International Atomic Energy Agency

INDC-257E
October 1968

INDC INTERNATIONAL NUCLEAR DATA COMMITTEE

V.I. Popov, V.M. Sluchevskaja, V.I. Trykova

METHOD FOR THE INTERPOLATION, EVALUATION AND COMPACT PRESENTATION OF INELASTIC NEUTRON SCATTERING DATA

IAEA NUCLEAR DATA UNIT, KARNTNERRING 11, A-1010 VIENNA

INDC - 257E
October 1968

Note:

This document (INDC-257E) is an English translation of an original (INDC-257) in Russian. The data tables 1-7 have been entered in the neutron data file DASTAR of the IAEA Nuclear Data Unit, under the following accession-numbers:

DASTAR-00721	Niobium (table 1)
DASTAR-00722	Iron (table 2)
DASTAR-00723	Nickel 58 (table 3)
DASTAR-00724	Nickel 60 (table 4)
DASTAR-00725	Lead 206 (table 5)
DASTAR-00726	Lead 207 (table 6)
DASTAR-00727	Lead 208 (table 7)

The data holdings of the IAEA Nuclear Data Unit are regularly publicized in "CINDU". The data can be obtained through one of the following data centers:

IAEA Nuclear Data Unit	Vienna, Austria
ENEA Neutron Data Compilation Centre	Saclay, France
National Neutron Cross Section Center	Brookhaven, U.S.A.
Informacionnyj Centr po Jadernym Dannym	Obninsk, U.S.S.R.

IAEA Nuclear Data Unit
Kärntnerring 11-13
A-1010 Vienna, Austria

68-8166
Translated from Russian

V.I. Popov, V.M. Sluchevskaya, V.I. Trykova

METHOD FOR THE INTERPOLATION, EVALUATION AND COMPACT PRESENTATION OF INELASTIC NEUTRON SCATTERING DATA

The authors propose a method for the interpolation, evaluation and compact presentation of data on the spectra of inelastically scattered neutrons [1], for subsequent use especially in calculating reactor constants.

All inelastic neutron scattering data take the form of a set of excitation functions for groups of nuclear levels with width 0.2 MeV, such that the mean energies in each group are 0.1; 0.3; 0.5; .... MeV. The excitation function of each group of levels is approximated by an analytical form of the type:

$$S(x)=\sum_{i=1}^{3} C_i x^{d_i} \exp(-b_i x),$$

where X represents the excess initial energy of the neutrons above the excitation threshold for the level. C_i and d_i are expressed in turn through the parameters A, T, f, F as follows:

$$C_1 = A T^{-d_1}(1-f-F)\exp(d_1);$$

$$C_2 = A T^{-d_2} f \exp(d_2);$$

$$C_3 = A T^{-d_3} F \exp(d_3);$$

$$d_i = b_i T.$$

Seven parameters of an approximating function (A, T, b_1, b_2, b_3, f, F) are found for each group of levels by comparison with the experimental data.

As a result, all information on inelastic neutron scattering for one nucleus in the initial energy range 0-15 MeV reduces to values for the 120 constants set out in Table 1 for niobium-93.

The first eight numbers in Table 1 indicate the numbers of the first eight conditional groups of levels containing as little as one real excitation level of the nucleus. The following 56 numbers indicate the values of seven parameters for each of the first eight levels. The remaining 56 constants are coefficients of the power series expansion of the energy dependence of each of the seven parameters:

$$A(E)=\sum_{i=0}^{7} a_i E^i .$$

We can obtain the necessary information on inelastic neutron scattering by using the constants given in Table 1 and a computer programme. In particular, it is simple to calculate the spectra of inelastically scattered neutrons, level excitation functions, inelastic scattering cross-sections, grouped inelastic transition cross-sections, etc. The appropriate programmes in ALGOL-60 for the calculation of these values are given below.

