User's Manual for the DLL version of ORGANON 9.1

The ORGRUN.DLL and ORGRUN.LIB Microsoft compatible import files were created using the Lahey/Fujitsu FORTRAN 95 compiler. Two subroutines have been exposed for usage by other programs: EXECUTE and GET_ORGRUN_EDITION

```
SUBROUTINE EXECUTE(CYCLG,VERSION,NPTS,NTREES1,STAGE,BHAGE,TREENO,
                   PTNO,SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,
                   EXPAN1,MGEXP,RVARS,ACALIB,PN,YSF,BABT,BART,
                   YST,NPR,PRAGE,PRLH,PRDBH,PRHT,PRCR,PREXP,
                   BRCNT,BRHT,BRDIA,JCORE,SERROR,TERROR,SWARNING,
                   TWARNING,IERROR,DGRO,HGRO,CRCHNG,SCRCHNG,
                   MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,STOR)

DIMENSIONS TREENO(2000),PTNO(2000),SPECIES(2000),USER(2000),
           INDS(30),DBH1(2000),HT1(2000),CR1(2000),SCR1(2000),
           EXPAN1(2000),MGEXP(2000),RVARS(30),ACALIB(3,18),
           PN(5),YSF(5),BART(5),YST(5),NPR(2000),PRAGE(2000,3),
           PRLH(2000,3),PRDBH(2000,3),PRHT(2000,3),PRCR(2000,3),
           PREXP(2000,3),BRCNT(2000,3),BRHT(2000,40),
           BRDIA(2000,40),JCORE(2000,40),SERROR(35),
           TERROR(2000,6),SWARNING(9),TWARNING(2000),DGRO(2000),
           HGRO(2000),CRCHNG(2000),SCRCHNG(2000),MORTEXP(2000),
           NTREES2,DBH2(2000),HT2(2000),CR2(2000),SCR2(2000),
           EXPAN2(2000),STOR(30)
```

The following variables will include a classification describing whether each variable is strictly an "INPUT" variable (i.e., it is entered into the DLL and is not modified by the DLL), strictly an "OUTPUT" variable (i.e., it is created with in the DLL and then outputted by the DLL), or a combination "INPUT/OUTPUT" variable (i.e., it is entered into the DLL, modified by the DLL, and the modified variable is outputted by the DLL).

Description of Variables

CYCLG	INTEGER*4	Total number of five-year growth cycles previously grown in ORGANON. (INPUT/OUTPUT variable)
VERSION	INTEGER*4	Version of ORGANON to be used: 1 = Southwest Oregon (SWO), 2 = Northwest Oregon (NWO), 3 = Stand Management Cooperative (SMC), 4 = Red Alder Plantation (RAP). (INPUT variable)

NPTS	INTEGER*4	Total number of sample plots/points used to collect the tree list data. Include all treeless plots/points in the count.(INPUT variable)
NTREES1	INTEGER*4	Total number of sample trees measured in the stand at the start of the growth period (NTREES1 cannot exceed 2000). The total number of sample trees in the stand would increase over the run if tripling or ingrowth is used. (INPUT/OUTPUT variable)
STAGE	INTEGER*4	Total stand age at the start of the current growth cycle(s). ORGANON will update this value during projections. STAGE should be 0 for an uneven-aged stand. (INPUT/OUTPUT variable)
BHAGE	INTEGER*4	Breast height stand age at the start of the current growth cycle(s). ORGANON will update this value during projections. BHAGE should be 0 for an uneven-aged stand. (INPUT/OUTPUT variable)
TREENO(I)	INTEGER*4	Tree number for the I^{th} sample tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period. (INPUT variable)
PTNO(I)	INTEGER*4	Point number for the I^{th} sample tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period. (INPUT variable)
SPECIES(I)	INTEGER*4	Species code for the I^{th} sample tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period. (INPUT variable)

USER(I)	INTEGER*4	User code for the I^{th} sample tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period. (INPUT variable)
INDS(1)	INTEGER*4	1 = Use height/DBH calibration factors for the projection; 0 = Do not use height/DBH calibration factors for the projection. (INPUT variable)
INDS(2)	INTEGER*4	1 = Use crown ratio calibration factors for the projection; 0 = Do not use crown ratio calibration factors for the projection. (INPUT variable)
INDS(3)	INTEGER*4	1 = Use diameter growth rate calibration factors for the projection; 0 = Do not use diameter growth rate ratio calibration factors for the projection. (INPUT variable)
INDS(4)	INTEGER*4	1 = Stand is even-aged; 0 = Stand is uneven-aged. (INPUT variable)
INDS(5)	INTEGER*4	1 = Triple the sample tree list during projection (if possible); 0 = Do not triple the sample tree list during projection. Tripling will continue until a maximum of 2000 sample trees is achieved. NTREES2 will be larger than NTREES1 if tripling is selected and if it can be applied to the run (because NTREES1 is small enough).(INPUT variable)
INDS(6)	INTEGER*4	1 = Stand has been pruned either at the start of this growth cycle(s) or in the past; 0 = Stand has never been pruned. (INPUT variable)
INDS(7)	INTEGER*4	1 = Stand has been partially cut either at the start of this growth cycle(s) or in the past; 0 = Stand has never been partially cut. (INPUT variable)

