

Hamburg, den 9.12.1965

G. Lutz/P

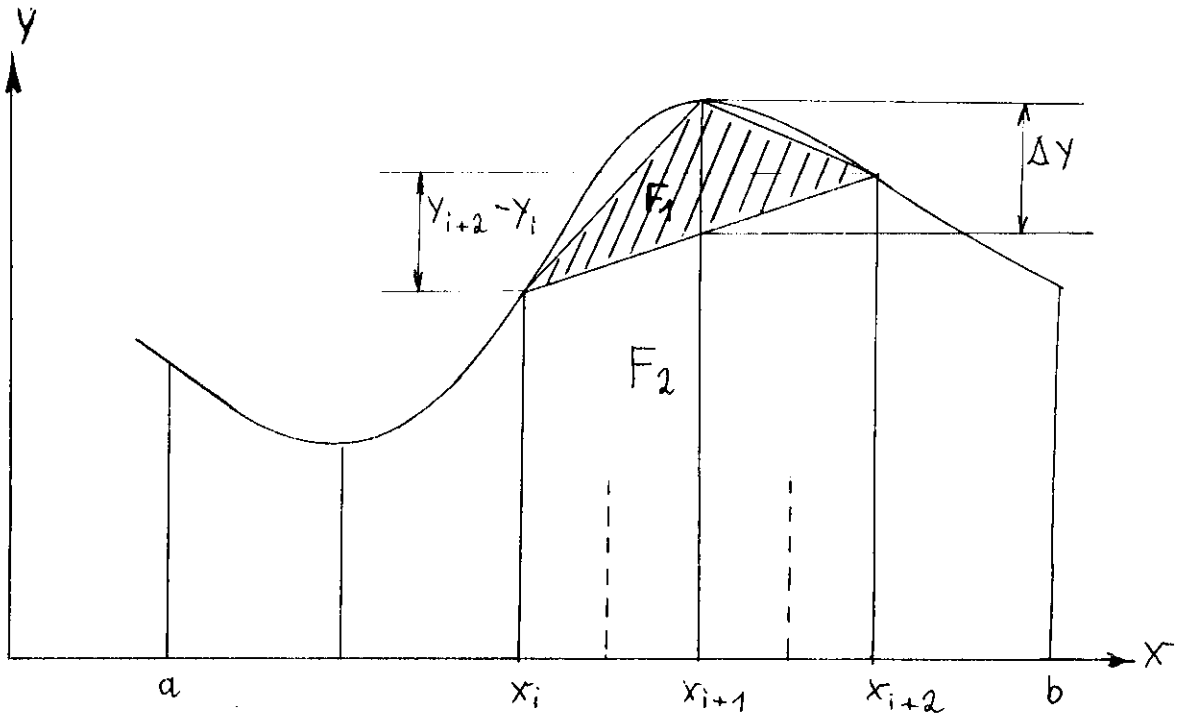
Allgemeines Integrationsprogramm mit variabler Schrittweite

Das Programm, dessen Funktionsweise im folgenden beschrieben wird, paßt die Schrittweite der jeweiligen Funktion an, über die integriert wird. Durch die dabei verwendete spezielle Methode ist es möglich, auch über unstetige Funktionen zu integrieren. Die Genauigkeit des Resultats hängt von der Art der Funktion und verschiedenen frei wählbaren Parametern ab (siehe Beschreibung).

Das Programm hat sich in vielen Anwendungen als nützlich erwiesen und soll deshalb einem breiteren Kreis zur Kenntnis gebracht werden.

Das Integral habe die Form

$$Z = \int_a^b Y(x) dx$$



Die einzelnen Schritte des Programms sind:

- 1.) Einteilung des Integrationsbereichs in gleiche Intervalle (Hauptprogramm)
- 2.) Prüfung, ob die Intervalle noch verkleinert werden sollen (Unterprogramm).