The programme (No. 1) for calculating the spectra brings in as single figures the atomic weight of the nucleus (AT), the energy spacing (h), the block of numbers "M", consisting of the coefficients in Table 1 less the first column, which forms the following block "L", and, finally, the initial neutron energies (EN). The spectra are printed out with the given spacing h, the first point corresponding to zero energy. With a spacing of 0.2 MeV, the spectra are normalized with reference to the full inelastic cross-sections (in barns) plus the doubled cross-section for the (n, 2n) reaction.

The programme (No. 2) for calculating grouped inelastic transition cross-sections is based on formula 33 of Reference [2]. Here, the initial neutron spectrum is assumed to be close to the reactor spectrum, i.e. in the form $1/E$ where $E \leq 2$ MeV and $\exp(-0.72\,E)$ with $E > 2$ MeV. The initial data for this programme are: integration step (h2, not greater than 0.05), number of groups (jj, not greater than 25), atomic weight (AT), number block "M" and "L" (the same as in the previous programme), block of boundary energies of the groups in descending order (El [1:26] - not more than 15 MeV). When there are less than 25 groups, block E 1 is increased to 26 by means of any differing numbers.

Programme No. 3 prints out the total inelastic neutron scattering cross-section (in barns) with the addition of the doubled cross-section for the (n, 2n) reaction as a function of the initial neutron energy. The energy spacing is 0.1 MeV, the first point corresponding to zero energy.

Programme No. 4 calculates the excitation function (in barns) of a group of nuclear levels with width 0.2 MeV and a given mean energy E.

Tables 2-7 contain co-efficients for natural iron and isotopes of lead and nickel.

REFERENCES

1. POPOV, V.I., SLUCHEVSKAYA, V.M., TRYKOVA, V.I: Paper ACC-68/11, Anglo-Soviet Seminar, Dubna, 1968.

2. Voprosy fisiki zaščity reaktorov (Problems of reactor shielding physics): Atomizdat, 1966.

Table 1

Niobium

Level No.	A	T	b_1	b_2	b_3	ψ	F
4	0,350	0,430	3,I0	0,700	0,350	0,300	0,0I9
5	0,760	0,4I0	2,3I	0,900	0,340	0,300	0,020
6	0,5I0	0,430	6,40	0,450	0,340	0,250	0,02I
7	0,425	0,420	6,20	0,900	0,330	0,500	0,022
8	0,400	0,4I0	8,00	0,800	0,320	0,500	0,023
9	0,387	0,405	9,5I	0,820	0,3I0	0,340	0,024
I0	0,378	0,392	8,92	0,790	0,300	0,370	0,027
II	0,370	0,384	8,47	0,790	0,290	0,400	0,030

Table 1 continued

	A^*	T^*	b_1^*	b_2^*
a_0	0,502325	0,93569I	$0,294764.I0^{2}$	$0,472300.I0^{I}$
a_1	$-0,995I3I.I0^{-I}$	-0,683I07	$-0,260377.I0^{2}$	$-0,56I05I.I0^{I}$
a_2	$0,2649I0.I0^{-I}$	C,322234	$0,I299I0.I0^{2}$	$0,292853.I0^{I}$
a_3	$-0,54235I.I0^{-2}$	$-0,772896.I0^{-I}$	$-0,343II7.I0^{I}$	-0,7I404I
a_4	$0,540908.I0^{-3}$	$0,I08208.I0^{-I}$	0,49536I	$0,92I294.I0^{-I}$
a_5	$-0,I85269.I0^{-4}$	$-0,878239.I0^{-3}$	$-0,394760.I0^{-I}$	$-0,653324.I0^{-2}$
a_6	$-0,3266674.I0^{-6}$	$0,379446.I0^{-4}$	$0,I63303.I0^{-2}$	$0,24I504.I0^{-3}$
a_7	$0,233698.I0^{-7}$	$-0,67I972.I0^{-6}$	$-0,2740I6.I0^{-4}$	$-0,364573.I0^{-5}$