INDS(8)	INTEGER*4	1 = Stand has been fertilized either at the start of this growth cycle(s) or in the past; 0 = Stand has never been fertilized. (INPUT variable)
INDS(9)	INTEGER*4	1 = Use limit of maximum SDI; 0 = Do not use limit of maximum SDI. (INPUT variable)
INDS(10)	INTEGER*4	1 = Wood quality variables are being computed; 0 = Wood quality variables are not being computed. (INPUT variable)
INDS(11)	INTEGER*4	1 = Overstory trees were removed at the start of the current growth cycle; 0 = Overstory trees were not removed at the start of the current growth cycle. (INPUT variable)
INDS(12)	INTEGER*4	1 = Ingrowth was added at the start of the current growth cycle; 0 = Ingrowth was not added at the start of the current growth cycle. (INPUT variable)
INDS(13)	INTEGER*4	1 = Major conifer trees were cut at the start of the current growth cycle; 0 = Major conifer trees were not cut at the start of the current growth cycle. For the southwest Oregon version of ORGANON, major conifer species are Douglas-fir, white fir, grand fir, ponderosa pine, sugar pine, and incense-cedar. For the northwest Oregon and Stand Management Cooperative versions of ORGANON, major conifer species are Douglas-fir, grand fir, and western hemlock. (INPUT variable)
INDS(14)	INTEGER*4	1 = The even-aged stand has been planted with genetically improved Douglas-fir; 0 = The even-aged stand was not planted with genetically improved Douglas-fir. (INPUT variable)

INDS(15)	INTEGER*4	1 = The Douglas-fir in the stand have been infected with Swiss needle cast; 0 = The Douglas-fir in the stand have not been infected with Swiss needle cast. Only applicable to the NWO and SMC versions of ORGANON. (INPUT variable)
INDS(16+)	INTEGER*4	Currently not used, set to 0. (INPUT variable)
DBH1(I)	REAL*4	DBH for the I^{th} sample tree at the start of the growth period, $1 \leq I \leq$ NTREES1 (maximum of 2000). (INPUT variable)
HT1(I)	REAL*4	Total height for the I^{th} sample tree at the start of the growth period, $1 \leq I \leq$ NTREES1 (maximum of 2000). (INPUT variable)
CR1(I)	REAL*4	Crown ratio for the I^{th} sample tree at the start of the growth period, $1 \leq I \leq$ NTREES1 (maximum of 2000) (INPUT variable).
SCR1(I)	REAL*4	Shadow crown ratio for the I^{th} sample tree at the start of the growth period, $1 \leq I \leq$ NTREES1 (maximum of 2000). Shadow crown ratio is used for pruned trees and it is the crown ratio of the tree if it had not been pruned. ORGANON will update shadow crown ratios. Therefore, it is strongly recommended that stands which had been pruned before the measurement of the input tree listing not be projected in ORGANON. For unpruned stands, set SCR1 to zero. For the first pruning conducted in ORGANON, set SCR1 for each pruned tree to the CR before pruning. For a subsequent pruning in ORGANON, set SCR1 for each newly pruned tree to either the CR before the subsequent pruning or to the previous SCR value, whichever is larger. (INPUT variable)

EXPAN1(I)	REAL*4	The plot/point level expansion factor for the I^{th} sample tree at the start of the growth period (i.e., the expansion factors should NOT be divided by the total number of plots/points measured in the stand), $1 \leq I \leq$ NTREES1 (maximum of 2000). (INPUT variable)
MGEXP(I)	REAL*4	The plot/point level expansion factor for the I^{th} sample tree at the start of the growth period that was removed by cutting just prior to the start of the current growth period (i.e., do not include trees cut at the start of previous growth periods), $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period. Again, the expansion factors should NOT be divided by the total number of plots/points measured in the stand. (INPUT variable)
RVARS(1)	REAL*4	The value of SITE_1: Douglas-fir site index: Hann and Scriviani (1987) for the SWO version, and Bruce (1981) for the NWO and SMC. Red alder site index: Weiskittel et al. (2009) for the RAP version. For the SWO, NWO, and SMC versions, ORGANON will calculate this value from SITE_2 if it is set to zero. For the RAP version, SITE_1 must be entered. (INPUT variable)
RVARS(2)	REAL*4	Other site index (SITE_2): Hann and Scriviani (1987) ponderosa pine site index for the SWO version and Flewelling's site index for western hemlock in the NWO and SMC versions. For the RAP version, Bruce (1981) Douglas-fir site index. For the SWO, NWO, and SMC versions, ORGANON will calculate this value from SITE_1 if it is set to zero. For the RAP version, it will be reset to 115 if not entered by the user. (INPUT variable)

RVARS(3)	REAL*4	Maximum stand density index of Douglas-fir for the SWO, NWO, and SMC versions. Maximum stand density index of red alder for the RAP version. A value of 0 will result in the version specific default value being used. (INPUT variable)
RVARS(4)	REAL*4	Maximum stand density index of white fir and/or grand fir for the SWO, NWO, and SMC versions. Maximum stand density index of Douglas-fir for the RAP version. A value of 0 will result in the version specific default value being used. (INPUT variable)
RVARS(5)	REAL*4	Maximum stand density index of ponderosa pine for the SWO version and western hemlock in the NWO, SMC, and RAP versions. A value of 0 will result in the version specific default value being used. (INPUT variable)
RVARS(6)	REAL*4	Douglas-fir genetic worth value for diameter growth rate. Only applicable to even-aged stands. (INPUT variable)
RVARS(7)	REAL*4	Douglas-fir genetic worth value for height growth rate. Only applicable to even-aged stands. (INPUT variable)
RVARS(8)	REAL*4	Douglas-fir foliage retention as a result of infection by Swiss needle cast. Only applicable to the NWO and SMC versions of ORGANON. (INPUT variable)
RVARS(9)	REAL*4	For RAP-ORGANON, the starting number of red alder trees per acre (i.e., planting density) for the plantation. Must be specified for RAP-ORGANON only, otherwise it is set to zero. (INPUT variable)
RVARS(10+)	REAL*4	Currently unused, set to 0. (INPUT variable)