Es werden für je drei aufeinander-folgende Punkte X_i, X_{i+1}, X_{i+2} die Verhältnisse $V_1 = \frac{\Delta Y}{Y_{i+2} - Y_i}$ und $V_2 = \frac{F_1}{F_2}$ (zur Definition von $\Delta Y, F_1, F_2$ siehe Abb.) gebildet.

Ist

- a) $|V_1| > V_{1M}$ oder $|Y_{i+2} - Y_i| < 10^{-8}$
und
- b) $|V_2| > V_2$ oder $|F_2| < 10^{-8} \cdot (b - a)$
und
- c) $|F_1| > 10^{-10} (b - a)$

so werden zwei neue x-Werte in der Mitte der beiden Intervalle berechnet und gespeichert. Sind beim vorhergehenden Intervall bereits solche Zwischenwerte berechnet worden, so entfällt der linke Zwischenpunkt.

Die Bedingungen

$$|Y_{i+2} - Y_i| < 10^{-8} \quad \text{und} \quad |F_2| < 10^{-8}(b-a)$$

sollen verhindern, dass $|V_1|$ bzw. $|V_2|$ grösser als 10^{38} wird.

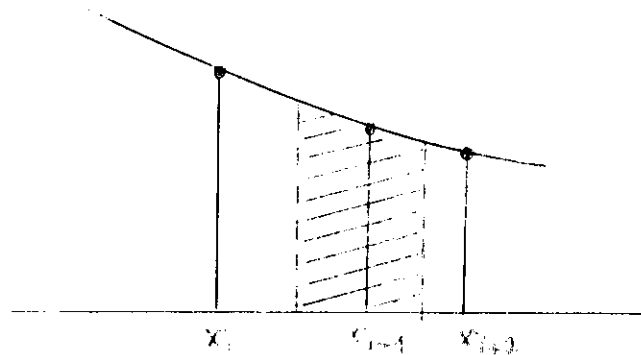
$|F_1| > 10^{-10}(b-a)$ verhindert, dass für $Y(x) = 0$ immer neue Zwischenpunkte berechnet werden.

- 3.) Berechnung der Funktionswerte für die neuen Punkte im Hauptprogramm
- 4.) Umordnung der Werte nach steigendem x, (UP)
- 5.) Prüfung wie 2.).

Die Prozedur wird solange wiederholt, bis

- a) keine Zwischenpunkte mehr berechnet werden oder
- b) eine vorgegebene Anzahl von Wiederholungen erreicht ist oder
- c) die Anzahl der Punkte 90 überschritten hat.

Hierauf folgt die Integration, bei der die Funktion stückweise durch ein durch drei aufeinanderfolgende Punkte gehendes Polynom 2. Grades ersetzt wird. Die Integrationsgrenzen liegen immer in der Mitte aufeinanderfolgender Intervalle. (Ausgenommen 1. und letztes Intervall).



Es sind Vorkehrungen dagegen getroffen worden, dass Differenzen fast gleich grosser Zahlen und Multiplikationen und Divisionen von sehr grossen oder kleinen Zahlen Fehlresultate zur Folge haben.

Im Anhang werden zwei Beispiele gegeben. Das erste Beispiel ist eine Integration über eine Variable. Ein Teil der im Hauptprogramm vorkommenden Karten wird für jede Anwendung unverändert übernommen. Diese sind mit IR gekennzeichnet. Die Berechnung des Funktionswertes $YR(NXR)$ erfolgt jeweils zwischen IR 25 und IR 101, dabei ist XR die Variable über die integriert wird.

Die Bedeutung einiger wichtiger im Hauptprogramm vorkommender Grössen ist in der folgenden Tabelle zusammengestellt:

Bezeichnung	Bedeutung	Zahlenwert einzusetzen	Zahlenwert falls vom eingesetzten Wert unterschiedlich
XANFR	a } b } Integrationsgrenzen	ja	-
XENDR		ja	-
RKRIT 2	$V_1 M$	{ ja nein	$2 \cdot 10^{-2}$ für RKRIT 2 = 0
RKRIT 1	$V_2 M$	{ ja nein	$2 \cdot 10^{-2}$ für RKRIT 1 = 0
KINR	Anzahl der X_i bei erster Unterteilung des Integrationsbereichs	{ ja nein	{ 180 für KINR > 180 11 für KINR < 4
NRRRM	Maximale Anzahl der Wiederholungen	{ ja nein	{ 10 für NRRRM < 1 16 für NRRRM > 16
XR(I)	$X_i - \frac{b-a}{2}$	nein	
XRX	X_i für jeweiligen Punkt	nein	
YR(I)	Y_i	nein	
LMAXR	jeweilige Anzahl der Punkte	nein	
RINTEG	Resultat der Integration	nein	

Das Programm ist auch für Integration über mehrere Variable brauchbar. Das zweite Beispiel im Anhang zeigt eine Integration über zwei Variable. Für die beiden Integrationen muss der Subroutine Name der Common Block Name und die Bezeichnung der im Hauptprogramm vorkommenden Grössen und die im Hauptprogramm benutzten Statementnummern verschieden sein. Die Unterprogramme sind nicht noch einmal angegeben. Es ist bei ihnen nur der Subroutine Name durch SINT bzw. PINT und der Common Block Name durch ITS bzw. ITP ersetzt.

ISN	SOURCE STATEMENT	
0	SIIFTC INTGRY	
1	COMMON /ITR/ LR,KINR,YR(200),XR(200), RINTEG,XER,XAR,LAR,LER,	IR 1
	1 RKRIT1,RKRIT2,IKSR(100), KSMR, NRRR,NRRRM,LMAXR,DXR	IR 2
2	DIMENSION ZR(200)	
3	NRRRM=16	
4	XANFR=0.001	
5	XFNDR=1.	
6	XMITR=(XANFR+XFNDR)/2.	IR 11
7	XAR=XANFR-XMITR	IR 12
10	XER=XFNDR-XMITR	IR 13
11	LAR=1	IR 14
12	IF(KINR.GT.180) KINR=180	IR 15
15	IF(KINR.LT.4) KINR=11	IR 16
20	LFR=KINR	IR 17
21	AKAR=KINR	IR 18
22	DXR=(XER-XAR)/(AKAR-1.)	IR 19
23	SUMR=XAR-DXR	IR 20
24	DO 601 IXR=1,KINR	IR 21
25	SUMR=SUMR+DXR	IR 22
26	601 XR(IXR)=SUMR	IR 23
30	605 DO 603 NXR=1AR,LER	IR 24
31	XR(NXR)=XMITR	IR 25
32	YP(NXR)=1./XRX	
33	603 CONTINUE	IR 101
35	CALL PRINT	IR 102
36	IF(LR.EQ.3) GO TO 605	IR 103
41	DO 11 LMN=1,LMAXR	
42	11 ZR(LMN)=XR(LMN)+XMITR	
44	WRITE(6,125) RINTEG, (ZR(N),YR(N), N=1,LMAXR)	
51	139 FORMAT(/1HC ,E12.5/(1H , 10E11.3))	
52	STOP	
53	END	