	b_3^*	f^*	F^*
a_0	0,753595	0,804476	-0,I74II7
a_1	-0,632499	-0,898I5I	0,29III0
a_2	0,366272	0,560I34	-0,I66098
a_3	-0,II4493	-0,I37I40	$0,485730.I0^{-I}$
a_4	$0,I94023.I0^{-I}$	$0,I62376.I0^{-I}$	$-0,769760.I0^{-2}$
a_5	$-0,I76335.I0^{-2}$	$-0,99062I.I0^{-3}$	$0,667053.I0^{-3}$
a_6	$0,889324.I0^{-4}$	$0,294624.I0^{-4}$	$-0,297098.I0^{-4}$
a_7	$-0,I47439.I0^{-5}$	$-0,325788.I0^{-6}$	$0,53I773.I0^{-6}$

Iron

Table 2

№							
5	0,880	I,44	I,50	0,75	0,2I	0,55	0,044
8	0,060	I,I8	I,30	0,60	0,2I	0,50	0,053
II	0,I53	I,65	I,50	I,50	0,2I	0,45	0,070
I4	0,I70	0,93	I,50	I,50	0,2I	0,40	0,098
I5	0,I55	0,90	I,50	I,50	0,2I	0,30	0,II5
I6	0,I42	0,80	2,50	I,05	0,2I	0,40	0,I20
I7	0,II5	0,75	2,50	I,20	0,2I	0,40	0,I40
I8	0,II5	0,75	2,50	I,20	0,2I	0,40	0,I40

a				
	-0,726993	0,I04858	0,I92200.I0	0,II96600.I0
	0,884885	0,942695	$0,700000.I0^{-2}$	$-0,740000.I0^{-I}$
	-0,370570	-0,569410	0	0
	$0,8I6060.I0^{-I}$	0,I65627	0	0
	$-0,I03238.I0^{-I}$	$-0,250409.I0^{-I}$	0	0
	$0,750041.I0^{-3}$	$0,207589.I0^{-2}$	0	0
	$-0,289680.I0^{-4}$	$-0,903437.I0^{-4}$	0	0
	$0,460004.I0^{-6}$	$0,I6I447.I0^{-5}$	0	0

0,2I0	0,390	0,I40
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Nickel 58

Table 3

8	0,750	I,I0	I,70	I,30	0,I3	0,40	0,I8
I3	0,I50	I,20	3,00	2,00	0,25	0,40	0,I8
I4	0,I70	0,98	3,00	2,00	0,25	0,40	0,I8
I5	0,I42	I,03	2,74	I,84	0,25	0,40	0,I8
I6	0,I34	0,98	2,48	I,68	0,25	0,40	0,I8
I7	0,II2	0,94	2,20	I,50	0,25	0,40	0,I8
I8	0,I07	0,90	2,06	I,44	0,25	0,40	0,I8
I9	0,I03	0,87	I,94	I,34	0,25	0,40	0,I8

0,I4II4I	0,II246I.I0^{2}	0,I09.I0	0,I30.I0
-0,220868	-0,968317.I0	0,200	0
0,278219.I0^{-2}	0,365318.I0	0	0
-0,366I06.I0^{-3}	-0,728360	0	0
0,528589.I0^{-4}	0,835490.I0^{-I}	0	0
-0,532945.I0^{-5}	-0,552069.I0^{-2}	0	0
0,278046.I0^{-6}	0,I94508.I0^{-3}	0	0
-0,560I05.I0^{-8}	-0,282I38.I0^{-5}	0	0

0,250	0,400	0,I80
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Nickel 60

Table 4

7	0,583	I,27	I,60	I,30	0,I2	0,40	0,I8
II	0,I80	0,94	3,00	2,30	0,25	0,40	0,I8
I2	0,I80	0,94	3,00	2,30	0,25	0,40	0,I8
I3	0,I50	0,80	2,50	2,50	0,25	0,40	0,I8
I4	0,I50	0,80	2,50	2,50	0,25	0,40	0,I8
I6	0,I23	0,80	2,I8	I,96	0,25	0,40	0,I8
I7	0,II6	0,79	2,00	I,50	0,25	0,40	0,I8
I8	0,I08	0,8I	I,92	I,44	0,25	0,40	0,I8