ACALIB(I,J)	REAL*4	Actual calibration values for the I^{th} attribute (when I=1, the height/DBH equation; I=2, the crown ratio equation; and I=3, the diameter growth rate equation); and the J_{th} species group depending upon the version of ORGANON, $1 \leq J \leq 18$. Species groups are calculated in the DLL based upon version and species for the tree. (INPUT variable)
PN(I)	REAL*4	Number of pounds of nitrogen applied per acre at the I^{th} application, $1 \leq I \leq 5$. I=1 for the most recent application, I=2 for the next most recent application, etc. PN must be ≤ 400. The whole array should be set to 0 if no fertilization has occurred. As an example, a stand was fertilized with 100 lbs. of nitrogen 7 years before the stand was measured, a second fertilization of 200 lbs. of nitrogen was conducted at the start of the run, and a third fertilization of 300 lbs. of nitrogen was conducted after two growth cycles: PN(1)=300.0, PN(2)=200.0, PN(3)=100.0, PN(4)=0.0, PN(5)=0.0. (INPUT variable)

YSF(I)	REAL*4	Number of years since start of the run that the I^{th} application of nitrogen fertilizer was applied, $1 \leq I \leq 5$. I=1 for the most recent application, I=2 for the next most recent application, etc. The whole array should be set to 0.0 if no fertilization has occurred. For fertilizations conducted before the stand was measured, YSF(I)=-(number of years from the I^{th} fertilization to when the stand was measured). For fertilizations conducted in ORGANON, YSF(I)=5(the value of CYCLG at the time of fertilization). As an example, a stand was fertilized 7 years before the stand was measured, a second time at the start of the run, and a third time after two growth cycles: YSF(1)=10.0 (i.e., 5x2), YSF(2)=0.0 (i.e., 5x0), YSF(3)=-7.0, YSF(4)=0.0, YSF(5)=0.0. (INPUT variable)
BABT	REAL*4	Basal area per acre of the stand just before the most recent removal of trees. BABT should be set to 0.0 if no tree removal has occurred. (INPUT variable)
BART(I)	REAL*4	Basal area per acre cut at the I^{th} removal of trees, $1 \leq I \leq 5$. I=1 for the most recent removal, I=2 for the next most recent removal, etc. The whole array should be set to 0 if no tree removal has occurred. As an example, a stand had 50.0 ft^2 removed 7 years before the stand was measured, a second thinning removed 100.0 ft^2 at the start of the run, and a third thinned removed 150.0 ft^2 after two growth cycles: BART(1)=150.0, BART(2)=100.0, BART(3)=50.0, BART(4)=0.0, BART(5)=0.0. (INPUT variable)

YST(I)	REAL*4	Number of years since the start of the run that the I^{th} removal of trees occurred, $1 \leq I \leq 5$. I=1 for the most recent removal, I=2 for the next most recent removal, etc. The whole array should be set to 0.0 if no thinning has occurred. For thinnings conducted before the stand was measured, YST(I)=-(number of years from the I^{th} thinning to when the stand was measured). For thinnings conducted in ORGANON, YST(I)=5(the value of CYCLG at the time of thinning). As an example, a stand was thinned 7 years before the stand was measured, a second time at the start of the run, and a third time after two growth cycles: YST(1)=10.0 (i.e., 5x2), YST(2)=0.0 (i.e., 5x0), YST(3)=-7.0, YST(4)=0.0, YST(5)=0.0. (INPUT variable)
NPR(I)	INTEGER*4	Number of prunings conducted on the I^{th} tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period. A maximum of three prunings are allowed on a tree. (INPUT variable)
PRAGE(I,J)	INTEGER*4	Age of the I^{th} tree when the J^{th} pruning was conducted on the tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)
PRLH(I,J)	REAL*4	Lift height, in feet, for the I^{th} tree when the J^{th} pruning was conducted on the tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)

PRDBH(I,J)	REAL*4	DBH of the I^{th} tree when the J^{th} pruning was conducted on the tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)
PRHT(I,J)	REAL*4	Total height of the I^{th} tree when the J^{th} pruning was conducted on the tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)
PRCR(I,J)	REAL*4	Crown ratio of the I^{th} tree when the J^{th} pruning was conducted on the tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)
PREXP(I,J)	REAL*4	Plot/point level expansion factor of the I^{th} tree when the J^{th} pruning was conducted on the tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)
BRCNT(I,J)	INTEGER*4	Wood quality branch count of type J for the I^{th} tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 3$. (INPUT variable)
BRHT(I,J)	INTEGER*4	Height to the J^{th} branch on the I^{th} tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 40$. (INPUT variable)

