

Fast Fourier Transform—FAST

The program FAST (**F**ast Fourier Transform) is a subroutine subprogram that performs the Fast Fourier Transform or Inverse Fast Fourier Transform of equally spaced complex type data using the Cooley-Tukey method. If the data are real numbers, it must be converted to complex numbers with zero imaginary part beforehand, and the number of data must be a power of two, so if the number of data is any other number, a complex number (0, 0) must be added after the actual data until the total number of data is a power of two .

FAST (Fast Fourier Transform)

【Purpose】

To perform a Fast Fourier Transform or an Inverse Fast Fourier Transform of given equally spaced complex type data.

【Usage】

(1) How to connect

CALL FAST (N, X, ND, IND)

Argument	Type	Parameter in calling program	Return Parameter
N	I	Total number of complex data X	Unchanged
X	C 1-D array (ND)	Equal interval complex data	Inverse Fourier Transform Value or N times the Fourier transform value
ND	I	Dimension size of X in calling program	Unchanged
IND	I	IND= -1 : Fourier Transform IND=+1 : Inverse Fourier Transform	Unchanged

(2) Necessary subroutines and function subprograms

None

(3) Remarks

- i) N must be a power of 2.
- ii) For the Fourier transform ($IND = -1$), the transform value must be divided by N.

【Program List】

C	*****	FAST	1
C	SUBROUTINE FOR FAST FOURIER TRANSFORM	FAST	2
C	*****	FAST	3
C		FAST	4
C	CODED BY Y. OHSAKI	FAST	5
C		FAST	6
C	PURPOSE	FAST	7
C	TO PERFORM FAST FOURIER OR INVERSE FOURIER TRANSFORM OF	FAST	8
C	A SERIES OF EQUI-SPACED DATA	FAST	9
C		FAST	10
C	USAGE	FAST	11
C	CALL FAST(N, X, ND, IND)	FAST	12
C		FAST	13
C	DESCRIPTION OF ARGUMENTS	FAST	14
C	N - TOTAL NUMBER OF COMPLEX DATA AND TRANSFORMED VALUES	FAST	15
C	X(ND) - EQUI-SPACED COMPLEX DATA/TRANSFORMED VALUES AT	FAST	16
C	CALL/RETURN	FAST	17
C	ND - DIMENSION OF X IN CALLING PROGRAM	FAST	18
C	IND - IND=-1 FOR FOURIER TRANSFORM	FAST	19
C	+1 FOR INVERSE FOURIER TRANSFORM	FAST	20
C		FAST	21
C	REMARKS	FAST	22
C	(1) N MUST BE EQUAL TO POWER OF 2	FAST	23
C	(2) WHEN IND=-1, TRANSFORMED VALUES ARE MULTIPLIED BY N	FAST	24
C	(3) EXAMPLE OF CALLING	FAST	25
C	COMPLEX A(1024)	FAST	26
C	DIMENSION DATA(1024)	FAST	27
C	DATA NN/1024/	FAST	28
C	DO 1 M=1, NN	FAST	29
C	A(M)=CMPLX(DATA(M), 0. 0)	FAST	30
C	1 CONTINUE	FAST	31
C	CALL FAST(NN, A, 1024, -1)	FAST	32
C		FAST	33
C	REFERENCE	FAST	34
C	N. M. BRENNER/ THREE FORTRAN PROGRAMS THAT PERFORM THE COOLEY-	FAST	35
C	TUKEY FOURIER TRANSFORM / MIT JULY 1967	FAST	36
C		FAST	37
C	SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED	FAST	38
C	NONE	FAST	39
C		FAST	40
C	SUBROUTINE FAST(N, X, ND, IND)	FAST	41
C		FAST	42
C	COMPLEX X(ND), TEMP, THETA	FAST	43
C		FAST	44
C	J=1	FAST	45
C	DO 140 I=1, N	FAST	46
C	IF(I. GE. J) GO TO 110	FAST	47
C	TEMP=X(J)	FAST	48
C	X(J)=X(I)	FAST	49
C	X(I)=TEMP	FAST	50

110 M=N/2	FAST	51
120 IF (J. LE. M) GO TO 130	FAST	52
J=J-M	FAST	53
M=M/2	FAST	54
IF (M. GE. 2) GO TO 120	FAST	55
130 J=J+M	FAST	56
140 CONTINUE	FAST	57
KMAX=1	FAST	58
150 IF (KMAX. GE. N) RETURN	FAST	59
ISTEP=KMAX*2	FAST	60
DO 170 K=1, KMAX	FAST	61
THETA=CMPLX (0. 0, 3. 141593*REAL (IND*(K-1)) /REAL (KMAX))	FAST	62
DO 160 I=K, N, ISTEP	FAST	63
J=I+KMAX	FAST	64
TEMP=X (J) *CEXP (THETA)	FAST	65
X (J)=X (I) -TEMP	FAST	66
X (I)=X (I) +TEMP	FAST	67
160 CONTINUE	FAST	68
170 CONTINUE	FAST	69
KMAX=ISTEP	FAST	70
GO TO 150	FAST	71
END	FAST	72