$0,470809.I0^{-4}$	$0,II246I.I0^{2}$	0,I09.I0	0,I30.I0
0,I86409	$-0,9683I7.I0$	0,200	0
$-0,925975.I0^{-I}$	$0,365318.I0$	0	0
$0,207398.I0^{-I}$	-0,728360	0	0
$-0,252979.I0^{-2}$	$0,835490.I0^{-I}$	0	0
$0,I73893.I0^{-3}$	$-0,552069.I0^{-2}$	0	0
$-0,634I84.I0^{-5}$	$0,I94508.I0^{-3}$	0	0
$0,956325.I0^{-7}$	$-0,282I38.I0^{-5}$	0	0

Table 4 continued

0,250	0,400	0,I80
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Table 5

5	0,488	I,66	I,50	I,50	0,II	0,60	0,080
7	0,250	I,38	I,50	I,50	0,II	0,60	0,I04
8	0,400	I,20	I,50	0,6I	0,II	0,60	0,II2
9	0,320	I,55	I,65	I,65	0,II	0,60	0,I30
II	0,I70	I,I5	I,70	I,70	0,II	0,60	0,I40
I2	0,I70	I,I5	I,75	I,75	0,II	0,60	0,I40
I4	0,I70	I,00	I,58	0,98	0,II	0,60	0,I40
I6	0,I70	I,00	I,83	0,60	0,II	0,60	0,I40

0,I02354.I0	0,268355.I0	0,I73.I0	0,I37634.I0
-0,969240	-0,538958	$-0,560.I0^{-I}$	-0,I23902.I0
0,440724	-0,27I3I8	0	0,524766
-0,I84I68	0,I46782	0	$-0,97I397.I0^{-I}$
$0,I38384.I0^{-I}$	$-0,265253.I0^{-I}$	0	$0,959970.I0^{-2}$
$-0,I03709.I0^{-2}$	$0,233055.I0^{-2}$	0	$-0,525617.I0^{-3}$
$0,409733.I0^{-4}$	$-0,I0II66.I0^{-3}$	0	$0,I49935.I0^{-4}$
$-0,663906.I0^{-6}$	$0,I74077.I0^{-5}$	0	$-0,I727I7.I0^{-6}$

0,II	0,6	0,I4
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Lead 207

Table 6

3	0,730	I,60	2,00	I,35	0,II	0,60	0,I4
5	0,560	I,78	2,00	2,00	0,II	0,60	0,I4
9	0,I30	I,30	2,00	I,40	0,II	0,60	0,I4
I2	0,I40	I,26	4,00	I,55	0,II	0,60	0,I4
I4	0,420	I,04	2,50	0,95	0,II	0,60	0,I4
I6	0,320	0,90	I,80	I,20	0,II	0,60	0,I4
I8	0,220	I,60	2,00	I,70	0,II	0,60	0,I4
I9	0,200	0,80	3,00	I,50	0,II	0,60	0,I4

0,I02354.I0	0,268355.I0	0,I73.I0	0,I37634.I0
-0,969240	-0,538958	$-0,560.I0^{-I}$	-0,I23902.I0
0,440724	-0,27I3I8	0	0,524766
-0,I04I68	0,I46782	0	$-0,97I397.I0^{-I}$
$0,I38384.I0^{-I}$	$-0,265253.I0^{-I}$	0	$0,959970.I0^{-2}$
$-0,I03709.I0^{-2}$	$0,233055.I0^{-2}$	0	$-0,525617.I0^{-3}$
$0,409733.I0^{-4}$	$-0,I0II66.I0^{-3}$	0	$0,I49935.I0^{-4}$
$-0,663906.I0^{-6}$	$0,I74077.I0^{-5}$	0	$-0,I727I7.I0^{-6}$

