stability of some common differencing methods for the diffusion equation. Section IV illustrates instabilities in the solution of the first order wave equation. Section V contains the summary and conclusions.

II. ORDINARY DIFFERENTIAL EQUATIONS

Let us consider the Adams-Bashforth method for integrating ordinary differential equations. Consider the equation

$$\frac{dy}{dx} = f(y, x). \quad (1)$$

The Adams-Bashforth method is

$$y_{j+1} = y_j + \frac{h}{2} [3f(y_j, x_j) - f(y_{j-1}, x_{j-1})], \quad (2)$$

where h is the step size in x , and y_j is the numerical approximation to $y(x_j) = y(j \times h)$. This method is second-order. We shall compare this method with the simple first-order Euler method,

$$y_{j+1} = y_j + h f(y_j, x_j). \quad (3)$$

First we perform a stability analysis on both methods. We apply both difference approximations to the linear equation

$$\frac{dy}{dx} = -Ky, \quad K > 0, \quad (4)$$

which has the solution

$$y(x) = y(0) \exp(-Kx). \quad (5)$$

Substituting Eq. 4 into Eqs. 2 and 3, we obtain

$$y_{j+1} = y_j - \frac{hK}{2} (3y_j - y_{j-1}) \quad (6)$$

and

$$y_{j+1} = y_j - hK y_j. \quad (7)$$

Next we look for solutions of the form $y_j = r^j$. Making this substitution into Eq. 6 and letting $H = hK$, we find

$$r^2 + (1.5H - 1)r - 0.5H = 0, \quad (8)$$

This equation has 2 solutions,

$$r_{\pm} = \frac{1 - 1.5H \pm [(1 - 1.5H)^2 + 2H]^{1/2}}{2}. \quad (9)$$

In the limit $H \ll 1$,

$$r_{\pm} \approx 1 - H, \quad -0.5H. \quad (10)$$

The general solution of Eq. 6 in this limit is

$$y_j = A(1 - H)^j + B(-0.5H)^j, \quad (11)$$

where A and B are constants of integration. For $H < 1$, the first term decays monotonically, just as the solution of the differential equation does. The second solution introduces nonphysical oscillatory behavior into the numerical solution, even for small H , which is an undesirable feature of the Adams-Bashforth method. A more careful examination of Eq. 9 shows that the exact formal stability limit is $H = 1$; for larger H , $r_- < -1$ and the numerical solution grows without bound.

The analogous analysis of Eq. 7 shows that the difference equation has only one solution with

$$r = 1 - H. \quad (12)$$

The solution becomes oscillatory for $H > 1$, and the method becomes unstable for $H > 2$. We see the Euler method is qualitatively well-behaved for $H < 1$, whereas the Adams-Bashforth method has a nonphysical oscillatory parasitic solution for all H , no matter how close to zero.

Now we consider a nonlinear example, the logistic equation. This equation is widely used as a particularly simple population dynamics model. It can also be used as a simple surrogate for a nonlinear chemical rate equation.

$$\frac{dN}{dt} = \alpha N - \beta N^2 = \alpha N \left(\frac{\mathcal{K} - N}{\mathcal{K}} \right), \quad (13)$$

where N is the population, and $\mathcal{K} = \alpha/\beta$ is the carrying capacity of the environment. The first term on the right hand side represents the familiar exponential growth of the population via a constant difference between birth and mortality rates. The nonlinear term is one plausible representation of an increase in mortality or decrease in fertility due to crowding.

The logistic equation has a simple general solution. At first glance this equation appears to have two free parameters, but it has a self-similar solution with no parameters. Let us make the linear changes of variables $\mathcal{N} = N/\mathcal{K}$ and $\tau = \alpha t + t_0$, where t_0 is a constant adjusted to make $\tau = 0$ at the start of the problem. Then Eq. 13 becomes

$$\frac{d\mathcal{N}}{d\tau} = \mathcal{N} - \mathcal{N}^2. \quad (14)$$

We assume that at $\tau = 0$ we know the value of $\mathcal{N}(0)$. Equation 14 is easily solved by separation of variables, which gives

$$\mathcal{N}(\tau) = \frac{\mathcal{N}(0)}{\mathcal{N}(0) + [1 - \mathcal{N}(0)] \exp(-\tau)}. \quad (15)$$

This solution is valid provided the interval of integration of Eq. 14 over \mathcal{N} does not include either zero or unity, where the integrand has singularities. These singularities are related to two constant equilibrium solutions, $\mathcal{N} = 0$ and 1.

The ecologically interesting case is $0 < \mathcal{N}(0) < 1$. In this case, the solution rises monotonically to unity at late times. The solution does not overshoot the carrying capacity. The equilibrium solution $\mathcal{N}(\tau) = 1$ is stable. If the equilibrium is perturbed by decreasing the population, \mathcal{N} grows monotonically back to unity. If the perturbation causes the population to exceed the carrying capacity, it decays monotonically back to equilibrium.

The equilibrium solution at $\mathcal{N} = 0$ is unstable. For a positive perturbation (that is, the creation of a new species), the solution follows Eq. 15 to unity. For a negative perturbation, the formal solution diverges to $-\infty$ in a finite time. The negative perturbation is not realizable physically for either chemistry or population dynamics.

To illustrate these theoretical results, we now show some simple numerical examples. We solve both Eqs. 4 and 14 using both Eqs. 2 and 3. The listing of the Fortran computer program is given in Table 1, and selected numerical results are shown in Table 2. The listing shows the code as used to solve the differential equations with the Adams-Bashforth method,

Eq. 2. The complete listing is provided for the purpose of completely, precisely, and concisely documenting how the calculations were performed. It is trivial to change the calculations of the variables y_j and z_j to use Euler's method, Eq. 3, simply by using the statements that have been commented out.

The output in Table 2 shows four cases. The first is the Adams-Bashforth method with $H = \text{delx} = 0.9$, which should be stable according to the linear stability analysis. The table provides both the numerical and exact solutions for each differential equation. We see that both the decaying exponential and the logistic numerical solutions show the oscillatory parasitic solution. For the exponential, the amplitude of the oscillations is several orders of magnitude larger than the exact solution, a highly undesirable result. The oscillations in the logistic solution are seen around $J = 22$ and are not large. For large J , the oscillations damp out in both solutions and the correct steady state is approached. The next page shows the same problem run with the Euler technique. There are no oscillations. The time marching is not terribly accurate, especially for the exponential. Accuracy of the logistic solution is not all that bad considering the large step size. The $y = 0.5$ point is off by about 3 steps out of 10. At least the method preserves the qualitative features of the analytic solutions.

The next two cases are for $H = 1.05$, for which Adams-Bashforth is linearly unstable and Euler should exhibit damped oscillations. That is precisely what is observed for the exponential decay solutions. Note that the Euler method's oscillations are much smaller than the magnitude of the exact solution, which is the opposite of what was seen with Adams-Bashforth with $H = 0.9$. The logistic solutions are far more interesting. The Euler solution has a slight oscillation around $J = 13$, but is otherwise well-behaved. The Adams-Bashforth solution is initially well-behaved, but it develops a growing oscillatory mode that can be seen starting around $J = 12$. However, the oscillation saturates around $J = 100$, whereas the linear stability analysis predicts that it should grow without bound.

This unanticipated behavior is easily explained. These two finite difference methods are nonlinear iterated maps. Furthermore, they are related to the well-studied logistic map. It is well known that as one increases the parameter in the logistic map, the long-term solution changes from a steady state to a set of periodic solutions (that exhibit period doubling) to chaotic solutions to an unbounded solution. We hypothesize that what we have seen here is the first periodic solution on the route to chaotic solutions, and finally unbounded solutions, as H is increased (Yee *et al.* 1991). This question will be examined in more detail in a future report.

III. THE DIFFUSION EQUATION

The diffusion equation is

$$\frac{\partial T}{\partial t} = K \frac{\partial^2 T}{\partial x^2}, \quad (16)$$

where K is the constant thermal diffusivity. Consider the finite difference approximation

$$T_j^{n+1} = T_j^n + \frac{K \delta t}{\delta x^2} \left[(1 - \phi)(T_{j+1}^n - 2T_j^n + T_{j-1}^n) + \phi(T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}) \right], \quad (17)$$

where T_j^n is the value of T at grid point j and time step n , δt is the time step, δx is the space step, and $0 \leq \phi \leq 1$ is the time-centering parameter. For $\phi = 0.5$, the method is second order in both space and time. Stability analysis is performed by considering solutions of the form

$$T_j^n = r^n \exp(i j k \delta x), \quad (18)$$

where $i = -1^{1/2}$ and k is the wavenumber of the mode. Substituting Eq. 18 into Eq. 17, we find

$$r = \frac{1 + 2(1 - \phi)\xi[\cos(k \delta x) - 1]}{1 - 2\phi\xi[\cos(k \delta x) - 1]} = \frac{1 - 4(1 - \phi)\xi}{1 + 4\phi\xi}, \quad (19)$$

where $\xi = K \delta t / \delta x^2$. We have made use of the fact that the most unstable mode is for $\cos(k \delta x) = -1$. There are three interesting special cases. First, the explicit case with $\phi = 0$, which is unstable if $\xi > 0.5$. Second, the implicit case with $\phi = 1$, which is unconditionally stable. Furthermore, the worst mode decays monotonically for all ξ . Third, the Crank-Nicholson method with $\xi = 0.5$, which is formally unconditionally stable. However, its solutions can contain a nonphysical oscillatory component for $\xi > 1$.