【Example 1】

Perform a Fourier transform of the sequence of numbers given in the DATA statement and print the real, imaginary, and absolute parts.

```

C
  COMPLEX C (16)
  DIMENSION DATA (16), AMP (16)
  DATA  NN/16/, DATA/5. 0, 32. 0, 38. 0, -33. 0,
&          -19. 0, -10. 0, 1. 0, -8. 0, -20. 0, 10. 0,
&          -1. 0, 4. 0, 11. 0, -1. 0, -7. 0, -2. 0/

C
  DO 110 M=1, NN
  C (M)=CMPLX (DATA (M), 0. 0)
110 CONTINUE
  CALL FAST (NN, C, 16, -1)

C
  DO 120 K=1, NN
  C (K)=C (K) /REAL (NN)
  AMP (K)=CABS (C (K))
120 CONTINUE
  WRITE (6, 601) NN, (K-1, C (K), AMP (K), K=1, NN)
  STOP

C
601 FORMAT (' EXAMPLE WAVE' // ' --- FINITE COMPLEX FOURIER COEFFICIENTS ---'
*          //T5,
*          ' TOTAL NUMBER OF DATA =', I3//T8, ' K', TR7, ' REAL (C)', TR4
*          ' IMAG (C)', TR4, ' ABS (C)' //
*          (I8, TR2, 3F11. 3) )
  END

```

Output :

EXAMPLE WAVE

-- FINITE COMPLEX FOURIER COEFFICIENTS --

TOTAL NUMBER OF DATA = 16

K	REAL (C)	IMAG (C)	ABS (C)
0	0.000	0.000	0.000
1	3.880	2.071	4.398
2	2.744	-4.190	5.009
3	2.479	-5.976	6.470
4	-3.375	-4.375	5.526
5	-2.094	1.928	2.846
6	-3.619	1.185	3.808
7	1.985	2.476	3.173
8	1.000	0.000	1.000
9	1.985	-2.476	3.173
10	-3.619	-1.185	3.808
11	-2.094	-1.928	2.846
12	-3.375	4.375	5.526
13	2.479	5.976	6.470
14	2.744	4.190	5.009
15	3.880	-2.071	4.398

【Example2】

Perform a Fourier transform and then an inverse Fourier transform of the sequence of numbers given in the DATA statement, and print the real and imaginary parts of the sequence of numbers, the transformed values, and the inverse transformed values.

```

C
  COMPLEX C(16), X(16)
  DIMENSION DATA(16)
  DATA NN/16/, DATA/5.0, 32.0, 38.0, -33.0,
&          -19.0, -10.0, 1.0, -8.0, -20.0, 10.0,
&          -1.0, 4.0, 11.0, -1.0, -7.0, -2.0/
C
  DO 110 M=1, NN
    C(M)=CMPLX(DATA(M), 0.0)
110 CONTINUE
C
  CALL FAST(NN, C, 16, -1)
  DO 120 K=1, NN
    C(K)=C(K)/REAL(NN)
    X(K)=C(K)
120 CONTINUE
C
  CALL FAST(NN, X, 16, +1)
C

```

```
WRITE(6, 601) (K-1, DATA(K), K-1, C(K), X(K), K=1, NN)
STOP
```

C

```
601 FORMAT(' EXAMPLE WAVE' // ' -- FOURIER AND INVERSE TRANSFORMS --'
*        //T7, ' M', TR3, ' DATA', TR7, ' K', TR4, ' FOURIER TRANSFORM', TR2,
*        ' INVERSE TRANSFORM' // (T6, I2, F7. 0, I8, TR5, 2F7. 3, TR5, 2F7. 3) )
END
```

Output:

EXAMPLE WAVE

-- FOURIER AND INVERSE TRANSFORMS --

M	DATA	K	FOURIER TRANSFORM	INVERSE TRANSFORM
0	5.	0	0.000 0.000	5.000 -0.000
1	32.	1	3.880 2.071	32.000 -0.000
2	38.	2	2.744 -4.190	38.000 -0.000
3	-33.	3	2.479 -5.976	-33.000 0.000
4	-19.	4	-3.375 -4.375	-19.000 0.000
5	-10.	5	-2.094 1.928	-10.000 -0.000
6	1.	6	-3.619 1.185	1.000 0.000
7	-8.	7	1.985 2.476	-8.000 -0.000
8	-20.	8	1.000 0.000	-20.000 0.000
9	10.	9	1.985 -2.476	10.000 -0.000
10	-1.	10	-3.619 -1.185	-1.000 -0.000
11	4.	11	-2.094 -1.928	4.000 0.000
12	11.	12	-3.375 4.375	11.000 -0.000
13	-1.	13	2.479 5.976	-1.000 0.000
14	-7.	14	2.744 4.190	-7.000 0.000
15	-2.	15	3.880 -2.071	-2.000 0.000