0,II	0,6	0,I4
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Lead 208

Table 7

I4	0,700	I,30	3,5	I,0	0,I8	0,5	0,I8
I6	0,203	0,65	4,0	I,0	0,I8	0,4	0,I8
I7	0,202	0,66	4,0	I,0	0,I8	0,4	0,I8
I8	0,200	0,66	4,0	I,0	0,I8	0,4	0,I8
I9	0,200	0,77	4,0	I,0	0,I8	0,5	0,I8
20	0,244	0,75	3,0	I,0	0,I8	0,6	0,I8
2I	0,238	0,72	3,0	I,0	0,I8	0,6	0,I8
22	0,234	0,72	3,0	I,0	0,I8	0,6	0,I8

-0,I20828.I0	0,268355.I0	0,362.I0
0,787958	-0,538958	-0,I52
-0,I2043I	-0,27I3I8	0
$-0,733367.I0^{-2}$	0,I46782	0
$0,3975I2.I0^{-2}$	$-0,265253.I0^{-I}$	0
$-0,440I08.I0^{-3}$	$0,233055.I0^{-2}$	0
$0,2I0243.I0^{-4}$	$-0,I0II66.I0^{-3}$	0
$-0,379657.I0^{-6}$	$0,I74077.I0^{-5}$	0

I	0,I8	0,6	0,I8
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

ALGOL programme No. 1 for calculating the recommended inelastically scattered neutron spectra.

```
begin real a, b, h, g, x, t, E, E1, EN, A, T, b1, b2, b3,
      f, F, d1, d2, d3, C1, C2, C3, En, AT, SN, W;
integer i, j, k, m, n;
array M[1:112], Aa, Ta, ba1, ba2, ba3, fa, Fa[1:50],
  Ab, Tb, bb1, bb2, bb3, fb, Fb[1:8], S, SP[0:150];
integer array L[1:8];
inout ('2 10-2', AT, h, M, L);
L3: inout ('2 10-2', EN); n:=L[8];
for i:=1 step 1 until n do begin
Aa[i]:=Ta[i]:=ba1[i]:=ba2[i]:=ba3[i]:=fa[i]:=Fa[i]:=0
                                    end;
for i:=1 step 1 until 8 do begin j:=L[i];
Aa[j]:=M[i]; Ta[j]:=M[i+8]; ba1[j]:=M[i+16];
ba2[j]:=M[i+24]; ba3[j]:=M[i+32]; fa[j]:=M[i+40];
Fa[j]:=M[i+48]; Ab[i]:=M[i+56]; Tb[i]:=M[i+64];
bb1[i]:=M[i+72]; bb2[i]:=M[i+80]; bb3[i]:=M[i+88];
fb[i]:=M[i+96]; Fb[i]:=M[i+104]
                                    end;
SN:=0; g:=AT/(AT+1);
for j:=0 step 1 until 150 do begin
E1:=j*h; E:=EN*g-E1/g;
if E<0 then begin SP[j]:=0; goto L1 end;
for i:=1 step until n do begin
if E>=0.2*(i-1) then begin if E<0.2*i then begin
A:=Aa[i]; T:=Ta[i]; b1:=ba1[i]; b2:=ba2[i];
b3:=ba3[i]; f:=fa[i]; F:=Fa[i]; goto L2 end
                end end;
```

```
      A:=A6[8]; T:=T6[8]; b1:=b61[8]; b2:=b62[8];
      b3:=b63[8]; f:=f6[8]; F:=F6[8];
      for k:=7 step -1 until 1 do begin
      A:=A*E+A6[k]; T:=T*E+T6[k]; b1:=b1*E+b61[k];
      b2:=b2*E+b62[k]; b3:=b3*E+b63[k]; f:=f*E+f6[k];
      F:=F*E+F6[k]                                    end;
L2:   a1:=b1*T; a2:=b2*T; a3:=b3*T;
      if T>0 then b:=ln(T) else b:=0;
      C1:=A*exp(a1*(1-b))*(1-f-F);
      C2:=A*exp(a2*(1-b))*f; C3:=A*exp(a3*(1-b))*F;
      En:=E/g; x:=EN-En; if x>0 then begin a:=ln(x);
      W:=C1*exp(a1*a-b1*x)+C2*exp(a2*a-b2*x)+
      C3*exp(a3*a-b3*x)                    end else W:=0;
      SP[j]:=W/g; SN:=SN+SP[j];
L1:   end; SN:=SN*h/0.2;
      inout('p2-10', EN, SN, SP); goto L3 end;
```