Table 3 lists the Fortran code that was used to produce the numerical examples given in Table 4. The boundary conditions are zero gradient on the left boundary ($J = 1.5$) and $T = 100$ on the right boundary ($J = 46.5$). The initial condition is $T = 100$ in zones 1-15 and $T = 300$ elsewhere. This problem has two discontinuities, one at $J = 15$ and the other at the right boundary. Zones 1 and 47 are fictitious zones used for imposing boundary conditions. The column labels indicate the numerical parameters used in the calculation. For example, T(CN,10.0) means that the column contains values of T calculated with Crank-Nicholson ($\phi = 0.5$) and $\xi = 10.0$. The indicator IM means the solution was computed with the fully implicit method, $\phi = 1.0$. Solutions are shown for $\xi = 0.5$ and 10.0 at times of 0, 50, 100, and 500. Both methods get about the same answers for $\xi = 0.5$, which is the explicit stability limit. This is not too surprising, even though one method is first order and the other is second order, since the explicit stability limit is also a reasonable accuracy criterion for time marching. At TIME = 50, the Crank-Nicholson solution with $\xi = 10.0$ shows large nonphysical oscillations in the vicinity of both initial discontinuities. Taking the two

solutions with $\xi = 0.5$ to be correct to the extent that they agree with one another, we see that in the neighborhood of the initial discontinuities, the first-order implicit method is more accurate. In the smooth part of the solution, the Crank Nicholson method is excellent, but the errors in the implicit method are only a couple of percent. As expected, the oscillations slowly damp out at late times. However, the first order implicit scheme never develops either large errors or nonphysical behavior. The slow damping of the every-other zone mode is unacceptable for large eddy simulations (LES) or direct numerical simulations (DNS) of turbulent flows because, physically, the diffusion operator for the eddy viscosity terms will damp the highest wavenumbers the fastest, and that property of the differential equation does not carry over to the Crank-Nicholson difference operator at time steps significantly larger than the explicit stability limit.

IV. THE FIRST ORDER WAVE EQUATION

The Adams-Bashforth method is sometimes advocated for treating advection terms in fluid dynamics codes. A simple surrogate problem is the linear wave equation

$$\frac{\partial T}{\partial t} + c \frac{\partial T}{\partial x} = 0, \quad (20)$$

where $c > 0$ is the constant advection speed and T is the dependent variable, for example, the fluid temperature. Any differentiable function $f(x - ct)$ is a solution to this equation. We wish to point out that spatially centered differencing of Eq. 20 has some severe stability problems and should never be used without a careful stability analysis. So, let us consider both the Euler method and Adams-Bashforth for time marching and centered differences in space. Then the difference equations are

$$\frac{T_j^{n+1} - T_j^n}{\delta t} + c \frac{T_{j+1}^n - T_{j-1}^n}{2 \delta x} = 0 \quad (21)$$

and

$$\frac{T_j^{n+1} - T_j^n}{\delta t} + c \frac{1.5 (T_{j+1}^n - T_{j-1}^n) - 0.5 (T_{j+1}^{n-1} - T_{j-1}^{n-1})}{2 \delta x} = 0. \quad (22)$$

Again, stability analysis is performed by considering solutions of the form

$$T_j^n = r^n \exp(i j k \delta x), \quad (23)$$

where $i = -1^{1/2}$ and k is the wavenumber of the mode. Substituting Eq. 23 into Eq. 21, we obtain

$$r = 1 - i \beta \sin(k \delta x) \quad (24)$$

and

$$rr^* = 1 + \beta^2 \sin^2(k \delta x), \quad (25)$$

where $\beta = c \delta t / \delta x$. Clearly, this method is unconditionally unstable. No matter how small we make the time step, the numerical solution will eventually blow up. The most unstable mode has a wavelength of $4 \delta x$. Interestingly, the every-other-zone mode is neutrally stable. If we apply the same analysis to Eq. 22, we obtain a quadratic equation for two growth rates,

$$r^2 - \left[1 - \frac{3i\beta}{2} \sin(k \delta x) \right] r - \frac{i\beta}{2} \sin(k \delta x) = 0. \quad (26)$$

If we take $\sin(k \delta x) = 0$ (the wavelength of the mode is either infinite or equal to twice the zone size), then the values of r are zero and unity, and the method is neutrally stable. If we take $\sin(k \delta x) = 1$, then

$$r = \frac{1 - 1.5i\beta \pm (1 - i\beta - 2.25\beta^2)^{1/2}}{2}. \quad (27)$$

Table 5 shows values of r and rr^* for various values of β . These were evaluated numerically with a short Fortran program using complex arithmetic (also presented in Table 5), so they are accurate to about 6 digits. The method is neutrally stable for $\beta = 0$. For larger values of β , the method is weakly unstable. For computational efficiency, we want to run fluid dynamical calculations with β around 0.2. So let us look at a simple test problem with $\beta = 0.1$.

Table 6 lists a simple Fortran code that calculates the numerical examples given in Table 7. The solution is a sine wave of wavelength $4 \delta x$ and amplitude 1.0. Periodic boundary conditions are used, so the analytic solution simply propagates the sine wave to the right with $c = 1.0$ without a change of shape or amplitude. The analytic solution is constrained to have values between -1.0 and +1.0 at all grid points at all times. Table 7 shows solutions for the 12 zone solutions using both Euler's method and the Adams-Bashforth method at several different times. Both solutions develop nonphysical numerical oscillations during the first few cycles. The Euler's method solution has a growth rate of 1.01, and it blows up very quickly. The Adams-Bashforth solution has a theoretical growth rate of 1.000051, which is consistent with what we observe in this example. For thousands of cycles, we see no clear evidence of the unbounded growth because it is masked by numerical oscillations caused by dispersive truncation errors. However, by cycle 100,000, the amplitude is clearly outside the physical limit of unity and is growing.

There is a clear warning here. Consider Lax's Equivalence Theorem: Given a properly posed initial-value problem and a finite-difference approximation to it that satisfies the consistency condition, stability is the necessary and sufficient condition for convergence

(Richtmyer 1957). Both the Euler method and the Adams-Bashforth method satisfy the consistency condition, but they are unstable. Therefore, they are not convergent. There is little danger of the Euler time marching being used due to its strong instability. However, the weak instability of Adams-Bashforth may be overlooked if problems are run for too short a time for the instability to manifest itself. The real danger is that a nonconvergent solution may be accepted as an approximate solution to the differential equation. Since this problem will carry over to numerical solution of the Euler equations, Adams-Bashforth time marching and centered differencing of the advection terms should be avoided.

V. CONCLUSIONS

The purpose of this study is to show that some elementary finite difference methods have some undesirable properties that should preclude their use in computational fluid dynamics. This is especially true for LES and DNS, which make severe demands on numerical methods. In both cases, we are trying to obtain significant results near the grid scale. Since small scale oscillatory flow features (vortices) are continuously driven by the turbulence, we cannot recommend the use of Adams-Bashforth differencing because it can produce unphysical oscillations in time that switch sign on alternate time steps.

We also recommend prudence in use of the Crank-Nicholson method for treating the diffusion terms. This method has spurious oscillations unless the time step is kept below twice the explicit stability limit (which is also a good accuracy limit). Given the increased amount of arithmetic needed to implement implicit methods, it may in fact be more efficient to use the explicit method. One could eliminate the oscillation problem by going fully implicit, but accuracy will suffer if large time steps are used unless the solution is nearly steady.

ACKNOWLEDGMENTS

I thank John Ramshaw and Nick Marinov for helpful discussions. This work was performed under the auspices of the U. S. Department of Energy and the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48.

REFERENCES

R. D. Richtmyer, "Difference Methods for Initial-Value Problems," Interscience Publishers, New York, 1957.

H. C. Yee, P. K. Sweby, and D. F. Griffiths, "Dynamical Approach Study of Spurious Steady-State Numerical Solutions of Nonlinear Differential Equations. I. The Dynamics of Time Discretization and Its Implications for Algorithm Development in Computational Fluid Dynamics," *J. Comput. Phys.* **97**, 249-310 (1991).

Table 1.
Adams-Bashforth ODE Program

Test Program for Ordinary Differential Equations

```

program adams
c +++ Adams-Bashforth testing
  implicit double precision (a-h,o-z)
  open (unit=12,file='adams.out',form='formatted')
c +++ initialize parameters
  delx = 0.9d+00
  npts = 200
  j = 0
  yj = 1.d+00
c +++ Euler first step
  yjm = yj
c +++ Exact solution for first step
  yjm = exp(delx)
  yjp = yj
  yex = yj
  zj = 1.d-03
  zjp = zj
  zex = zj
  z0 = zj
c +++ Euler first step
  zjm = zj
c +++ Exact solution for first step
  zjm = z0 / (z0 + (1.d+00 - z0) * yjm)

  write (12,10) delx
  10 format (/ ' Adams-Bashforth tests for delx =',1p,d16.8/
  1 ' j x exponential logistic'/
  2 ' numerical exact numerical',
  3 ' exact')
  write (12,20) j, x, yjp, yex, zjp, zex
  20 format (1x,i4,1p,5d14.6)

  do j = 1, npts
c +++ dy/dh = -y
ccc   yjp = yj - delx * yj
      yjp = yj - delx * (1.5d+00 * yj - 0.5d+00 * yjm)

c +++ dz/dh = z - z**2
ccc   zjp = zj + delx * (zj - zj**2)
      zjp = zj + delx * (1.5d+00 * (zj - zj**2)
      1 - 0.5d+00 * (zjm - zjm**2))

c +++ exact solutions
  x = real(j) * delx
  yex = exp(-x)
  zex = z0 / (z0 + (1.d+00 - z0) * yex)
  write (12,20) j, x, yjp, yex, zjp, zex