BRDIA(I,J)	INTEGER*4	Branch diameter of the J^{th} branch on the I^{th} tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 40$. (INPUT variable)
JCORE(I,J)	INTEGER*4	Diameter of the juvenile wood core at the J^{th} branch on the I^{th} tree, $1 \leq I \leq$ NTREES1 (maximum of 2000) at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period, $1 \leq J \leq 40$. (INPUT variable)
SERROR(I)	INTEGER*4	If SERROR(I)=1 ($1 \leq I \leq 35$), then a stand level error of type "I" has occurred (a value of 0 indicated no error). See the following tables for a description of the particular errors. (OUTPUT variable)
TERROR(I,J)	INTEGER*4	If TERROR(I,J)=1 ($1 \leq I \leq$ NTREES1, $1 \leq J \leq 6$), then a tree level error of type "J" has occurred for the I^{th} tree (a value of 0 indicated no error). See the following tables for a description of the particular errors. (OUTPUT variable)
SWARNING(I)	INTEGER*4	If SWARNING(I)=1 ($1 \leq I \leq 9$), then a stand level warning of type "I" has occurred (a value of 0 indicated no error). See the following tables for a description of the particular warnings. (OUTPUT variable)
TWARNING(I)	INTEGER*4	If TWARNING(I)=1 ($1 \leq I \leq$ NTREES1), then a tree warning has occurred for the I^{th} tree (a value of 0 indicated no error). See the following tables for a description of the particular warnings. (OUTPUT variable)

IERROR	INTEGER*4	If IERROR=1, then a stand or tree level error has occurred and the error must be corrected before proceeding. (OUTPUT variable)
DGRO(I)	REAL*4	The 5-year diameter growth rate for the I^{th} sample tree, $1 \leq I \leq$ NTREES2 (maximum of 2000). (OUTPUT variable)
HGRO(I)	REAL*4	The 5-year height growth rate for the I^{th} sample tree, $1 \leq I \leq$ NTREES2 (maximum of 2000). (OUTPUT variable)
CRCHNG(I)	REAL*4	The 5-year change in crown ratio for the I^{th} sample, $1 \leq I \leq$ NTREES2 (maximum of 2000). (OUTPUT variable)
SCRCHNG(I)	REAL*4	The 5-year change in the shadow crown ratio for the I^{th} sample tree, $1 \leq I \leq$ NTREES2 (maximum of 2000). (OUTPUT variable)
MORTEXP(I)	REAL*4	The plot/point level expansion factor for 5-year mortality on the I^{th} sample tree, $1 \leq I \leq$ NTREES1 at the start of the growth period and $1 \leq I \leq$ NTREES2 at the end of the growth period (maximum of 2000). The expansion factors have NOT been divided by the total number of plots/points measured in the stand (OUTPUT variable)
NTREES2	INTEGER*4	Total number of sample trees measured in the stand at the end of the growth period (NTREES2 cannot exceed 2000). NTREES2 would be larger than NTREES1 if tripling or ingrowth are used, otherwise it is the same as NTREES1. (INPUT/OUTPUT variable)
DBH2(I)	REAL*4	DBH for the I^{th} sample tree at the end of the growth period, $1 \leq I \leq$ NTREES2 (maximum of 2000). (INPUT variable)

HT2(I)	REAL*4	Total height for the I^{th} sample tree at the end of the growth period, 1 ≤ I ≤ NTREES2 (maximum of 2000). (INPUT variable)
CR2(I)	REAL*4	Crown ratio for the I^{th} sample tree at the end of the growth period, 1 ≤ I ≤ NTREES2 (maximum of 2000) (INPUT variable).
SCR2(I)	REAL*4	Shadow crown ratio for the I^{th} sample tree at the end of the growth period, 1 ≤ I ≤ NTREES2 (maximum of 2000). See the description of SCR1 for more details. (INPUT variable)
EXPAN2(I)	REAL*4	The plot/point level expansion factor for the I^{th} sample tree at the end of the growth period (i.e., the expansion factors should NOT be divided by the total number of plots/points measured in the stand), 1 ≤ I ≤ NTREES2 (maximum of 2000). (INPUT variable)
STOR(I)	REAL*4	An array of 30 internal variables used by ORGANON which must not change over multiple calls of the ORGANON DLL. Initialize to zero before the first call to the ORGANON DLL. ORGANON will then calculate these values when CYCLG=0. The resulting values should not be modified by the DLL user in subsequent continued projections of the tree list. (OUTPUT variable)

SUBROUTINE GET_ORGRUN_EDITION(EDITION)

Description of Variable

EDITION	REAL*4	Edition of the ORGRUN DLL (OUTPUT variable)

Descriptions of the SERROR(I) Array

I	Description of the Error
1	NTREES < 1 or NTREES > 2000
2	VERSION < 1 or VERSION > 4
3	NPTS < 1
4	Both SITE_1 and SITE_2 are set to 0
5	There are no major tree species for the VERSION
6	BHAGE has been set to 0 for an uneven-aged stand
7	BHAGE > 0 for an uneven-aged stand
8	STAGE is too small for the BHAGE
9	An uneven-aged stand cannot be fertilized
10	YSF and/or PN variables are not zero for an unfertilized stand
11	The implied stand age of fertilization (based on YSF) must be less than or equal to current stand age or less than or equal to 70 years.
12	PN < 0 or PN > 400 lbs per acre.
13	BART(1) ≥ BABT
14	YST and/or BART variables are not zero for an uncut stand
15	For an even-aged stand, the implied stand age of cutting (based on YST) must be less than or equal to current stand age.
16	For multiple cuttings in which YST ≠ 0, BART ≤ 0
17	BABT < 0 for a stand with cuttings
18	Some MGEXP values must be > 0 in a stand that has been cut at the start of the growth period
19	CYCLG < 0
20	ACALIB < 0.5 or ACALIB > 2.0
21	MSDI_1, MSDI_2, and/or MSDI_3 > 1000
22	Stand not even-aged so genetic gain cannot be applied
23	A genetic worth value cannot be < 0%