ALGOL programme No. 2 for calculating grouped inelastic transition cross-sections.

```
begin real a, b, h1, h2, h3, A, T, b1, b2, b3, f, F, d1, d2, d3
E, EN, En, AT, g, t, C1, C2, C3, S, Cd, al, bb, x, W;
integer i, j, k, m, n, jj, l, ii;
array M[1:112], Aa, Ta, ba1, ba2, ba3, fa, Fa[1:50];
Ab, Tb, bb1, bb2, bb3, fb, Fb[0:8], MA, MT, Mb1, Mb2, Mb3,
Mf, MF[1:150], E1[0:25], C2[0:30], Sm[1:25],
SG[1:25], G[1:10];
integer array L[1:8];
inout ('2 10-2', h2, jj, AT, M, L, E1);
h1:=0.1; h3:=0.5; g:=AT/(AT+1); n:=L[8];
for i:=1 step 1 until n do begin
Aa[i]:=Ta[i]:=ba1[i]:=ba2[i]:=ba3[i]:=fa[i]:=Fa[i]:=0
                                    end;
for i:=1 step 1 until 8 do begin j:=L[i];
Aa[j]:=M[i]; Ta[j]:=M[i+8]; ba1[j]:=M[i+16];
ba2[j]:=M[i+24]; ba3[j]:=M[i+32]; fa[j]:=M[i+40];
Fa[j]:=M[i+48]; Ab[i]:=M[i+56]; Tb[i]:=M[i+64];
bb1[i]:=M[i+72]; bb2[i]:=M[i+80]; bb3[i]:=M[i+88];
fb[i]:=M[i+96]; Fb[i]:=M[i+104]
                                    end;
for j:=1 step 1 until 150 do begin
E:=(2*j-1)*h1/2;
for i:=1 step 1 until n do begin
if E≥0.2*(i-1) then begin if E<0.2*i then begin
A:=Aa[i]; T:=Ta[i]; b1:=ba1[i]; b2:=ba2[i];
b3:=ba3[i]; f:=fa[i]; F:=Fa[i]; goto L2 end
                                    end end;
A:=Ab[8]; T:=Tb[8]; b1:=bb1[8]; b2:=bb2[8];
b3:=bb3[8]; f:=fb[8]; F:=Fb[8];
```

```
for k := 7 step -1 until 1 do begin
A := A*E + A6[k]; T := T*E + T6[k]; 61 := 61*E + 661[k];
62 := 62*E + 662[k]; 63 := 63*E + 663[k]; f := f*E + f6[k];
F := F*E + F6[k]                                    end;
L2: MA[j] := A, MT[j] := T; M61[j] := 61; M62[j] := 62;
M63[j] := 63; Mf[j] := f; MF[j] := F
                                                    end;
for j := 1 step 1 until 25 do begin S := 0;
EN := E1[j] - h2/2;
L4: EN := EN + h2; if EN > E1[j-1] + h2/2 then goto L5;
if EN < 2 then Cd := 1.042/EN else Cd := exp(-0.72*EN);
if abs(EN - E1[i-1]) > (h2/2 + 10^-4) then m := 1
else begin Cd := Cd*(E1[j-1] - (EN - h2/2))/h2; m := 0 end;
S := S + Cd; if m = 1 then goto L4;
L5: Sm[j] := S*h2/(E1[j-1] - E1[j])
                                                    end;
for i := 1 step 1 until 25 do SG[i] := 0;
for j := 1 step 1 until 25 do begin
for k := 1 step 1 until 25 do begin
S := 0; if j > jj then goto L8;
if k > jj then goto L7;
if k < j then goto L3;
for l := 1 step 1 until 150 do begin E := h1*(l - 0.5);
al := E1[k-1] + E/g; if E1[j-1] < al then al := E1[j-1];
bl := E1[k] + E/g; if E1[j] > bl then bl := E1[j],
if al > bl then begin EN := bl - h2/2;
L1: EN := EN + h2; if EN > al + h2/2 then goto L6;
if EN < 2 then Cd := 1.042/EN else Cd := exp(-0.72*EN);
A := MA[l], T := MT[l]; 61 := M61[l]; 62 := M62[l];
63 := M63[l]; f := Mf[l]; F := MF[l], A1 := 61*T;
A2 := 62*T; A3 := 63*T;
```

```
        if T>0 then b:=ln(T) else b:=0;
        C1:=A*exp(d1*(1-b))*(1-f-F);
        C2:=A*exp(d2*(1-b))*f; C3:=A*exp(d3*(1-b))*F;
        En:=E/g; x:=EN-En;
        if x>0 then begin a:=ln(x);
        W:=C1*exp(d1*a-b1*x)+C2*exp(d2*a-b2*x)+
        C3*exp(d3*a-b3*x) end else W:=0;
        if abs(EN-al)>(h2/2+10-4) then m:=1
        else begin W:=W*(al-(EN-h2/2))/h2; m:=0 end;
        S:=S+W*Cd; if m=1 then goto L1 end; L6: end;
        S:=S*h2*h1/0.2/Sm[j]/(E1[j-1]-E1[j]);
L3:     SG[k]:=S end; L7: inout('p2-10',j, SG) end;
L8:     end;
```

