Author(s) : C. Iselin
Library: MATHLIB
Submitter :
Submitted: 04.09.1972
Language : Fortran
Revised: 15.01.1977

Real Fast Fourier Transform

Let the discrete Fourier transform be defined by

$$y_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \exp\left(\frac{2\pi ijk}{N}\right) x_k, \qquad (j = 0, 1, \ldots, N).$$

The subroutines of package `RFT` compute this transform or its inverse

$$x_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \exp\left(\frac{-2\pi ijk}{N}\right) y_j, \qquad (k = 0, 1, \ldots, N)$$

for real functions, with the restriction that N is a power of 2.

Structure:

`SUBROUTINE` subprograms
User Entry Names: `RFT`, `RCA`, `RPA`, `RPS`, `RSA`
Internal Entry Names: `D700SU`
Files Referenced: Printer
`COMMMON` Block Names and Lengths: `/D700DT/ 6`, `/FWORK/ 321`

Usage:

```
CALL RFT(M,X,IX,Y,IY,MODE)      or
CALL RCA(M,X,IX,Y,IY)      or
CALL RPA(M,X,IX,Y,IY)      or
CALL RPS(M,X,IX,Y,IY)      or
CALL RSA(M,X,IX,Y,IY)
```

`M` (`INTEGER`) Number m (such that $n = 2^m$) of input values (full period or half period).

`X` (`REAL`) Input array. The input values are taken from $\mathtt{X}(k{*}\mathtt{IX}+1)$ for $k = 0, 1, \ldots, n$.

`Y` (`REAL`) Output array. The results are stored in $\mathtt{Y}(k{*}\mathtt{IY}+1)$ for $j = 0, 1, \ldots, n$.

`MODE` (`INTEGER`) Selects the mode of operation for `RFT` as follows:

`MODE = 1`: **Analysis of a general real function.**

```
CALL RFT(M,X,IX,Y,IY,1)      or
CALL RPA(M,X,IX,Y,IY)
```

assumes $x_k = \mathtt{X}(k{*}\mathtt{IX}+1)$ $(k = 0, 1, \ldots, n-1)$; $n = 2^m = N$ to define a full period of the function to be analysed. The value x_n is ignored. The results are returned in the following order:

$$y_0 = y_n = \mathtt{Y}(1)$$
$$y_j = y_{n-j} = \mathtt{Y}(j{*}\mathtt{IY}+1) + i\mathtt{Y}((j+n/2){*}\mathtt{IY}+1), \quad (j = 1, 2, \ldots, n/2).$$

The other values in `Y` are not changed.

`MODE = 4`: **Synthesis of a general real function.**

```
CALL RFT(M,X,IX,Y,IY,4)      or
CALL RPS(M,X,IX,Y,IY)
```

is exactly the inverse of `MODE=1` as described above. The value x_n is set equal to x_0.

`MODE=2/5`: **Analysis/Synthesis of a real even function.**
For an even function, the transform is identical to its inverse.

```
CALL RFT(M,X,IX,Y,IY,2)      or
CALL RFT(M,X,IX,Y,IY,5)      or
CALL RCA(M,X,IX,Y,IY)
```

all assume that $x_k = \mathtt{X}(k*\mathtt{IX}+1),\ (k = 0, 1, \ldots, n),\ n = 2^m = N/2$ define a *half-period* of the function to be analysed and that the other half period is generated by *even* continuation. The results returned are the cosine terms

$$y_j = y_{2n-j} = \mathtt{Y}(j*\mathtt{IY}+1), \quad (j = 0, 1, \ldots, n).$$

Note that the full period has $2n = N$ points.

`MODE = 3/6`: **Analysis/Synthesis of a real odd function.**
For an odd function the transform is also identical to its inverse. All assume that $x_k = \mathtt{X}(k*\mathtt{IX}+1),\ (k = 1, 2, \ldots, n)$;

```
CALL RFT(M,X,IX,Y,IY,3)      or
CALL RFT(M,X,IX,Y,IY,6)      or
CALL RSA(M,X,IX,Y,IY)
```

$n = 2^m = N/2$ define a *half-period* of the function to be analysed and that the other half period is generated by *odd* continuation. The results returned are the sine terms

$$y_j = -y_{2n-j} = \mathtt{Y}(j*\mathtt{IY}+1), \quad (j = 1, 2, \ldots, n).$$

Note that $y_0 = y_n = 0$ and that the values returned are $\mathtt{Y(1)} = \mathtt{X(1)}$ and $\mathtt{Y}(n*\mathtt{IY}+1) = \mathtt{X}(n*\mathtt{IX}+1)$. Again the full period has $2n = N$ points.

Restrictions:

These subroutines work for any input such that the *full period* has at least four points, i.e., $m \geq 2$ for general functions, or $m \geq 1$ for odd or even functions. If the number of data points exceeds 129 ($m \leq 7$), the calling program must provide sufficient working storage by using the statement

```
COMMON /FWORK/ W(nnn)
```

where $\mathtt{nnn} = 5 * 2^m$.

References:

1. C. Iselin, An approach to fast Fourier transform, CERN 71-19.

A copy of Ref. 1 is available.

•