```

```
yjm = yj  
yj = yjp  
zjm = zj  
zj = zjp
```

```
enddo
```

```
close(unit=12)  
stop  
end
```

Table 2.
Ordinary Differential Equation Solutions

Exponential Decay and Logistic Equations					
Adams-Bashforth tests for delx = 9.00000000D-01					
j	x	exponential numerical	exact	logistic numerical	exact
0	0.000000D+00	1.000000D+00	1.000000D+00	1.000000D-03	1.000000D-03
1	9.000000D-01	7.568214D-01	4.065697D-01	2.165659D-03	2.456018D-03
2	1.800000D+00	1.851125D-01	1.652989D-01	4.633418D-03	6.019252D-03
3	2.700000D+00	2.757803D-01	6.720551D-02	9.887114D-03	1.467603D-02
4	3.600000D+00	-1.322246D-02	2.732372D-02	2.102737D-02	3.534019D-02
5	4.500000D+00	1.287290D-01	1.110900D-02	4.441221D-02	8.265906D-02
6	5.400000D+00	-5.100525D-02	4.516581D-03	9.244254D-02	1.814202D-01
7	6.300000D+00	7.577987D-02	1.836305D-03	1.866055D-01	3.527998D-01
8	7.200000D+00	-4.947532D-02	7.465858D-04	3.537601D-01	5.727904D-01
9	8.100000D+00	5.141730D-02	3.035391D-04	5.940861D-01	7.673210D-01
10	9.000000D+00	-4.025995D-02	1.234098D-04	8.167594D-01	8.902449D-01
11	9.900000D+00	3.722877D-02	5.017468D-05	9.102886D-01	9.522680D-01
12	1.080000D+01	-3.114705D-02	2.039950D-05	9.531854D-01	9.800279D-01
13	1.170000D+01	2.765441D-02	8.293819D-06	9.766780D-01	9.917826D-01
14	1.260000D+01	-2.369522D-02	3.372015D-06	9.873481D-01	9.966427D-01
15	1.350000D+01	2.073781D-02	1.370959D-06	9.939619D-01	9.986323D-01
16	1.440000D+01	-1.792108D-02	5.573904D-07	9.964428D-01	9.994435D-01
17	1.530000D+01	1.560439D-02	2.266180D-07	9.985272D-01	9.997737D-01
18	1.620000D+01	-1.352602D-02	9.213601D-08	9.989175D-01	9.999080D-01
19	1.710000D+01	1.175609D-02	3.745971D-08	9.997155D-01	9.999626D-01
20	1.800000D+01	-1.020134D-02	1.522998D-08	9.996129D-01	9.999848D-01
21	1.890000D+01	8.860708D-03	6.192048D-09	1.000007D+00	9.999938D-01
22	1.980000D+01	-7.691851D-03	2.517499D-09	9.998233D-01	9.999975D-01
23	2.070000D+01	6.679466D-03	1.023539D-09	1.000065D+00	9.999990D-01
24	2.160000D+01	-5.799146D-03	4.161397D-10	9.998977D-01	9.999996D-01
25	2.250000D+01	5.035461D-03	1.691898D-10	1.000065D+00	9.999998D-01
26	2.340000D+01	-4.372027D-03	6.878744D-11	9.999312D-01	9.999999D-01
27	2.430000D+01	3.796167D-03	2.796688D-11	1.000053D+00	1.000000D+00
28	2.520000D+01	-3.296071D-03	1.137049D-11	9.999504D-01	1.000000D+00
29	2.610000D+01	2.861900D-03	4.622895D-12	1.000041D+00	1.000000D+00
30	2.700000D+01	-2.484897D-03	1.879529D-12	9.999632D-01	1.000000D+00
31	2.790000D+01	2.157569D-03	7.641594D-13	1.000032D+00	1.000000D+00
32	2.880000D+01	-1.873353D-03	3.106840D-13	9.999724D-01	1.000000D+00
33	2.970000D+01	1.626579D-03	1.263147D-13	1.000024D+00	1.000000D+00
34	3.060000D+01	-1.412311D-03	5.135572D-14	9.999792D-01	1.000000D+00
35	3.150000D+01	1.226270D-03	2.087968D-14	1.000018D+00	1.000000D+00
36	3.240000D+01	-1.064735D-03	8.489044D-15	9.999844D-01	1.000000D+00
195	1.755000D+02	1.876946D-13	6.043916D-77	1.000000D+00	1.000000D+00
196	1.764000D+02	-1.629698D-13	2.457273D-77	1.000000D+00	1.000000D+00
197	1.773000D+02	1.415020D-13	9.990526D-78	1.000000D+00	1.000000D+00
198	1.782000D+02	-1.228621D-13	4.061845D-78	1.000000D+00	1.000000D+00
199	1.791000D+02	1.066777D-13	1.651423D-78	1.000000D+00	1.000000D+00
200	1.800000D+02	-9.262514D-14	6.714184D-79	1.000000D+00	1.000000D+00

Euler tests for delx = 9.0000000D-01

j	x	exponential		logistic	
		numerical	exact	numerical	exact
0	0.000000D+00	1.000000D+00	1.000000D+00	1.000000D-03	1.000000D-03
1	9.000000D-01	1.000000D-01	4.065697D-01	1.899100D-03	2.456018D-03
2	1.800000D+00	1.000000D-02	1.652989D-01	3.605044D-03	6.019252D-03
3	2.700000D+00	1.000000D-03	6.720551D-02	6.837887D-03	1.467603D-02
4	3.600000D+00	1.000000D-04	2.732372D-02	1.294990D-02	3.534019D-02
5	4.500000D+00	1.000000D-05	1.110900D-02	2.445389D-02	8.265906D-02
6	5.400000D+00	1.000000D-06	4.516581D-03	4.592419D-02	1.814202D-01
7	6.300000D+00	1.000000D-07	1.836305D-03	8.535784D-02	3.527998D-01
8	7.200000D+00	1.000000D-08	7.465858D-04	1.556225D-01	5.727904D-01
9	8.100000D+00	1.000000D-09	3.035391D-04	2.738863D-01	7.673210D-01
10	9.000000D+00	1.000000D-10	1.234098D-04	4.528716D-01	8.902449D-01
11	9.900000D+00	1.000000D-11	5.017468D-05	6.758726D-01	9.522680D-01
12	1.080000D+01	1.000000D-12	2.039950D-05	8.730346D-01	9.800279D-01
13	1.170000D+01	1.000000D-13	8.293819D-06	9.727953D-01	9.917826D-01
14	1.260000D+01	1.000000D-14	3.372015D-06	9.966134D-01	9.966427D-01
15	1.350000D+01	1.000000D-15	1.370959D-06	9.996510D-01	9.986323D-01
16	1.440000D+01	1.000000D-16	5.573904D-07	9.999650D-01	9.994435D-01
17	1.530000D+01	1.000000D-17	2.266180D-07	9.999965D-01	9.997737D-01
18	1.620000D+01	1.000000D-18	9.213601D-08	9.999996D-01	9.999080D-01
19	1.710000D+01	1.000000D-19	3.745971D-08	1.000000D+00	9.999626D-01
20	1.800000D+01	1.000000D-20	1.522998D-08	1.000000D+00	9.999848D-01
21	1.890000D+01	1.000000D-21	6.192048D-09	1.000000D+00	9.999938D-01
22	1.980000D+01	1.000000D-22	2.517499D-09	1.000000D+00	9.999975D-01
23	2.070000D+01	1.000000D-23	1.023539D-09	1.000000D+00	9.999990D-01
24	2.160000D+01	1.000000D-24	4.161397D-10	1.000000D+00	9.999996D-01
25	2.250000D+01	1.000000D-25	1.691898D-10	1.000000D+00	9.999998D-01
26	2.340000D+01	1.000000D-26	6.878744D-11	1.000000D+00	9.999999D-01
27	2.430000D+01	1.000000D-27	2.796688D-11	1.000000D+00	1.000000D+00
28	2.520000D+01	1.000000D-28	1.137049D-11	1.000000D+00	1.000000D+00
29	2.610000D+01	1.000000D-29	4.622895D-12	1.000000D+00	1.000000D+00
30	2.700000D+01	1.000000D-30	1.879529D-12	1.000000D+00	1.000000D+00
31	2.790000D+01	1.000000D-31	7.641594D-13	1.000000D+00	1.000000D+00
32	2.880000D+01	1.000000D-32	3.106840D-13	1.000000D+00	1.000000D+00
33	2.970000D+01	1.000000D-33	1.263147D-13	1.000000D+00	1.000000D+00
34	3.060000D+01	1.000000D-34	5.135572D-14	1.000000D+00	1.000000D+00
35	3.150000D+01	1.000000D-35	2.087968D-14	1.000000D+00	1.000000D+00
36	3.240000D+01	1.000000D-36	8.489044D-15	1.000000D+00	1.000000D+00
37	3.330000D+01	1.000000D-37	3.451388D-15	1.000000D+00	1.000000D+00
38	3.420000D+01	1.000000D-38	1.403230D-15	1.000000D+00	1.000000D+00
39	3.510000D+01	1.000000D-39	5.705106D-16	1.000000D+00	1.000000D+00
40	3.600000D+01	1.000000D-40	2.319523D-16	1.000000D+00	1.000000D+00
192	1.728000D+02	1.000000-192	8.993185D-76	1.000000D+00	1.000000D+00
193	1.737000D+02	1.000000-193	3.656356D-76	1.000000D+00	1.000000D+00
194	1.746000D+02	1.000000-194	1.486563D-76	1.000000D+00	1.000000D+00
195	1.755000D+02	1.000000-195	6.043916D-77	1.000000D+00	1.000000D+00
196	1.764000D+02	1.000000-196	2.457273D-77	1.000000D+00	1.000000D+00
197	1.773000D+02	1.000000-197	9.990526D-78	1.000000D+00	1.000000D+00
198	1.782000D+02	1.000000-198	4.061845D-78	1.000000D+00	1.000000D+00
199	1.791000D+02	1.000000-199	1.651423D-78	1.000000D+00	1.000000D+00
200	1.800000D+02	1.000000-200	6.714184D-79	1.000000D+00	1.000000D+00