ALGOL programme No. 3 for calculating recommended inelastic neutron cross-sections.

```
begin real a, b, h, g, t, x, W, E, E1, EN, En, B1, B2, B3,
F1, f1, D1, D2, D3, AT;
integer i, j, k, m, n;
array M[1:112], A, T, b1, b2, b3, f, F, C1, C2, C3, d1,
d2, d3[1:76], SN[0:150], Ab, Tb, bb1, bb2, bb3, fb,
Fb[1:8];
integer array L[1:8];
inout ('2 10-2', AT, M, L);
g := AT/(AT+1); h := 0.1; SN[0] := 0; n = L[8];
for i:=1 step 1 until n do begin
A[i]:=T[i]:=b1[i]:=b2[i]:=b3[i]:=f[i]=F[i]:=0
                                    end;
for i:=1 step 1 until 8 do begin j:=L[i];
A[j]:=M[i]; T[j]:=M[i+8]; b1[j]:=M[i+16];
b2[j]:=M[i+24]; b3[j]:=M[i+32]; f[j]:=M[i+40];
F[j]:=M[i+48]; Ab[i]:=M[i+56]; Tb[i]:=M[i+64];
bb1[i]:=M[i+72]; bb2[i]:=M[i+80]; bb3[i]:=M[i+88];
fb[i]:=M[i+96]; Fb[i]:=M[i+104]
                                    end;
m:=n-1;
for i:=m step 1 until 76 do begin
E:=(2*i-1)*0.1;
a:=Ab[8]; t:=Tb[8]; B1:=bb1[8]; B2:=bb2[8];
B3:=bb3[8]; F1:=fb[8]; f1:=Fb[8];
for k:=7 step -1 until 1 do begin
a:=a*E+Ab[k]; t:=t*E+Tb[k]; B1:=B1*E+bb1[k];
B2:=B2*E+bb2[k]; B3:=B3*E+bb3[k];
F1=F1*E+fb[k]; f1:=f1*E+Fb[k]
                                    end;
```

```
A[i]:=a, T[i]:=t, b1[i]:=B1; b2[i]:=B2; b3[i]:=B3;
f[i]:=F1, F[i]:=f1    end;
for i:=1 step 1 until 76 do begin
d1[i]:=b1[i]*T[i]; d2[i]:=b2[i]*T[i]; d3[i]:=b3[i]*T[i];
if T[i]>0 then b:=ln(T[i]) else b:=0;
C1[i]:=A[i]*exp(d1[i]*(1-b))*(1-f[i]-F[i]);
C2[i]:=A[i]*exp(d2[i]*(1-b))*f[i];
C3[i]:=A[i]*exp(d3[i]*(1-b))*F[i]
                                    end;
for j:=1 step 1 until 150 do begin
SN[j]:=0; EN:=h*j;
for i:=1 step 1 until 76 do begin
E:=(2*i-1)*0.1; En:=E/g; x:=EN-En;
if x>0 then begin a:=ln(x);
    W:=C1[i]*exp(d1[i]*a-b1[i]*x)+
       C2[i]*exp(d2[i]*a-b2[i]*x)+
       C3[i]*exp(d3[i]*a-b3[i]*x)
          end else W:=0;
SN[j]:=SN[j]+W                      end
                                    end;
inout('p2-10', SN)                  end;
```