24	A genetic worth value must be ≤ 20%
25	A genetic worth value is > 0% when no genetic gain is indicated
26	Swiss needle cast cannot be applied to this version of ORGANON
27	Swiss needle cast cannot be applied to an unevenaged stand
28	Foliage retention cannot be < 0.85
29	Foliage retention cannot be > 7.0
30	Fertilization cannot be applied to a stand with foliage retention < 3.0
31	Foliage retention is ≥ 0.85 when no Swiss needle cast impact is indicated
32	SITE_1 is set to 0 for RAP-ORGANON
33	PDEN is set to zero for RAP-ORGANON
34	Stand must be even-aged for RAP-ORGANON
35	Stand must have at least 90% of basal area in red alder for RAP-ORGANON

Descriptions of the TERROR(I,J) Array

J	Description of the Error
1	Illegal species code for the VERSION
2	DBH ≤ 0.0
3	HT ≤ 4.5
4	CR ≤ 0.0 or CR > 1.0
5	EXPAN < 0.0
6	SCR < 0.0 or SCR > 1.0

Descriptions of the SWARNING(I) Array

I	Description of the Error
1	SITE_1 is out of range for the VERSION
2	SITE_2 is out of range for the VERSION
3	Tree heights are too large for the site index value
4	BHAGE is too young for the VERSION
5	Amount of minor species is higher than recommended for the VERSION
6	Number of sample trees is below recommended minimum
7	Majority of the input stand is over the upper age recommended for the VERSION
8	Majority of the projected stand is now over the upper age recommended for the VERSION
9	Number of cycles to be projected will make the resulting stand older than that recommended for the VERSION.

Descriptions of the TWARNING(I) Array

J	Description of the Error
1	HT to DBH ratio is too large for the species

Examples of Using the ORGRUN.DLL in LAHEY FORTRAN

The following three examples demonstrate how the EXECUTE subroutine in the ORGRUN.DLL can be used to conduct various management activities using the ORGRUN.DLL. These example runs read data from and all ready created ORGANON INP file which does not contain tree number data. Therefore, temporary tree numbers are created for the examples (of course data can be entered in other ways and "real" tree numbers can be used. The following is a brief description of each example:

1. The stand is grown for one five-year growth cycle; the stand is fertilized with 200 pounds per acre of nitrogen; the stand is grown for two five-year growth cycles; the stand is fertilized again with 200 pounds of nitrogen per acre; and then the stand is grown for one more five-year growth cycle.

2. The stand is grown for one five-year growth cycle; all trees with a DBH greater than 10.0-inches are removed; the stand is grown for two five-year growth cycles; all hardwood trees are removed; and then the stand is grown for one more five-year growth cycle.

3. The stand is grown for one five-year growth cycle; all Douglas-fir trees are pruned to a crown ratio of 0.5 or a maximum lift height of 16.0-feet, whichever is lower;, the stand is grown for two five-year growth cycles; all Douglas-fir trees are pruned to a crown ratio of 0.3 or a maximum lift height of 32.0-feet, whichever is lower; and then the stand is grown for one more five-year growth cycle.