Adams-Bashforth tests for delx = 1.05000000D+00

j	x	exponential numerical	exact	logistic numerical	exact
0	0.000000D+00	1.000000D+00	1.000000D+00	1.000000D-03	1.000000D-03
1	1.050000D+00	9.252668D-01	3.499377D-01	2.389653D-03	2.852352D-03
2	2.100000D+00	-7.028431D-03	1.224564D-01	5.619886D-03	8.108066D-03
3	3.150000D+00	4.898064D-01	4.285213D-02	1.316989D-02	2.282622D-02
4	4.200000D+00	-2.853286D-01	1.499558D-02	3.070544D-02	6.257595D-02
5	5.250000D+00	4.212123D-01	5.247518D-03	7.075843D-02	1.601981D-01
6	6.300000D+00	-3.919946D-01	1.836305D-03	1.586919D-01	3.527998D-01
7	7.350000D+00	4.465334D-01	6.425924D-04	3.344487D-01	6.090320D-01
8	8.400000D+00	-4.625539D-01	2.248673D-04	6.149402D-01	8.165649D-01
9	9.450000D+00	5.003985D-01	7.868957D-05	8.710213D-01	9.271184D-01
10	1.050000D+01	-5.305699D-01	2.753645D-05	9.236477D-01	9.732276D-01
11	1.155000D+01	5.677869D-01	9.636043D-06	9.757406D-01	9.904654D-01
12	1.260000D+01	-6.050267D-01	3.372015D-06	9.759979D-01	9.966427D-01
13	1.365000D+01	6.459785D-01	1.179995D-06	1.000467D+00	9.988226D-01
14	1.470000D+01	-6.890766D-01	4.129249D-07	9.874327D-01	9.995877D-01
15	1.575000D+01	7.353578D-01	1.444980D-07	1.007223D+00	9.998557D-01
16	1.680000D+01	-7.845960D-01	5.056531D-08	9.892499D-01	9.999495D-01
17	1.785000D+01	8.372055D-01	1.769471D-08	1.009819D+00	9.999823D-01
18	1.890000D+01	-8.933060D-01	6.192048D-09	9.886194D-01	9.999938D-01
19	1.995000D+01	9.531839D-01	2.166831D-09	1.011545D+00	9.999978D-01
20	2.100000D+01	-1.017066D+00	7.582560D-10	9.872447D-01	9.999992D-01
21	2.205000D+01	1.085235D+00	2.653424D-10	1.013209D+00	9.999997D-01
22	2.310000D+01	-1.157970D+00	9.285333D-11	9.855188D-01	9.999999D-01
23	2.415000D+01	1.235581D+00	3.249288D-11	1.015023D+00	1.000000D+00
24	2.520000D+01	-1.318393D+00	1.137049D-11	9.835139D-01	1.000000D+00
25	2.625000D+01	1.406756D+00	3.978963D-12	1.017057D+00	1.000000D+00
26	2.730000D+01	-1.501041D+00	1.392389D-12	9.812215D-01	1.000000D+00
27	2.835000D+01	1.601646D+00	4.872495D-13	1.019350D+00	1.000000D+00
28	2.940000D+01	-1.708993D+00	1.705070D-13	9.786105D-01	1.000000D+00
29	3.045000D+01	1.823535D+00	5.966684D-14	1.021934D+00	1.000000D+00
30	3.150000D+01	-1.945754D+00	2.087968D-14	9.756412D-01	1.000000D+00
31	3.255000D+01	2.076164D+00	7.306588D-15	1.024840D+00	1.000000D+00
32	3.360000D+01	-2.215315D+00	2.556851D-15	9.722686D-01	1.000000D+00
33	3.465000D+01	2.363792D+00	8.947387D-16	1.028099D+00	1.000000D+00
34	3.570000D+01	-2.522221D+00	3.131028D-16	9.684443D-01	1.000000D+00
35	3.675000D+01	2.691268D+00	1.095665D-16	1.031743D+00	1.000000D+00
36	3.780000D+01	-2.871645D+00	3.834145D-17	9.641170D-01	1.000000D+00
37	3.885000D+01	3.064112D+00	1.341712D-17	1.035799D+00	1.000000D+00
38	3.990000D+01	-3.269478D+00	4.695158D-18	9.592348D-01	1.000000D+00
39	4.095000D+01	3.488608D+00	1.643013D-18	1.040290D+00	1.000000D+00
40	4.200000D+01	-3.722426D+00	5.749522D-19	9.537474D-01	1.000000D+00
192	2.016000D+02	-7.132576D+04	2.794039D-88	8.237351D-01	1.000000D+00
193	2.026500D+02	7.610623D+04	9.777397D-89	1.128646D+00	1.000000D+00
194	2.037000D+02	-8.120710D+04	3.421480D-89	8.237351D-01	1.000000D+00
195	2.047500D+02	8.664986D+04	1.197305D-89	1.128646D+00	1.000000D+00
196	2.058000D+02	-9.245740D+04	4.189822D-90	8.237351D-01	1.000000D+00
197	2.068500D+02	9.865418D+04	1.466177D-90	1.128646D+00	1.000000D+00
198	2.079000D+02	-1.052663D+05	5.130707D-91	8.237351D-01	1.000000D+00
199	2.089500D+02	1.123216D+05	1.795428D-91	1.128646D+00	1.000000D+00
200	2.100000D+02	-1.198497D+05	6.282881D-92	8.237351D-01	1.000000D+00

Euler tests for delx = 1.05000000D+00

j	x	exponential		logistic	
		numerical	exact	numerical	exact
0	0.000000D+00	1.000000D+00	1.000000D+00	1.000000D-03	1.000000D-03
1	1.050000D+00	-5.000000D-02	3.499377D-01	2.048950D-03	2.852352D-03
2	2.100000D+00	2.500000D-03	1.224564D-01	4.195939D-03	8.108066D-03
3	3.150000D+00	-1.250000D-04	4.285213D-02	8.583190D-03	2.282622D-02
4	4.200000D+00	6.250000D-06	1.499558D-02	1.751818D-02	6.257595D-02
5	5.250000D+00	-3.125000D-07	5.247518D-03	3.559005D-02	1.601981D-01
6	6.300000D+00	1.562500D-08	1.836305D-03	7.162961D-02	3.527998D-01
7	7.350000D+00	-7.812500D-10	6.425924D-04	1.414534D-01	6.090320D-01
8	8.400000D+00	3.906250D-11	2.248673D-04	2.689699D-01	8.165649D-01
9	9.450000D+00	-1.953125D-12	7.868957D-05	4.754262D-01	9.271184D-01
10	1.050000D+01	9.765625D-14	2.753645D-05	7.372922D-01	9.732276D-01
11	1.155000D+01	-4.882813D-15	9.636043D-06	9.406692D-01	9.904654D-01
12	1.260000D+01	2.441406D-16	3.372015D-06	9.992704D-01	9.966427D-01
13	1.365000D+01	-1.220703D-17	1.179995D-06	1.000036D+00	9.988226D-01
14	1.470000D+01	6.103516D-19	4.129249D-07	9.999982D-01	9.995877D-01
15	1.575000D+01	-3.051758D-20	1.444980D-07	1.000000D+00	9.998557D-01
16	1.680000D+01	1.525879D-21	5.056531D-08	1.000000D+00	9.999495D-01
17	1.785000D+01	-7.629395D-23	1.769471D-08	1.000000D+00	9.999823D-01
18	1.890000D+01	3.814697D-24	6.192048D-09	1.000000D+00	9.999938D-01
19	1.995000D+01	-1.907349D-25	2.166831D-09	1.000000D+00	9.999978D-01
20	2.100000D+01	9.536743D-27	7.582560D-10	1.000000D+00	9.999992D-01
21	2.205000D+01	-4.768372D-28	2.653424D-10	1.000000D+00	9.999997D-01
22	2.310000D+01	2.384186D-29	9.285333D-11	1.000000D+00	9.999999D-01
23	2.415000D+01	-1.192093D-30	3.249288D-11	1.000000D+00	1.000000D+00
24	2.520000D+01	5.960464D-32	1.137049D-11	1.000000D+00	1.000000D+00
25	2.625000D+01	-2.980232D-33	3.978963D-12	1.000000D+00	1.000000D+00
26	2.730000D+01	1.490116D-34	1.392389D-12	1.000000D+00	1.000000D+00
27	2.835000D+01	-7.450581D-36	4.872495D-13	1.000000D+00	1.000000D+00
28	2.940000D+01	3.725290D-37	1.705070D-13	1.000000D+00	1.000000D+00
29	3.045000D+01	-1.862645D-38	5.966684D-14	1.000000D+00	1.000000D+00
30	3.150000D+01	9.313226D-40	2.087968D-14	1.000000D+00	1.000000D+00
31	3.255000D+01	-4.656613D-41	7.306588D-15	1.000000D+00	1.000000D+00
32	3.360000D+01	2.328306D-42	2.556851D-15	1.000000D+00	1.000000D+00
33	3.465000D+01	-1.164153D-43	8.947387D-16	1.000000D+00	1.000000D+00
34	3.570000D+01	5.820766D-45	3.131028D-16	1.000000D+00	1.000000D+00
35	3.675000D+01	-2.910383D-46	1.095665D-16	1.000000D+00	1.000000D+00
36	3.780000D+01	1.455192D-47	3.834145D-17	1.000000D+00	1.000000D+00
37	3.885000D+01	-7.275958D-49	1.341712D-17	1.000000D+00	1.000000D+00
38	3.990000D+01	3.637979D-50	4.695158D-18	1.000000D+00	1.000000D+00
39	4.095000D+01	-1.818989D-51	1.643013D-18	1.000000D+00	1.000000D+00
40	4.200000D+01	9.094947D-53	5.749522D-19	1.000000D+00	1.000000D+00
193	2.026500D+02	-7.965460-252	9.777397D-89	1.000000D+00	1.000000D+00
194	2.037000D+02	3.982730-253	3.421480D-89	1.000000D+00	1.000000D+00
195	2.047500D+02	-1.991365-254	1.197305D-89	1.000000D+00	1.000000D+00
196	2.058000D+02	9.956824-256	4.189822D-90	1.000000D+00	1.000000D+00
197	2.068500D+02	-4.978412-257	1.466177D-90	1.000000D+00	1.000000D+00
198	2.079000D+02	2.489206-258	5.130707D-91	1.000000D+00	1.000000D+00
199	2.089500D+02	-1.244603-259	1.795428D-91	1.000000D+00	1.000000D+00
200	2.100000D+02	6.223015-261	6.282881D-92	1.000000D+00	1.000000D+00