ALGOL programme No. 4 for calculating recommended excitation functions for groups of nuclear levels.

```
begin real a, b, h, g, x, t, E, E1, EN, En, A, T, b1, b2, b3,
      f, F, d1, d2, d3, C1, C2, C3, AT, SN, W;
integer i, j, k, m, n;
array M[1:112], Aa, Ta, ba1, ba2, ba3, fa, Fa[1:50],
  Ab, Tb, bb1, bb2, bb3, fb, Fb[1:8], S, SP[0:150];
integer array L[1:8];
inout ('2 10-2', AT, h, M, L);
L3: inout ('2 10-2', E);
n:= L[8];
for i:=1 step 1 until n do begin
Aa[i]:= Ta[i]:= ba1[i]:= ba2[i]:= ba3[i]:= fa[i]:= Fa[i]:= 0
                                        end;
for i=1 step 1 until 8 do begin j:=L[i];
Aa[j]:= M[i]; Ta[j]:= M[i+8]; ba1[j]:= M[i+16];
ba2[j]:= M[i+24]; ba3[j]:= M[i+32]; fa[j]:= M[i+40];
Fa[j]:= M[i+48]; Ab[i]:= M[i+56]; Tb[i]:= M[i+64];
bb1[i]:= M[i+72]; bb2[i]:= M[i+80]; bb3[i]:= M[i+88];
fb[i]:= M[i+96]; Fb[i]:= M[i+104]
                                        end;
g:= AT/(AT+1);
for i:=1 step 1 until n do begin
if E≥0.2*(i-1) then begin if E<0.2*i then begin
A:= Aa[i]; T:= Ta[i]; b1:= ba1[i]; b2:= ba2[i];
b3:= ba3[i]; f:= fa[i]; F:= Fa[i]; goto L2      end
                                    end end;
A:= Ab[8]; T:= Tb[8]; b1:= bb1[8]; b2:= bb2[8]; b3:= bb3[8];
f:= fb[8]; F:= Fb[8];
for k:=7 step -1 until 1 do begin
A:= A*E + Ab[k]; T:= T*E + Tb[k]; b1:= b1*E + bb1[k];
b2:= b2*E + bb2[k]; b3:= b3*E + bb3[k]; f:= f*E + fb[k];
F:= F*E + Fb[k]                         end;
```

```
L2: d1:=b1*T; d2:=b2*T; d3:=b3*T;
    if T>0 then b:=ln(T) else b:=0;
    C1:=A*exp(d1*(1-b))*(1-f-F);
    C2:=A*exp(d2*(1-b))*f; C3:=A*exp(d3*(1-b))*F;
    En:=E/g;
    for i:=0 step 1 until 150 do begin
    EN:=i*h; x:=EN-En;
    if x>0 then begin a:=ln(x);
    W:=C1*exp(d1*a-b1*x)+C2*exp(d2*a-b2*x)+
    C3*exp(d3*a-b3*x)
                    end else W:=0; S[i]:=W
                                        end;
    inout('p2-10', E, S); goto L3
    end;
```