Example #1

```
C     THIS VERSION OF TESTDLL IS AN EXAMPLE OF APPLYING FERTILIZER USING
C     THE EXECUTE SUBROUTINE IN THE ORGRUN.DLL
C
      PROGRAM TESTDLL
C
      IMPLICIT NONE
C
C
      INTEGER*4  LU10,LU20,L,LEN,IANS,IYN,I,J
      INTEGER*4  NPTS,NTREES1,STAGE,BHAGE,BIG6,OTHER,MISS,NEST,VERSION,
     1           EDITION,FTYPE(5),TDATAI(2000,3),PTNO(2000)
      INTEGER*4  CYCLG,ISTAGE,IBHAGE,TREENO(2000),SPECIES(2000),
     1           USER(2000),NPR(2000),PRAGE(2000,3),BRCNT(2000,3),
     2           BRHT(2000,40),BRDIA(2000,40),JCORE(2000,40),SERROR(35),
     3           TERROR(2000,6),SWARNING(9),TWARNING(2000)
      INTEGER*4  INDS(30),IERROR,NTREES2
      REAL*4     SITE_1,SITE_2,MSDI_1,MSDI_2,MSDI_3,ACALIB(3,18),BAF(5),
     1           VALUE(5),TDATAR(2000,4),PGROWTH(2000)
      REAL*4     DBH1(2000),HT1(2000),CR1(2000),EXPAN1(2000),SCR1(2000),
     1           PN(5),YSF(5),BABT,BART(5),YST(5),PRLH(2000,3),
     2           PRDBH(2000,3),PRHT(2000,3),PRCR(2000,3),PREXP(2000,3),
     3           OLD,MGEXP(2000),DGRO(2000),HGRO(2000),CRCHNG(2000),
     4           SCRCHNG(2000),MORTEXP(2000),STOR(30), DBH2(2000),
     5           HT2(2000),CR2(2000),EXPAN2(2000),SCR2(2000),RVARS(30)
      CHARACTER  ANS1*1
      CHARACTER  DNAME*128,SNAME*128,ONAME*128
      CHARACTER  TITLE*20,TYPE*1(5)
      LOGICAL    HERE
      LOGICAL*2  EVEN,RAD
C
C     IMPORT THE EXECUTE SUBROUTINE FOR THE ORGRUN.DLL
C
      DLL_IMPORT EXECUTE
C
C     IMPORT THE READDATA SUBROUTINE FROM THE READINP.DLL FOR
C     READING ORGANON .INP FILES
C
      DLL_IMPORT READDATA
C
C
      LU10=10
      LU20=20
C
C     DETERMINE THE NAME OF THE ORGANON CREATED .INP FILE TO OPEN
C
   10 WRITE(*,1000)
 1000 FORMAT(' Enter tree file name [ NO extension (i.e., ',
     1       'C:MYFILE) ]',/' ---> '\)
      READ(*,'(A128)') DNAME
      L = ICHAR(DNAME(1:1))
      IF ( L .EQ. 17 .OR. L .EQ. 27 ) GOTO 99
      LEN = LEN_TRIM(DNAME)
      IF ( LEN .GT. 124 ) LEN = 124
      SNAME=DNAME(1:LEN)//'.INP'
      INQUIRE(FILE=SNAME,EXIST=HERE)
      IF(HERE) THEN
         CLOSE(LU10)
      ELSE
         WRITE(*,1100)DNAME
 1100    FORMAT(
```

```
     1   ' FILE DOES NOT EXIST!'/1X,A124/' Either re-enter',
     2   ' a data file name (check DRIVE) or enter <CTRL-Q>'/
     3   ' to restart ORGANON to enter or edit a new data file.')
         L=65
         GO TO 10
       ENDIF
C
C     OPEN OUTPUT FILE
C
   20 WRITE(*,1200)
 1200 FORMAT(1X,'Enter output file name ---> '\)
   30 READ(*,'(A128)')ONAME
      IF(ONAME .EQ. ' ')THEN
           WRITE(*,1300)
 1300      FORMAT(14X,'*** ERROR--REENTER ---> '\)
           GO TO 30
      ENDIF
      LEN = LEN_TRIM(ONAME) - 1
      DO I = 1,LEN
        IF(ONAME(I:I) .EQ. ' ' .AND. ONAME(I+1:I+1) .NE. ' ') THEN
          WRITE(*,1400)
 1400     FORMAT(14X,'INVALID FILE NAME - REENTER ---> '\)
          GO TO 30
        ENDIF
      ENDDO
      INQUIRE(FILE=ONAME,EXIST=HERE)
      IF(HERE)THEN
        IF(LEN .GT. 64) THEN
           WRITE(*,1500)ONAME
 1500      FORMAT(1X,A128/' FILE EXISTS! -- OVERWRITE? (Y) --->'\)
        ELSE
           WRITE(*,1600)ONAME
 1600   FORMAT('FILE "',A64,'" EXISTS!'/,' OVERWRITE? (Y) --->'\)
        ENDIF
   40   IANS = IYN(1)
        IF(IANS .EQ. 2)GO TO 20
        IF(IANS .EQ. 0)THEN
          WRITE(*,1300)
          GO TO 40
        ENDIF
      ENDIF
      OPEN(LU20,FILE=ONAME,STATUS='UNKNOWN')
C
C     READ TREE DATA FROM THE ORGANON CREATED .INP FILE USING THE
C     READDATA SUBROUTINE FROM THE READINP.DLL
C
      CALL READDATA(SNAME,TITLE,NPTS,NTREES1,SITE_1,SITE_2,EVEN,RAD,
     1     STAGE,BHAGE,BIG6,OTHER,MISS,NEST,VERSION,EDITION,
     2     MSDI_1,MSDI_2,MSDI_3,ACALIB,TYPE,FTYPE,BAF,VALUE,TDATAI,
     3     TDATAR,PGROWTH,PTNO)
C
C     TRANSFER DATA TO THE EXECUTE SUBROUTINE VARIABLES AND ARRAYS
C
      DO I=1,30
         INDS(I)=0
      ENDDO
      ISTAGE=STAGE
      IBHAGE=BHAGE
      IF(EVEN) THEN
        INDS(4)=1
      ENDIF
      DO I=1,NTREES1
        TREENO(I)=I                          !  TEMPORARY TREE NUMBERS
```

```
        SPECIES(I)=TDATAI(I,1)
        USER(I)=TDATAI(I,3)
        DBH1(I)=TDATAR(I,1)
        HT1(I)=TDATAR(I,2)
        CR1(I)=TDATAR(I,3)
        EXPAN1(I)=TDATAR(I,4)
      ENDDO
C
C     INITIALIZE SELECTED VARIABLES TO ZERO BEFORE RUNNING EXECUTE DLL
C
      CYCLG=0
      DO I=1,NTREES1
        SCR1(I)=0.0
        MGEXP(I)=0.0