ISN	SOURCE STATEMENT	
0	SIIFTC ITGR	
1	SUBROUTINE RINT	ITR 1
2	COMMON /ITR/ LR,KINR,YR(200),XR(200), RINTEG,XER,XAR,LAR,LFR,	ITR 2
	1 RKRIT1,RKRIT2,IKSR(100), KSMR, NRRR,NRRRM,LMAXR,DXP	ITR 3
3	DIMENSION ZP(200), VP(200)	ITR 4
4	IF(LR.EQ.3) GO TO 890	ITR 5
7	851 LR=4	ITR 6
10	IF(NRRRM.LT.1) NRRRM=10	ITR 7
13	IF(NRRRM.GT.16) NRRRM=16	ITR 8
16	IF(RKRIT1.LT.1.E-10) RKRIT1=2.E-2	ITR 9
21	IF(RKRIT2.LT.1.E-10) RKRIT2=2.E-2	ITR 10
24	NRRR=0	ITR 11
25	LMAXR=KINR	ITR All
26	GO TO 877	ITR 12
27	876 LR=2	ITR 13
30	877 KSMR=LMAXR-2	ITR 14
31	KLMP=101	ITR 15
32	LKP=1	ITR 16
33	DO 852 KSP=1,KSMR	ITR 17
34	IF(ABS(YR(KSP+2)-YR(KSP)).LT.1.E-8) GO TO 862	ITR 18
37	AKRIP=(XR(KSP+1)-XR(KSP))/(YR(KSP+2)-YR(KSP))	ITR 19
	1 -YR(KSP+1)-YR(KSP))/(YR(KSP+2)-YR(KSP))	ITR 20
40	AKRIP=ABS(AKRIP)	ITR 21
41	IF(AKRIP.LT.RKRIT2) GO TO 815	ITR 22
44	862 FDP=(YR(KSP)+XR(KSP+2)-XR(KSP+1))+YR(KSP+1)*(XR(KSP)-XR(KSP+2))	ITR 23
	1+YR(KSP+2)*(XR(KSP+1)-XR(KSP))/(XER-XAR)	ITR 24
45	IF(ABS(FDP).LT.1.E-10) GO TO 815	ITR 25
50	F3P=(YR(KSP)+YR(KSP+2))*(XR(KSP+2)-XR(KSP))/(XER-XAR)	ITR 26
51	IF(ABS(F3P).LT.1.E-8) GO TO 863	ITR 27
54	AKRT1=ABS(FDP/F3P)	ITR 28
55	IF(AKRT1.LT.RKRIT1) GO TO 815	ITR 29
60	863 LR=3	ITR 30
61	GO TO (858, 859),LKP	ITR 31
62	858 XR(KLMP)=(XR(KSP+1)+XR(KSP))/2.	ITR 32
63	IKSR(KLMP-100)=KSP+1	ITR 33
64	KLMP=KLMP+1	ITR 34
65	859 XR(KLMP)=(XR(KSP+2)+XR(KSP+1))/2.	ITR 35
66	IKSR(KLMP-100)=KSP+2	ITR 36
67	KLMP=KLMP+1	ITR 37
70	LKP=2	ITR 38
71	GO TO 852	ITR 39
72	815 LKP=1	ITR 40
73	852 CONTINUE	ITR 41
75	IF(LR.EQ.4) GO TO 866	ITR 42
		ITR 43

```

100 LMAXR=LMAXR+KLMF-101 ITR 44
101 LAR=101 ITR 45
102 LER=KLMF-1 ITR 46
103 IKSR(KLMF-100)=KSMR+3 ITR 47
104 NRRR=NRRR+1 ITR 48
105 IF(LR.EQ.2) GO TO 865 ITR 49
110 RETURN ITR 50
111 890 NCKPM=LER- 99 ITR 51
112 NOPE=0 ITR 52
113 DO 800 NOKP=1,NOKPM ITR 53
114 NOKP=NOKP+1 ITR 54
115 NOKP=IRN(NOKP)-1 ITR 55
116 DO 800 NOKP=NOKP ITR 56
117 NOKP=NOKP+1 ITR 57
120 ZP( NOKP )=YR(NOKP) ITR 58
121 869 VP( NOKP )=XR(NOKP) ITR 59
123 NOKP=NOKP+NOKP ITR 60
124 ZP( NOKP )=YR(NOKP+100) ITR 61
125 VP( NOKP )=XP(NOKP+100) ITR 62
126 868 CONTINUE ITR 63
130 DO 861 NOLP=1,LMAXR ITR 64
131 XR(NOLP)=VP(NOLP) ITR 65
132 861 YP(NOLP)=ZP(NOLP) ITR 66
134 IF(NRRR.EC.NRRRM.OR.LMAXR.GT.90) GO TO 865 ITR 67
137 GO TO 876 ITR 68
140 865 RINTEG=C. ITR 69
141 NINTP=LMAXR-2 ITR 70
142 XNCP=XR-XAR ITR 71
143 DO 878 IV=1,LMAXR ITR 72
144 878 VP(IV)=XR(IV)/XNCR ITR 73
146 DO 881 NIP=1,NINTP ITR 74
147 IF (NIP.EQ.1) GO TO 882 ITR 75
152 IF (NIP.EQ.NINTP) GO TO 883 ITR 76
155 XAIP=(VP(NIP)+VP(NIP+1))/2. ITR 77
156 XEIP=(VP(NIP+1)+VP(NIP+2))/2. ITR 78
157 GO TO 884 ITR 79
160 882 XAIP=XR/XNCR ITR 80
161 XEIP=(VP(2)+VP(3))/2. ITR 81
162 GO TO 884 ITR 82
163 883 XAIP=(VP(NIP)+VP(NIP+1))/2. ITR 83
164 XEIP=XR/XNCR ITR 84
165 884 IF (ABS(XEIP-XAIP).LT.1.E-3) GO TO 886 ITR 84A
170 DO 885 MIP=1,3 ITR 85
171 LIPC=NIP+MIP-1 ITR 86
172 LIP1=NIP+MIP-3*(MIP/3) ITR 87
173 LIP2=NIP+MIP+1-3*(MIP+1)/3 ITR 88
174 DSP=YR(LIPC)/ (VP(LIP1)-VP(LIP2)) * (VP(LIP2)+ ITR 89
1VP(LIP1)*(XEIP-XAIP)-C.5*(VP(LIP1)+VP(LIP2))*(XEIP**2-XAIP**2) ITR 90
2+(XEIP**3-XAIP**3)/3.)/ (VP(LIP2) -VP(LIP1)) ITR 91
175 RINTEG=RINTEG+DSP ITR 92
176 885 CONTINUE ITR 93
200 GO TO 887 ITR 94
201 886 RINTEG=RINTEG+(XEIP-XAIP)*(YR(NIP+1)+(YR(NIP+2)-YR(NIP))/(VP(NIP+2) ITR 95
1 -VP(NIP))*(XEIP+XAIP-2.*VP(NIP+1))/2.) ITR 96
202 887 CONTINUE ITR 97
203 881 CONTINUE ITR 98
205 RINTEG=RINTEG*XNCR ITR 99
206 LR=0 ITR 100
207 RETURN ITR 101
210 866 NINTP=KINR- 3 ITR 102
211 RINTEG=(C.9*(YR(1)+YR(KINR))*2.8*(YR(2)+YR(KINR-1))+2.3*(YR(3)+ ITR 103
1 YR(KINR-2)))/2.4 ITR 104
212 GO 867 NIP=4, NINTP ITR 105
213 867 RINTEG=RINTEG+YR(NIP) ITR 106
215 RINTEG=XR*RINTEG ITR 107
216 LR=C ITR 108
217 RETURN ITR 109
220 FNC ITR 110