Table 3.
Diffusion Equation Program

Fortran Source Code - Arbitrary Time Centering

```

program diff
implicit double precision (a-h,o-z)
common /hh1/ T(259),X(259),F(259),XC(259),XA(259)
common /hh2/ SB(259),SG(259),AA(259),BB(259),CC(259),DD(259),
1          TT(259)

open (unit=12,file='diff.out',form='formatted')

c +++ *****
c +++ * * SET INPUT VARIABLES * *
c +++ * JB = NUMBER OF REAL CELLS
      JB=45
c +++ * DX = SPATIAL STEP SIZE
      DX=1.d+00
c +++ * X0 = LEFT BOUNDARY LOCATION
      X0=0.d+00
c +++ * NT = NUMBER OF CYCLES TO BE RUN
      NT=50
c +++ * NP = NUMBER OF CYCLES BETWEEN PRINTS
      NP=5
c +++ * DT = TIME STEP
      DT=1.d+01
c +++ * DM = MAXIMUM TIME STEP
      DM=1.D+20
c +++ * IC = 0, CARTESIAN; 1, CYLINDRICAL; 2, SPHERICAL COORDINATES
      IC=0
c +++ * TIME CENTERING PARAMETER (=0, EXPLICIT)
      PHI=0.5d+00
c +++ * * BC FLAGS: 1 = zero slope, 2 = 0 order fixed T, 3 = 1 order fixed T
      ML=1
      TL=300.d+00
      MR=2
      TR=1.0d+02
c +++ * TIME = ELAPSED PROBLEM TIME
      TIME=0.d+00

write (12,5000) JB,DX,X0
5000 format (' No. of real cells JB =',i5,' DX =',1p,d14.6,' X0 =',
1 d14.6)
write (12,5001) NT, DT, DM
5001 format (' No. of time steps NT =',i5,' DT =',1p,d14.6,
1 ' DT MAX =',d14.6)
write (12,5002) IC, PHI, NP, ML, TL, MR, TR
5002 format (' IC =',i6,' PHI =',1p,d14.6,' Print freq. NP =',i6/
1 ' ML =',i6,' TL =',d14.6/' MR =',i6,' TR =',d14.6)
c +++
c +++ *****

```

```

c +++ * * INITIALIZE SOME VARIABLES * *
c +++
      J1=JB+1
      J2=JB+2
      J3=JB+3
      L2=J2
      MP=0
      PI=3.14159265358979d+00

      do 6000 J = 1, J3
        X(J) = real(J)
        X(J) = DX * (X(J) - 2.d+00) + X0
        IF (J .gt. 1) XC(J-1) = 0.5d+00 * (X(J) + X(J-1))
6000 continue
c +++
c +++ * * SET AVERAGE CELL CENTERS AND INITIAL T * *
c +++
      do 6001 J = 2, J2
        XA(J) = 0.5d+00 * (XC(J) + XC(J-1))
        T(J) = TR
        IF (J .gt. 15) T(J) = TL
6001 continue
      XA(1)=X(1)
      XA(J3)=X(J3)

c +++ *****
c +++ * * SET BOUNDARY CONDITIONS ON T * *
      IF (ML .eq. 1) T(1)=T(2)
      IF (ML .eq. 2) T(1)=2.d+00*TL-T(2)
      IF (ML .eq. 3) T(1)=(8.d+00*TL-6.d+00*T(2)+T(3))/3.d+00
      IF (MR .eq. 1) T(J2)=T(J1)
      IF (MR .eq. 2) T(J2)=2.d+00*TR-T(J1)
      IF (MR .eq. 3) T(J2)=(8.d+00*TR-6.d+00*T(J1)+T(JB))/3.d+00

      N=0

c +++ *****
c +++ * * PRINTING ROUTINE * *
      write (12,5008) N, TIME
5008 format (/ ' TIME STEP N =',i5, ' TIME =',1p,d14.6//
1 5x,'N',10x,'T'/)

      do J=1, J2
        write (12,5007) J, T(J)
5007 format (1x,i5,1p,d16.8)
      enddo
      MP=MP+NP

c +++ *****
c +++ * * TIME STEP LOOP * *
      do 6006 N=1, NT
        TIME=TIME+DT

c +++ * * COMPUTE CELL-FACE DIFFUSIVITIES * *

```

```

do J=1, J3
  F(J)=1.d+00
enddo

do 6007 J=2, J1
  D1=XC(J)-XC(J-1)
  D2=XC(J+1)-XC(J)
  A1=1.d+00
  A2=1.d+00
  IF (IC .eq. 0) GO TO 900
  do MQ=1, IC
    A1=A1*XA(J)
    A2=A2*XA(J+1)
  enddo
900 D3=(A2*XA(J+1)-A1*XA(J))/real(IC+1)
  AA(J)=-F(J)*PHI*DT*A1/(D3*D1)
  CC(J)=-F(J+1)*PHI*DT*A2/(D3*D2)
  BB(J)=1.d+00-AA(J)-CC(J)
  DD(J)=F(J)*(T(J)-T(J-1))*A1/D1
  DD(J)=F(J+1)*(T(J+1)-T(J))*A2/D2-DD(J)
  DD(J)=T(J)+(1.d+00 - PHI)*DT*DD(J)/D3
c +++ * NOW SET BOUNDARY CONDITIONS
  IF (J .gt. 2) GO TO 1060
  IF (ML .eq. 1) BB(J)=BB(J)+AA(J)
  IF (ML .eq. 2) BB(J)=BB(J)-AA(J)
  IF (ML .eq. 2) DD(J)=DD(J)-2.d+00*TL*AA(J)
  IF (ML .eq. 3) BB(J)=BB(J)-2.d+00 *AA(J)
  IF (ML .eq. 3) CC(J)=CC(J)+AA(J)/3.d+00
  IF (ML .eq. 3) DD(J)=DD(J)-8.d+00*AA(J)*TL/3.d+00
  AA(J)=0.d+00
1060 IF (J .lt. J1) GO TO 1140
  IF (MR .eq. 1) BB(J)=BB(J)+CC(J)
  IF (MR .eq. 2) BB(J)=BB(J)-CC(J)
  IF (MR .eq. 2) DD(J)=DD(J)-2.d+00*TR*CC(J)
  IF (MR .eq. 3) BB(J)=BB(J)-2.d+00*CC(J)
  IF (MR .eq. 3) AA(J)=AA(J)+CC(J)/3.d+00
  IF (MR .eq. 3) DD(J)=DD(J)-8.d+00*CC(J)*TR/3.d+00
  CC(J)=0.d+00

1140 continue
6007 continue

c +++ * * SOLVE THE LINEAR SYSTEM * *

c +++ *****
c +++ * * TRIDIAGONAL SOLVER I = 2 TO L1 * * * * *
c +++ * * USES GAUSSIAN ELIMINATION WITHOUT PIVOTING
c +++ * * AA(I)*TT(I-1)+BB(I)*TT(I)+CC(I)*TT(I+1)=DD(I)
c +++ * * FICTITIOUS CELLS 1 AND L2 ARE NOT SOLVED
c +++ * * AA(2) AND CC(L1) ARE NOT USED
c +++ * * WE ASSUME L2 IS SET BY THE CALLING PROGRAM
  L1=L2-1
  L0=L2-2
  SW=BB(2)