        DO J=1,3
           BRCNT(I,J)=0
        ENDDO
        DO J=1,40
           BRHT(I,J)=0
           BRDIA(I,J)=0
           JCORE(I,J)=0
        ENDDO
      ENDDO
      BABT=0.0
      DO I=1,5
        PN(I)=0.0
        YSF(I)=0.0
        BART(I)=0.0
        YST(I)=0.0
      ENDDO
      RVARS(1)=SITE_1
      RVARS(2)=SITE_2
      RVARS(3)=MSDI_1
      RVARS(4)=MSDI_2
      RVARS(5)=MSDI_3
C
C     IN THE FOLLOWING EXAMPLE, THE STAND IS GROWN ONE CYCLE, FERTILIZED,
C     GROWN FOR TWO CYCLES, FERTILIZED AGAIN, AND GROWN FOR AN ADDITIONAL
C     CYCLE
C
      INDS(9)=1                              !  USE LIMIT ON MAXIMUM SDI
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C     ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C     FOR THE NEXT CALL TO EXECUTE
C
      DO I=1,NTREES2
        DBH1(I)=DBH2(I)
        HT1(I)=HT2(I)
        CR1(I)=CR2(I)
        SCR1(I)=SCR2(I)
        EXPAN1(I)=EXPAN2(I)
      ENDDO
      NTREES1=NTREES2
C
C     SET FERTILIZATION VARIABLES
C
```

```
      INDS(8)=1                                  !  FERTILIZTION INDICATOR
      YSF(1)=5.0*FLOAT(CYCLG)
      PN(1)=200.0                                !  FERTILIZE WITH 200 LBS N
      NUMCY=2                                    !  GROW TWO CYCLES
      DO II=1,2
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C        ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C        FOR THE NEXT CALL TO EXECUTE
C
         DO I=1,NTREES2
           DBH1(I)=DBH2(I)
           HT1(I)=HT2(I)
           CR1(I)=CR2(I)
           SCR1(I)=SCR2(I)
           EXPAN1(I)=EXPAN2(I)
         ENDDO
         NTREES1=NTREES2
      ENDDO
C
C     SET FERTILIZATION VARIABLES
C
C     MOVE THE VARIABLES FOR THE FIRST FERTILIZATION INTO THE SECOND
C     POSITION
C
      DO I=5,2,-1
         YSF(I)=YSF(I-1)
         PN(I)=PN(I-1)
      ENDDO
      YSF(1)=5.0*FLOAT(CYCLG)
      PN(1)=200.0                                !  FERTILIZE WITH 200 LBS N
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C     ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C     FOR THE ENDING VALUES
C
      DO I=1,NTREES2
        DBH1(I)=DBH2(I)
        HT1(I)=HT2(I)
        CR1(I)=CR2(I)
        SCR1(I)=SCR2(I)
        EXPAN1(I)=EXPAN2(I)
      ENDDO
      NTREES1=NTREES2
C
C     OUTPUT RESULTS FROM THE RUN
C
      IF(IERROR .EQ. 1) THEN
C
C        OUTPUTS INFORMATION ABOUT ERRORS IF THEY OCCURED
C
```

```
         WRITE(LU20,1700)
 1700    FORMAT(' STAND LEVEL ERRORS')
         DO I=1,34
            WRITE(LU20, 1750) I,SERROR(I)
 1750       FORMAT(I2,', ',I1)
         ENDDO
         WRITE(LU20,1800)
 1800    FORMAT(' STAND LEVEL WARNINGS')
         DO I=1,9
            WRITE(LU20, 1850) I,SWARNING(I)
 1850       FORMAT(I2,', ',I1)
         ENDDO
         WRITE(LU20,1900)
 1900    FORMAT(' TREE LEVEL ERRORS AND A WARNING')
         DO I=1,NTREES1
            WRITE(LU20,1950) I,TERROR(I,1),TERROR(I,2),TERROR(I,3),
     1                       TERROR(I,4),TERROR(I,5),TERROR(I,6),
     2                       TWARNING(I)
 1950       FORMAT(I4,', ',6(I1,', '),I1)
         ENDDO
      ELSE
C
C        OUTPUTS TREE INFORMATION TO THE TREE LIST FILE
C
         DO I = 1,NTREES1
            IF(EXPAN1(I) .GT. 0.000000001) THEN
               WRITE(LU20,2000) CYCLG,PTNO(I),I,SPECIES(I),USER(I),
     1                          DBH1(I),HT1(I),CR1(I),EXPAN(1I),MORTEXP(I),
     2                          DGRO(I),HGRO(I)
 2000          FORMAT(I4,', ',I3,', ',I4,', ',I3,', ',I3,', ',F6.1,', ',
     1                F6.1,', ',F5.2,', ',F8.2,', ',F8.2,', ',F8.2,', ',
     2                F8.2)
            ENDIF
         ENDDO
      ENDIF
   99 CLOSE(LU10)
      CLOSE(LU20)
      STOP
      END
C**********************************************************************
      INTEGER*4 FUNCTION IYN(IA)
C  Reads a single character, returns 0 if not 'Y','N',or ' '.
C  Returns 1 if 'Y','y', 2 if 'N','n'.  If ' ', returns value of IA.
C      INCLUDE "LUS.INC"
C
      IMPLICIT NONE
      INTEGER*4 IA
      CHARACTER ANS*1
C
      READ(*,'(A1)')ANS
      IF(SCAN('yYy',ANS(1:1)).GT. 0)THEN
        IYN = 1
      ELSEIF (SCAN('nNn',ANS(1:1)).GT.0)THEN
        IYN = 2
      ELSEIF (ANS(1:1) .EQ. ' ')THEN
        IYN = IA
      ELSE
        IYN = 0
        WRITE(*,1000)
 1000   FORMAT(14X,'*** Please enter only "Y" or "N" ***')
      ENDIF
      RETURN
      END
```

