## 1 SUMMARY

To solve the two point boundary value problem for the second order non-linear differential equation

$$
y^{\prime \prime}+f\left(x, y, y^{\prime}\right) y^{\prime}+g\left(x, y, y^{\prime}\right) y=r\left(x, y, y^{\prime}\right) \quad x_{1} \leq x \leq x_{n}
$$

given boundary conditions of the form $a y^{\prime}+b y=c$ at the two points $x_{1}$ and $x_{n}$.
The number of points to be used over the range of $x$ must be specified and the user must choose this number so that the step-length is sufficiently small for the accuracy requirements to be achieved. The subroutine always indicates on return whether these requirements have been met. The user must also provide a subroutine to calculate values of the functions $f\left(x, y, y^{\prime}\right), g\left(x, y, y^{\prime}\right)$ and $r\left(x, y, y^{\prime}\right)$ over the range $x_{1} \leq x \leq x_{n}$.

An iterative scheme is used, based on a finite difference approximation to a linearized form of the equation. DD11A/AD is used to solve the linear form and corrections based on 3rd and 4th differences are applied to the solution approximations. The user has some control over this refinement of the solution.
ATTRIBUTES - Version: 1.0.0. Types: DD12A, DD12AD. Calls: DD11, TA03, TD01 and FUNCTS (a user subroutine). Original date: May 1965. Origin: P.Hallowell, Atlas Lab., Chilton, Oxon.

## 2 HOW TO USE THE PACKAGE

### 2.1 The argument list and calling sequence

## The single precision version:

CALL DD12A (Y, X1, XN, N, Y1, YN, E, LIM, EPS, NLIN, K, W)

## The double precision version:

CALL DD12AD (Y, X1, XN, N, Y1, YN, E, LIM, EPS, NLIN, K, W)
$Y$ is a REAL (DOUBLE PRECISION in the $D$ version) array of length at least $n$ which must be set by the user to an initial approximation to the solution $y\left(x_{i}\right)$ at the points $x_{i} i=1,2, \ldots, n$. On return from the subroutine these values will have been replaced by the final approximations to the solution.
$\mathrm{X1}, \mathrm{XN}$ are REAL (DOUBLE PRECISION in the D version) variables and must be set by the user to $x_{1}$ and $x_{n}$ the lower and upper limits of the range of $x$ over which the equation is to be integrated.
$\mathrm{N} \quad$ is an INTEGER variable and is set by the user to $n$ the number of equally spaced points $x_{i} i=1,2, \ldots, n$ to be used to cover the range of integration. It is the responsibility of the user to choose $n$ so that the step-length $\frac{x_{n}-x_{1}}{n-1}$ is small enough to allow both the finite difference approximations and the linearizations generated by the subroutine to converge and give the required accuracy. Restriction: $n \geq 3$.
Y1, YN are two REAL (DOUBLE PRECISION in the D version) arrays each of length three words in which the user must specify the boundary conditions at $x_{1}$ and $x_{n}$ respectively, i.e. if these are

$$
a_{1} y_{1}^{\prime}+b_{1} y_{1}=c_{1}
$$

$y_{1}^{\prime}$ and $y_{1}$ evaluated at $x_{1}$, and

$$
a_{n} y_{n}^{\prime}+b_{n} y_{n}=c_{n}
$$

$y_{n}^{\prime}$ and $y_{n}$ evaluated at $x_{n}$, then the user must set $\mathrm{Y} 1(1)=a_{1}, \mathrm{Y} 1(2)=b_{1}, \mathrm{Y} 1(3)=c_{1}$ and similarly for YN and $a_{n}, b_{n}$ and $c_{n}$.
E is a REAL (DOUBLE PRECISION in the D version) array of length at least $n$ which on return will have been set by
the subroutine to the differences between the final linearization values of the solution ignoring 3rd and 4th differences, i.e. gives the effect of the 3rd and 4th difference corrections on the final values. On a convergence failure E may contain other values, see NLIN

LIM is an INTEGER variable and must be set by the user to a non zero integer value, the absolute value of which should specify a limit on the number of times the equation may be linearized and is made negative if no 3rd or 4th difference corrections are to be applied. Note: LIM is always returned set to the absolute value of its value on entry.