```

```

SG(2)=DD(2)/SW
do J=3, L1
  SB(J-1)=CC(J-1)/SW
  SW=BB(J)-AA(J)*SB(J-1)
  SG(J)=(DD(J)-AA(J)*SG(J-1))/SW
enddo
TT(L1)=SG(L1)
do J=2, L0
  L4=L2-J
  TT(L4)=SG(L4)-SB(L4)*TT(L4+1)
enddo
c +++ end of linear system solver

do J=2, J1
  T(J)=TT(J)
enddo

c +++ *****
c +++ * * SET BOUNDARY CONDITIONS ON T * *
  IF (ML .eq. 1) T(1)=T(2)
  IF (ML .eq. 2) T(1)=2.d+00*TL-T(2)
  IF (ML .eq. 3) T(1)=(8.d+00*TL-6.d+00*T(2)+T(3))/3.d+00
  IF (MR .eq. 1) T(J2)=T(J1)
  IF (MR .eq. 2) T(J2)=2.d+00*TR-T(J1)
  IF (MR .eq. 3) T(J2)=(8.d+00*TR-6.d+00*T(J1)+T(JB))/3.d+00

c +++ * * CALL THE PRINT ROUTINE * *
  IF (N .eq. MP) THEN

    write (12,5008) N, TIME

    do J=1, J2
      write (12,5007) J, T(J)
    enddo
    MP=MP+NP

  endif

c +++ * * DT CONTROL * *
  IF (DT .gt. DM) DT=DM
6006 continue

  close (unit=12)
  stop
  end

```

Table 4.
Diffusion Equation Solutions

J	T(CN,10.0)	T(IM,10.0)	T(CN,0.5)	T(IM,0.5)
No. of real cells JB = 45 DX = 1.000000D+00 X0 = 0.000000D+00 No. of time steps NT = 50 DT = 1.000000D+01 DT MAX = 1.000000D+20 IC = 0 PHI = 5.000000D-01 Print freq. NP = 5 ML = 1 TL = 3.000000D+02 MR = 2 TR = 1.000000D+02				
TIME STEP N = 0 TIME = 0.000000D+00				
1	1.00000000D+02			
2	1.00000000D+02			
3	1.00000000D+02			
4	1.00000000D+02			
5	1.00000000D+02			
6	1.00000000D+02			
7	1.00000000D+02			
8	1.00000000D+02			
9	1.00000000D+02			
10	1.00000000D+02			
11	1.00000000D+02			
12	1.00000000D+02			
13	1.00000000D+02			
14	1.00000000D+02			
15	1.00000000D+02			
16	3.00000000D+02			
17	3.00000000D+02			
18	3.00000000D+02			
19	3.00000000D+02			
20	3.00000000D+02			
21	3.00000000D+02			
22	3.00000000D+02			
23	3.00000000D+02			
24	3.00000000D+02			
25	3.00000000D+02			
26	3.00000000D+02			
27	3.00000000D+02			
28	3.00000000D+02			
29	3.00000000D+02			
30	3.00000000D+02			
31	3.00000000D+02			
32	3.00000000D+02			
33	3.00000000D+02			
34	3.00000000D+02			
35	3.00000000D+02			
36	3.00000000D+02			
37	3.00000000D+02			

38 3.00000000D+02
 39 3.00000000D+02
 40 3.00000000D+02
 41 3.00000000D+02
 42 3.00000000D+02
 43 3.00000000D+02
 44 3.00000000D+02
 45 3.00000000D+02
 46 3.00000000D+02
 47 -1.00000000D+02

TIME STEP N = 5 TIME = 5.000000D+01

N	T(CN,10.0)	T(IM,10.0)	T(CN,0.5)	T(IM,0.5)
1	1.32320953D+02	1.30612804D+02	1.32333107D+02	1.32225268D+02
2	1.32320953D+02	1.30612804D+02	1.32333107D+02	1.32225268D+02
3	1.33136611D+02	1.31394133D+02	1.33169235D+02	1.33060247D+02
4	1.34770531D+02	1.32956174D+02	1.34832741D+02	1.34721750D+02
5	1.37230200D+02	1.35296925D+02	1.37306048D+02	1.37192753D+02
6	1.40528685D+02	1.38411510D+02	1.40562611D+02	1.40447455D+02
7	1.44675363D+02	1.42290044D+02	1.44566765D+02	1.44451033D+02
8	1.49643455D+02	1.46914990D+02	1.49273610D+02	1.49159424D+02
9	1.55300327D+02	1.52258198D+02	1.54629003D+02	1.54519204D+02
10	1.61302961D+02	1.58277923D+02	1.60569715D+02	1.60467650D+02
11	1.67029336D+02	1.64916145D+02	1.67023808D+02	1.66933043D+02
12	1.71818315D+02	1.72096636D+02	1.73911275D+02	1.73835256D+02
13	1.76306900D+02	1.79724229D+02	1.81144948D+02	1.81086656D+02
14	1.86870001D+02	1.87685731D+02	1.88631674D+02	1.88593309D+02
15	2.27880090D+02	1.95852779D+02	1.96273717D+02	1.96256445D+02
16	1.72316407D+02	2.04086684D+02	2.03970325D+02	2.03974122D+02
17	2.13334383D+02	2.12244851D+02	2.11619384D+02	2.11642994D+02
18	2.23912588D+02	2.20187844D+02	2.19119067D+02	2.19160089D+02
19	2.28422004D+02	2.27785763D+02	2.26369383D+02	2.26424471D+02
20	2.33234917D+02	2.34923000D+02	2.33273535D+02	2.33338711D+02
21	2.38984098D+02	2.41500963D+02	2.39739022D+02	2.39810057D+02
22	2.45001928D+02	2.47438811D+02	2.45678434D+02	2.45751274D+02
23	2.50656858D+02	2.52672479D+02	2.51009913D+02	2.51081100D+02
24	2.55592391D+02	2.57152460D+02	2.55657310D+02	2.55724364D+02
25	2.59657537D+02	2.60840780D+02	2.59550074D+02	2.59611786D+02
26	2.62801506D+02	2.63707592D+02	2.62622958D+02	2.62679570D+02
27	2.65003621D+02	2.65727756D+02	2.64815623D+02	2.64868868D+02
28	2.66241222D+02	2.66877674D+02	2.66072285D+02	2.66125276D+02
29	2.66481906D+02	2.67132586D+02	2.66341484D+02	2.66398457D+02
30	2.65687233D+02	2.66464506D+02	2.65576109D+02	2.65642033D+02
31	2.63819515D+02	2.64840894D+02	2.63733754D+02	2.63813840D+02
32	2.60847050D+02	2.62224167D+02	2.60777457D+02	2.60876604D+02
33	2.56745441D+02	2.58572110D+02	2.56676851D+02	2.56799075D+02
34	2.51494021D+02	2.53839256D+02	2.51409679D+02	2.51557585D+02
35	2.45067719D+02	2.47979263D+02	2.44963635D+02	2.45137968D+02
36	2.37427320D+02	2.40948294D+02	2.37338378D+02	2.37537720D+02
37	2.28516065D+02	2.32709370D+02	2.28547611D+02	2.28768238D+02
38	2.18278572D+02	2.23237583D+02	2.18621026D+02	2.18856969D+02
39	2.06727434D+02	2.12525960D+02	2.07605957D+02	2.07849257D+02

40	1.94084734D+02	2.00591651D+02	1.95568549D+02	1.95809704D+02
41	1.80993089D+02	1.87481942D+02	1.82594299D+02	1.82822868D+02
42	1.68654406D+02	1.73279441D+02	1.68787847D+02	1.68993168D+02
43	1.58351054D+02	1.58105643D+02	1.54271954D+02	1.54443911D+02
44	1.48771735D+02	1.42121962D+02	1.39185632D+02	1.39315410D+02
45	1.27131819D+02	1.25527400D+02	1.23681491D+02	1.23762253D+02
46	4.46534951D+01	1.08552247D+02	1.07922399D+02	1.07949814D+02
47	1.55346505D+02	9.14477533D+01	9.20776012D+01	9.20501863D+01

TIME STEP N = 10 TIME = 1.000000D+02

N	T(CN,10.0)	T(IM,10.0)	T(CN,0.5)	T(IM,0.5)
1	1.63893018D+02	1.60961097D+02	1.63842243D+02	1.63691870D+02
2	1.63893018D+02	1.60961097D+02	1.63842243D+02	1.63691870D+02
3	1.64339489D+02	1.61440635D+02	1.64293502D+02	1.64144663D+02
4	1.65229870D+02	1.62393579D+02	1.65190000D+02	1.65044209D+02
5	1.66559423D+02	1.63807706D+02	1.66519752D+02	1.66378496D+02
6	1.68319447D+02	1.65664783D+02	1.68264937D+02	1.68129652D+02
7	1.70488770D+02	1.67940693D+02	1.70402078D+02	1.70274138D+02
8	1.73015370D+02	1.70605619D+02	1.72902275D+02	1.72782980D+02
9	1.75797444D+02	1.73624283D+02	1.75731509D+02	1.75622068D+02
10	1.78708338D+02	1.76956251D+02	1.78850980D+02	1.78752503D+02
11	1.81766055D+02	1.80556309D+02	1.82217504D+02	1.82130992D+02
12	1.85531553D+02	1.84374908D+02	1.85783951D+02	1.85710289D+02
13	1.91273950D+02	1.88358686D+02	1.89499726D+02	1.89439676D+02
14	1.97739037D+02	1.92451051D+02	1.93311280D+02	1.93265480D+02
15	1.81493509D+02	1.96592819D+02	1.97162657D+02	1.97131619D+02
16	2.16687883D+02	2.00722899D+02	2.00996062D+02	2.00980174D+02
17	2.00347495D+02	2.04779008D+02	2.04752455D+02	2.04751980D+02
18	2.06620925D+02	2.08698390D+02	2.08372158D+02	2.08387239D+02
19	2.12071035D+02	2.12418534D+02	2.11795467D+02	2.11826130D+02
20	2.15437950D+02	2.15877872D+02	2.14963284D+02	2.15009441D+02
21	2.17983396D+02	2.19016430D+02	2.17817736D+02	2.17879185D+02
22	2.20259350D+02	2.21776448D+02	2.20302797D+02	2.20379224D+02
23	2.22273396D+02	2.24102933D+02	2.22364906D+02	2.22455881D+02
24	2.23887180D+02	2.25944171D+02	2.23953563D+02	2.24058538D+02
25	2.24986056D+02	2.27252187D+02	2.25021919D+02	2.25140217D+02
26	2.25504128D+02	2.27983159D+02	2.25527333D+02	2.25658142D+02
27	2.25405415D+02	2.28097804D+02	2.25431910D+02	2.25574273D+02
28	2.24665653D+02	2.27561745D+02	2.24703003D+02	2.24855804D+02
29	2.23264206D+02	2.26345859D+02	2.23313673D+02	2.23475631D+02
30	2.21183212D+02	2.24426633D+02	2.21243112D+02	2.21412771D+02
31	2.18409538D+02	2.21786516D+02	2.18477006D+02	2.18652730D+02
32	2.14936727D+02	2.18414276D+02	2.15007844D+02	2.15187818D+02
33	2.10765513D+02	2.14305362D+02	2.10835158D+02	2.11017394D+02
34	2.05902377D+02	2.09462256D+02	2.05965690D+02	2.06148046D+02
35	2.00356424D+02	2.03894793D+02	2.00413482D+02	2.00593680D+02
36	1.94136626D+02	1.97620451D+02	1.94199878D+02	1.94375538D+02
37	1.87254704D+02	1.90664562D+02	1.87353433D+02	1.87522111D+02
38	1.79742283D+02	1.83060434D+02	1.79909730D+02	1.80068966D+02
39	1.71688090D+02	1.74849355D+02	1.71911096D+02	1.72058467D+02
40	1.63276311D+02	1.66080446D+02	1.63406225D+02	1.63539402D+02
41	1.54737116D+02	1.56810360D+02	1.54449697D+02	1.54566506D+02