```

118, LUTZ, INTEGRATION R

IRLOR -- JOB 000000

12/08/65

OBJECT PROGRAM IS BEING ENTERED INTO STORAGE.

C.69033E C1

C.100E-C2	0.100E 04	0.120E-02	0.837E 03	0.139E-02	0.719E 03	0.159E-02	0.631E 03	0.178E-02	0.562E 03
C.100E-02	0.506E 03	0.217E-02	0.461E 03	0.256E-02	0.390E 03	0.295E-02	0.339E 03	0.334E-02	0.299E 03
C.506E-C2	C.268E 03	0.412E-02	0.243E 03	0.490E-02	0.204E 03	0.568E-02	0.176E 03	0.646E-02	0.155E 03
C.724E-C2	C.138E 03	0.802E-02	0.125E 03	C.880E-02	0.114E 03	0.104E-01	0.965E 02	0.119E-01	0.838E 02
C.138E-C1	0.741E 02	0.150E-01	0.665E 02	0.166E-01	0.602E 02	0.197E-01	0.507E 02	0.229E-01	0.438E 02
C.268E-01	C.385E 02	0.291E-01	0.344E 02	0.322E-01	0.310E 02	0.385E-01	0.260E 02	0.447E-01	0.224E 02
C.510E-C1	0.196E C2	0.572E-C1	0.175E C2	0.634E-01	0.158E 02	0.759E-01	0.132E 02	0.884E-01	0.113E 02
C.100E C0	0.991E 01	0.113E 00	0.882E 01	0.126E 00	0.794E 01	0.151E 00	0.663E 01	0.176E 00	0.569E 01
C.201E C0	0.498E 01	0.226E C0	0.443E 01	0.251E 00	0.399E 01	0.301E 00	0.333E 01	0.351E 00	0.285E 01
C.401E C0	0.250E C1	0.451E C0	0.222E C1	C.501E 00	0.200E 01	0.600E 00	0.167E 01	0.700E 00	0.143E 01
C.800E CC	C.125E 01	C.900E 00	0.111E 01	0.100E 01	0.100E 01				