EPS is a REAL (DOUBLE PRECISION in the D version) variable and is set by the user to the absolute accuracy required in the solution. Let $\eta_{i} i=1,2, \ldots, n$ be the differences between the current approximations to the solution and those of the previous linearization, then the accuracy criterion is such that the current values are accepted if $\left|\eta_{i}\right| \leq \varepsilon i=1,2, \ldots, n$ where $\varepsilon \geq 0$ is to be specified in EPS. If correction terms are requested they are applied only after this condition is satisfied and the test is repeated again after the corrections have been applied.

NLIN is an INTEGER variable and is set by the subroutine to the number of linearizations that were required to achieve the final values of the solution. It serves also as a failure flag and is set to -1 if the approximations have failed to converge within the limits imposed. This may be because the limit LIM on the number of linearizations has been reached in which case $E$ will be returned set to zero, unless the stage has been reached where the correction terms are being applied, where E is set to the difference between the final values with correction terms and without the correction terms.

K is an INTEGER variable set by the user to indicate whether the functions $f, g$ and $r$ require values of $y^{\prime}$ at $x_{i}$ $i=1,2, \ldots, n$ in which case the values are passed by the subroutine to the subroutine FUNCTS provided by the user. If these values are required set $\mathrm{K}>0$ but if they are not required set $\mathrm{K}=0$, see $\S$ 2.2.

W is a REAL (DOUBLE PRECISION in the D version) array of length at least $11 n$ words used by the routine as workspace.

### 2.2 The subroutine to evaluate $f, g$ and $r$

The user must provide a subroutine to evaluate the functions $f\left(x, y, y^{\prime}\right) g\left(x, y, y^{\prime}\right)$ and $r\left(x, y, y^{\prime}\right)$ at any point $x_{i}$ $i=1,2, \ldots, n$ equally spaced in $x_{1} \leq x \leq x_{n}$. The subroutine will be called by DD12A/AD $n$ times per linearization. Both single and double precision versions require the subroutine to be called FUNCTS and it must be specified as
The single precision version:
SUBROUTINE FUNCTS (F, G, R, X, Y, I, W)

## The double precision version:

```
SUBROUTINE FUNCTS(F,G,R,X,Y,I,W)
```

F is a REAL (DOUBLE PRECISION in the D version) variable which must be set to the value of $f\left(x, y, y^{\prime}\right)$.
G is a REAL (DOUBLE PRECISION in the D version) variable which must be set to the value of $g\left(x, y, y^{\prime}\right)$.
$\mathrm{R} \quad$ is a REAL (DOUBLE PRECISION in the D version) variable which must be set to the value of $r(x, y, y)$.
$\mathrm{X} \quad$ is a REAL (DOUBLE PRECISION in the D version) variable which will contain the value of $x$ and must not be altered.

Y is a REAL (DOUBLE PRECISION in the D version) variable which will contain the value of $y$ and must not be altered.

I is an INTEGER variable which, if $\mathrm{K}>0$ in main call, points to the value of the latest approximation to $y^{\prime}$ which is passed to FUNCTS in the array element $W(I)$.

W is the REAL (DOUBLE PRECISION in the D version) workspace array passed as the last argument to DD12A/AD.

## 3 GENERAL INFORMATION

Use of Common: None.
Workspace: $11 n$ words passed in argument $W$..
Other subroutines: the library subroutines DD11A/AD, MA35A/AD, TA03A/AD and TD01A/AD are called as well as a subroutine FUNCTS which must be provided by the user.

Input/Output: diagnostics concerned with the specification of LIM and $K$, a RETURN is also made when these occur.
System dependence: none.

## Restrictions:

$n \geq 3$,
$\mathrm{LIM} \neq 0$,
EPS $>0$,
$K \geq 0$.

## 4 METHOD

Starting from an initial approximation to the solution the equation is linearised and then solved by calling DD11A/AD. It is then repeatedly re-linearised until the requested accuracy is reached. At this stage the approximation to the solution has been obtained by ignoring the effect of 3rd and 4th differences, see DD11A/AD, so the solution is then corrected to take these into account.