42	1.46007917D+02	1.47102786D+02	1.45101413D+02	1.45199892D+02
43	1.35937005D+02	1.37027780D+02	1.35425939D+02	1.35504396D+02
44	1.21953324D+02	1.26660908D+02	1.25491770D+02	1.25548830D+02
45	1.06529646D+02	1.16082223D+02	1.15370535D+02	1.15405185D+02
46	1.36494318D+02	1.05375089D+02	1.05136137D+02	1.05147757D+02
47	6.35056823D+01	9.46249109D+01	9.48638634D+01	9.48522434D+01

TIME STEP N = 50 TIME = 5.000000D+02

N	T(CN,10.0)	T(IM,10.0)	T(CN,0.5)	T(IM,0.5)
1	1.72755621D+02	1.72795084D+02	1.72752317D+02	1.72755292D+02
2	1.72755621D+02	1.72795084D+02	1.72752317D+02	1.72755292D+02
3	1.72673827D+02	1.72715496D+02	1.72670560D+02	1.72673635D+02
4	1.72510259D+02	1.72556316D+02	1.72507069D+02	1.72510344D+02
5	1.72264949D+02	1.72317541D+02	1.72261895D+02	1.72265465D+02
6	1.71937966D+02	1.71999167D+02	1.71935113D+02	1.71939070D+02
7	1.71529519D+02	1.71601194D+02	1.71526827D+02	1.71531261D+02
8	1.71039891D+02	1.71123629D+02	1.71037176D+02	1.71042166D+02
9	1.70468619D+02	1.70566494D+02	1.70466331D+02	1.70471953D+02
10	1.69814657D+02	1.69929829D+02	1.69814505D+02	1.69820826D+02
11	1.69084261D+02	1.69213701D+02	1.69081957D+02	1.69089034D+02
12	1.68284966D+02	1.68418212D+02	1.68268995D+02	1.68276876D+02
13	1.67321730D+02	1.67543503D+02	1.67375986D+02	1.67384709D+02
14	1.66516158D+02	1.66589770D+02	1.66403359D+02	1.66412951D+02
15	1.65196056D+02	1.65557268D+02	1.65351613D+02	1.65362088D+02
16	1.64376714D+02	1.64446320D+02	1.64221325D+02	1.64232686D+02
17	1.62900178D+02	1.63257332D+02	1.63013152D+02	1.63025393D+02
18	1.61781914D+02	1.61990799D+02	1.61727848D+02	1.61740947D+02
19	1.60350076D+02	1.60647314D+02	1.60366260D+02	1.60380185D+02
20	1.58926795D+02	1.59227582D+02	1.58929342D+02	1.58944049D+02
21	1.57417728D+02	1.57732425D+02	1.57418159D+02	1.57433593D+02
22	1.55831283D+02	1.56162793D+02	1.55833893D+02	1.55849988D+02
23	1.54174764D+02	1.54519771D+02	1.54177850D+02	1.54194530D+02
24	1.52448347D+02	1.52804591D+02	1.52451463D+02	1.52468643D+02
25	1.50652966D+02	1.51018632D+02	1.50656301D+02	1.50673886D+02
26	1.48790470D+02	1.49163431D+02	1.48794068D+02	1.48811955D+02
27	1.46862810D+02	1.47240689D+02	1.46866608D+02	1.46884689D+02
28	1.44871968D+02	1.45252270D+02	1.44875910D+02	1.44894071D+02
29	1.42820057D+02	1.43200209D+02	1.42824105D+02	1.42842226D+02
30	1.40709354D+02	1.41086711D+02	1.40713471D+02	1.40731430D+02
31	1.38542286D+02	1.38914153D+02	1.38546429D+02	1.38564101D+02
32	1.36321418D+02	1.36685081D+02	1.36325543D+02	1.36342804D+02
33	1.34049455D+02	1.34402212D+02	1.34053520D+02	1.34070244D+02
34	1.31729232D+02	1.32068425D+02	1.31733203D+02	1.31749267D+02
35	1.29363731D+02	1.29686764D+02	1.29367569D+02	1.29382854D+02
36	1.26956092D+02	1.27260423D+02	1.26959723D+02	1.26974112D+02
37	1.24509559D+02	1.24792748D+02	1.24512892D+02	1.24526276D+02
38	1.22027265D+02	1.22287222D+02	1.22030417D+02	1.22042692D+02
39	1.19512371D+02	1.19747457D+02	1.19515747D+02	1.19526817D+02
40	1.16969693D+02	1.17177188D+02	1.16972428D+02	1.16982207D+02
41	1.14405387D+02	1.14580254D+02	1.14404095D+02	1.14412505D+02
42	1.11811175D+02	1.11960591D+02	1.11814460D+02	1.11821435D+02
43	1.09176385D+02	1.09322218D+02	1.09207303D+02	1.09212787D+02

44	1.06695676D+02	1.06669219D+02	1.06586459D+02	1.06590408D+02
45	1.03730576D+02	1.04005737D+02	1.03955807D+02	1.03958190D+02
46	1.01630387D+02	1.01335947D+02	1.01319258D+02	1.01320055D+02
47	9.83696131D+01	9.86640529D+01	9.86807415D+01	9.86799454D+01

Table 5.
Growth Factors for the Adams-Bashforth Wave Equation

β	r	rr^*
7.8125E-04	(9.99999702E-01, -7.81250128E-04) (2.98023224E-07, -3.90624919E-04)	1.000000E+00 1.525879E-07
1.5625E-03	(9.99998808E-01, -1.56250107E-03) (1.22189522E-06, -7.81249080E-04)	1.000000E+00 6.103516E-07
3.1250E-03	(9.99995112E-01, -3.12500773E-03) (4.88758087E-06, -1.56249246E-03)	1.000000E+00 2.441407E-06
6.2500E-03	(9.99980450E-01, -6.25006109E-03) (1.95503235E-05, -3.12493904E-03)	9.999999E-01 9.765627E-06
1.2500E-02	(9.99921918E-01, -1.25004891E-02) (7.81118870E-05, -6.24951208E-03)	1.000000E+00 3.906250E-05
2.5000E-02	(9.99687433E-01, -2.50039101E-02) (3.12536955E-04, -1.24960914E-02)	1.000000E+00 1.562500E-04
5.0000E-02	(9.98749256E-01, -5.00313491E-02) (1.25077367E-03, -2.49686539E-02)	1.000003E+00 6.249981E-04
1.0000E-01	(9.94987607E-01, -1.00253165E-01) (5.01239300E-03, -4.97468449E-02)	1.000051E+00 2.499873E-03
2.0000E-01	(9.79807079E-01, -2.02104285E-01) (2.01929212E-02, -9.78957266E-02)	1.000868E+00 9.991327E-03
4.0000E-01	(9.17540193E-01, -4.19748962E-01) (8.24598074E-02, -1.80251062E-01)	1.018069E+00 3.929007E-02
8.0000E-01	(7.43160725E-01, -1.01125073E+00) (2.56839305E-01, -1.88749343E-01)	1.574916E+00 1.015927E-01
1.6000E+00	(6.80870414E-01, -2.30576420E+00) (3.19129586E-01, -9.42360163E-02)	5.780133E+00 1.107241E-01

```

program roots
complex rootp, rootm, arg, one, half

```

```

one = ( 1., 0. )
half = ( 0.5, 0. )
beta = 0.1 / ( 2.**7)

```

```

do i = 1, 12
c +++ compute argument of the square root
arg = one - cmplx(2.25 * beta * beta, beta)
arg = sqrt(arg)