ISN	SOURCE STATEMENT		
0	\$IBFTC INTGRT		
1	COMMON /ITP/ LP,KINP,YP(200),XP(200),PINTEG,XEP,XAP,LAP,LEP,	IP	1
2	1 PKRIT1,PKRIT2, IKSP(100),KSMP,NPRP,NPRPM,LMAXP,DXP	IP	2
	COMMON /ITS/ LS,KINS,YS(200),XS(200),SINTEG,XES,XAS,LAS,LES,	IS	1
	1 SKRIT1,SKRIT2,IKSS(100),KSHS,NSRS,NSRSM,LMAXS,DXS	IS	2
3	XANFP=0.1		
4	XENDP=10.		
5	XANFS=0.1		
6	XENDS=10.		
7	XMITP=(XANFP+XENDP)/2.	IP	11
10	XAP=XANFP -XMITP	IP	12
11	XEP=XENDP-XMITP	IP	13
12	LAP=1	IP	14
13	IF(KINP.GT.180) KINP=180	IP	15
16	IF(KINP.LT.4) KINP=11	IP	16
21	LEP=KINP	IP	17
22	AKNP=KINP	IP	18
23	DXP=(XEP-XAP)/(AKNP-1.)	IP	19
24	SUMP=XAP-DXP	IP	20
25	DO 801 IXP=1,KINP	IP	21
26	SUMP=SUMP+DXP	IP	22
27	801 XP(IXP)=SUMP	IP	23
31	805 DO 803 NXP=LAP,LEP	IP	24
32	XPX=XP(NXP)+XMITP	IP	25
33	X=XPX		
34	XMITS=(XANFS+XENDS)/2.	IS	11
35	XAS=XANFS-XMITS	IS	12
36	XES=XENDS-XMITS	IS	13
37	LAS=1	IS	14
40	IF(KINS.LT.4) KINS=11	IS	15
43	IF(KINS.GT.180) KINS=180	IS	16
46	LES=KINS	IS	17
47	AKNS=KINS	IS	18
50	DXS=(XES-XAS)/(AKNS-1.)	IS	19
51	SUMS=XAS-DXS	IS	20
52	DO 701 IXS=1,KINS	IS	21
53	SUMS=SUMS+DXS	IS	22
54	701 XS(IXS)=SUMS	IS	23
56	705 DO 703 NXS=LAS,LES	IS	24
57	XSX=XS(NXS)+XMITS	IS	25
60	Y=XSX		
61	Z=1./X/Y		
62	YS(NXS)=Z		
63	703 CONTINUE	IS	101
65	CALL SINT	IS	102
66	IF(LS.EQ.3) GO TO 705	IS	103
71	YP(NXP)=SINTEG		
72	803 CONTINUE	IP	101
74	CALL PINT	IP	102
75	IF(LP.EQ.3) GO TO 805	IP	103
100	WRITE(6,10) XANFP, XENDP, PINTEG		
101	10 FORMAT (7H XANFP= ,F10.5, 7H XENDP= , F10.5, 3X, 7HPINTEG=,E15.5)		
102	STOP		
103	END		