```

```
c +++ compute complex roots
  rootp = (cplx(1., -1.5 * beta) + arg) * half
  rootm = (cplx(1., -1.5 * beta) - arg) * half
  write (*,10) beta, rootp, rootm
10 format (' beta =',1p,e14.6/' rootp =',2e16.8/' rootm =',
1 2e16.8)

c +++ magnitudes
  rpstar = rootp * conjg( rootp )
  rmstar = rootm * conjg( rootm )
  write (*,20) rpstar, rmstar
20 format (' magnitudes =',1p,2e14.6)

  beta = beta * 2.
  enddo

  stop
  end
```

Table 6.
Wave Equation Program

Fortran code for first order wave equation

```

program advect
implicit double precision (a-h,o-z)
common /hh1/ T(259),X(259),F(259)
common /hh2/ SB(259),SG(259),AA(259),BB(259),CC(259),DD(259),
1          TT(259)

open (unit=12,file='advect.out',form='formatted')

c +++ *****
c +++ * * SET INPUT VARIABLES * *
c +++ * JB = NUMBER OF REAL CELLS

      JB=12
c +++ * DX = SPATIAL STEP SIZE
      DX=1.d+00
c +++ * X0 = LEFT BOUNDARY LOCATION
      X0=0.d+00
c +++ * NT = NUMBER OF CYCLES TO BE RUN
      NT=25
c +++ * NP = NUMBER OF CYCLES BETWEEN PRINTS
      NP= 1
c +++ * DT = TIME STEP
      DT=0.1d+00
c +++ * TIME = ELAPSED PROBLEM TIME
      TIME=0.d+00

      write (12,5000) JB,DX,X0
5000 format (' advect output'// ' No. of real cells JB =',i5,
1 ' DX =',1p,d14.6,' X0 =', d14.6)
      write (12,5001) NT, DT
5001 format (' No. of time steps NT =',i7,' DT =',1p,d14.6)
      write (12,5002) NP
5002 format (' Print freq. NP =',i6/ )
c +++
c +++ *****
c +++ * * INITIALIZE SOME VARIABLES * *
c +++
      J1=JB+1
      J2=JB+2
      J3=JB+3
      L2=J2
      MP=0

      do 6000 J = 1, J3
          X(J) = real(J)
          X(J) = DX * (X(J) - 2.d+00) + X0
6000 continue

```

```

c +++
c +++ * * SET AVERAGE CELL CENTERS AND INITIAL T * *
c +++
      do 6001 J = 2, J2
        T(J) = 0.d+00
        IF (mod(J,4) .eq. 3) T(J)=1.d+00
        IF (mod(J,4) .eq. 1) T(J)= -1.d+00
        TT(J)=T(J)
6001 continue

c +++ *****
c +++ * * SET BOUNDARY CONDITIONS ON T * *
      T(1) = T(J1)
      T(J2) = T(2)
      TT(1) = T(1)
      TT(J2)= T(J2)

      N=0

c +++ *****
c +++ * * PRINTING ROUTINE * *
      write (12,5008) N, TIME
5008 format (/ ' TIME STEP N =',i7, ' TIME =',1p,d14.6//
      1 5x,'N',10x,'T'/)

      do J=1, J2
        write (12,5007) J, T(J)
5007 format (1x,i5,1p,d16.8)
      enddo
      MP=MP+NP

c +++ *****
c +++ * * TIME STEP LOOP * *
      do 6006 N=1, NT
        TIME=TIME+DT

        do j = 1, J2
          AA(j) = TT(j)
          TT(j) = T(j)
        enddo

        do 6007 J=2, J1
c all velocities = 1.0
c donor cell
          T(J)=TT(J)-(DT/DX)*(TT(J)-TT(J-1))
c Euler centered
          T(J)=TT(J)-(DT/(2.d+00*DX))*(TT(J+1)-TT(J-1))
c Adams-Bashforth
          T(J)=TT(J)-(DT/(4.d+00*DX))*(3.d+00*TT(J+1)-3.d+00*TT(J-1)
            1 -AA(J+1)+AA(J-1))
6007 continue

c +++ *****
c +++ * * SET BOUNDARY CONDITIONS ON T * *

```



```
T(1) = T(J1)
T(J2) = T(2)

c +++ * * CALL THE PRINT ROUTINE * *
  IF (N .eq. MP) THEN

    write (12,5008) N, TIME

    do J=1, J2
      write (12,5007) J, T(J)
    enddo
    MP=MP+NP

  endif

6006 continue

close (unit=12)
stop
end
```

Table 7.
Wave Equation Examples

Numerical examples for Euler's method and Adams-Bashforth

advect output - Euler's method

No. of real cells JB = 12 DX = 1.000000D+00 XO = 0.000000D+00
 No. of time steps NT = 50000 DT = 1.000000D-01
 Print freq. NP = 2000

TIME STEP N = 0 TIME = 0.000000D+00

N	T
1	-1.00000000D+00
2	0.00000000D+00
3	1.00000000D+00
4	0.00000000D+00
5	-1.00000000D+00
6	0.00000000D+00
7	1.00000000D+00
8	0.00000000D+00
9	-1.00000000D+00
10	0.00000000D+00
11	1.00000000D+00
12	0.00000000D+00
13	-1.00000000D+00
14	0.00000000D+00

TIME STEP N = 2000 TIME = 2.000000D+02

N	T
1	3.21141545D+03
2	2.07116638D+04
3	-3.21141545D+03
4	-2.07116638D+04
5	3.21141545D+03
6	2.07116638D+04
7	-3.21141545D+03
8	-2.07116638D+04
9	3.21141545D+03
10	2.07116638D+04
11	-3.21141545D+03
12	-2.07116638D+04
13	3.21141545D+03
14	2.07116638D+04

TIME STEP N = 4000 TIME = 4.000000D+02

N	T
1	4.18659827D+08
2	-1.33027514D+08
3	-4.18659827D+08
4	1.33027514D+08
5	4.18659827D+08

6 -1.33027514D+08
7 -4.18659827D+08
8 1.33027514D+08
9 4.18659827D+08
10 -1.33027514D+08
11 -4.18659827D+08
12 1.33027514D+08
13 4.18659827D+08
14 -1.33027514D+08

TIME STEP N = 26000 TIME = 2.600000D+03

N	T
1	1.36965341D+56
2	-6.26452543D+55
3	-1.36965341D+56
4	6.26452543D+55
5	1.36965341D+56
6	-6.26452543D+55
7	-1.36965341D+56
8	6.26452543D+55
9	1.36965341D+56
10	-6.26452543D+55
11	-1.36965341D+56
12	6.26452543D+55
13	1.36965341D+56
14	-6.26452543D+55

advect output - Adams-Bashforth

No. of real cells JB = 12 DX = 1.000000D+00 XO = 0.000000D+00
No. of time steps NT = 10000 DT = 1.000000D-01
Print freq. NP = 2000

TIME STEP N = 0 TIME = 0.000000D+00

N	T
1	-1.00000000D+00
2	0.00000000D+00
3	1.00000000D+00
4	0.00000000D+00
5	-1.00000000D+00
6	0.00000000D+00
7	1.00000000D+00
8	0.00000000D+00
9	-1.00000000D+00
10	0.00000000D+00
11	1.00000000D+00
12	0.00000000D+00
13	-1.00000000D+00
14	0.00000000D+00

TIME STEP N = 2000 TIME = 2.000000D+02

N	T
1	-1.03124710D+00

2 2.34795194D-01
3 1.03124710D+00
4 -2.34795194D-01
5 -1.03124710D+00
6 2.34795194D-01
7 1.03124710D+00
8 -2.34795194D-01
9 -1.03124710D+00
10 2.34795194D-01
11 1.03124710D+00
12 -2.34795194D-01
13 -1.03124710D+00
14 2.34795194D-01

TIME STEP N = 4000 TIME = 4.000000D+02

N	T
1	-1.00353378D+00
2	4.81324783D-01
3	1.00353378D+00
4	-4.81324783D-01
5	-1.00353378D+00
6	4.81324783D-01
7	1.00353378D+00
8	-4.81324783D-01
9	-1.00353378D+00
10	4.81324783D-01
11	1.00353378D+00
12	-4.81324783D-01
13	-1.00353378D+00
14	4.81324783D-01

TIME STEP N = 26000 TIME = 2.600000D+03

N	T
1	1.89599314D+00
2	4.58908413D-01
3	-1.89599314D+00
4	-4.58908413D-01
5	1.89599314D+00
6	4.58908413D-01
7	-1.89599314D+00
8	-4.58908413D-01
9	1.89599314D+00
10	4.58908413D-01
11	-1.89599314D+00
12	-4.58908413D-01
13	1.89599314D+00
14	4.58908413D-01

TIME STEP N = 50000 TIME = 5.000000D+03

N	T
1	-2.75465441D+00
2	-2.31462859D+00
3	2.75465441D+00
4	2.31462859D+00

5 -2.75465441D+00
6 -2.31462859D+00
7 2.75465441D+00
8 2.31462859D+00
9 -2.75465441D+00
10 -2.31462859D+00
11 2.75465441D+00
12 2.31462859D+00
13 -2.75465441D+00
14 -2.31462859D+00

TIME STEP N = 100000 TIME = 1.000000D+04

N	T
1	-2.21296301D+00
2	-1.26892226D+01
3	2.21296301D+00
4	1.26892226D+01
5	-2.21296301D+00
6	-1.26892226D+01
7	2.21296301D+00
8	1.26892226D+01
9	-2.21296301D+00
10	-1.26892226D+01
11	2.21296301D+00
12	1.26892226D+01
13	-2.21296301D+00
14	-1.26892226D+01