Example #2

```
C     THIS VERSION OF TESTDLL IS AN EXAMPLE OF APPLYING THINNING USING
C     THE EXECUTE SUBROUTINE IN THE ORGRUN.DLL

C
      PROGRAM TESTDLL
C
      IMPLICIT NONE
C
C
      INTEGER*4  LU10,LU20,L,LEN,IANS,IYN,I,J
      INTEGER*4  NPTS,NTREES,STAGE,BHAGE,BIG6,OTHER,MISS,NEST,VERSION,
     1           EDITION,FTYPE(5),TDATAI(2000,3),PTNO(2000)
      INTEGER*4  CYCLG,ISTAGE,IBHAGE,TREENO(2000),SPECIES(2000),
     1           USER(2000),NPR(2000),PRAGE(2000,3),BRCNT(2000,3),
     2           BRHT(2000,40),BRDIA(2000,40),JCORE(2000,40),SERROR(35),
     3           TERROR(2000,6),SWARNING(9),TWARNING(2000)
      INTEGER*4  INDS(30),IERROR,NTREES2
      REAL*4     SITE_1,SITE_2,MSDI_1,MSDI_2,MSDI_3,ACALIB(3,18),BAF(5),
     1           VALUE(5),TDATAR(2000,4),PGROWTH(2000)
      REAL*4     DBH1(2000),HT1(2000),CR1(2000),EXPAN1(2000),SCR1(2000),
     1           PN(5),YSF(5),BABT,BART(5),YST(5),PRLH(2000,3),
     2           PRDBH(2000,3),PRHT(2000,3),PRCR(2000,3),PREXP(2000,3),
     3           OLD,MGEXP(2000),DGRO(2000),HGRO(2000),CRCHNG(2000),
     4           SCRCHNG(2000),MORTEXP(2000),STOR(30), DBH2(2000),
     5           HT2(2000),CR2(2000),EXPAN2(2000),SCR2(2000),RVARS(30)
      CHARACTER  ANS1*1
      CHARACTER  DNAME*128,SNAME*128,ONAME*128
      CHARACTER  TITLE*20,TYPE*1(5)
      LOGICAL    HERE
      LOGICAL*2  EVEN,RAD
C
C     IMPORT THE EXECUTE SUBROUTINE FOR THE ORGRUN.DLL
C
      DLL_IMPORT EXECUTE
C
C     IMPORT THE READDATA SUBROUTINE FROM THE READINP.DLL FOR
C     READING ORGANON .INP FILES
C
      DLL_IMPORT READDATA
C
C
      LU10=10
      LU20=20
C
C     DETERMINE THE NAME OF THE ORGANON CREATED .INP FILE TO OPEN
C
   10 WRITE(*,1000)
 1000 FORMAT(' Enter tree file name [ NO extension (i.e., ',
     1       'C:MYFILE) ]',/' ---> '\)
      READ(*,'(A128)') DNAME
      L = ICHAR(DNAME(1:1))
      IF ( L .EQ. 17 .OR. L .EQ. 27 ) GOTO 99
      LEN = LEN_TRIM(DNAME)
      IF ( LEN .GT. 124 ) LEN = 124
      SNAME=DNAME(1:LEN)//'.INP'
      INQUIRE(FILE=SNAME,EXIST=HERE)
      IF(HERE) THEN
         CLOSE(LU10)
      ELSE
```

```
         WRITE(*,1100)DNAME
 1100    FORMAT(
     1   ' FILE DOES NOT EXIST!'/1X,A124/' Either re-enter',
     2   ' a data file name (check DRIVE) or enter <CTRL-Q>'/
     3   ' to restart ORGANON to enter or edit a new data file.')
         L=65
         GO TO 10
       ENDIF
C
C     OPEN OUTPUT FILE
C
   20 WRITE(*,1200)
 1200 FORMAT(1X,'Enter output file name ---> '\)
   30 READ(*,'(A128)')ONAME
      IF(ONAME .EQ. ' ')THEN
           WRITE(*,1300)
 1300      FORMAT(14X,'*** ERROR--REENTER ---> '\)
           GO TO 30
      ENDIF
      LEN = LEN_TRIM(ONAME) - 1
      DO I = 1,LEN
        IF(ONAME(I:I) .EQ. ' ' .AND. ONAME(I+1:I+1) .NE. ' ') THEN
          WRITE(*,1400)
 1400     FORMAT(14X,'INVALID FILE NAME - REENTER ---> '\)
          GO TO 30
        ENDIF
      ENDDO
      INQUIRE(FILE=ONAME,EXIST=HERE)
      IF(HERE)THEN
        IF(LEN .GT. 64) THEN
           WRITE(*,1500)ONAME
 1500      FORMAT(1X,A128/' FILE EXISTS! -- OVERWRITE? (Y) --->'\)
        ELSE
           WRITE(*,1600)ONAME
 1600   FORMAT('FILE "',A64,'" EXISTS!'/,' OVERWRITE? (Y) --->'\)
        ENDIF
   40   IANS = IYN(1)
        IF(IANS .EQ. 2)GO TO 20
        IF(IANS .EQ. 0)THEN
          WRITE(*,1300)
          GO TO 40
        ENDIF
      ENDIF
      OPEN(LU20,FILE=ONAME,STATUS='UNKNOWN')
C
C     READ TREE DATA FROM THE ORGANON CREATED .INP FILE USING THE
C     READDATA SUBROUTINE FROM THE READINP.DLL
C
      CALL READDATA(SNAME,TITLE,NPTS,NTREES1,SITE_1,SITE_2,EVEN,RAD,
     1     STAGE,BHAGE,BIG6,OTHER,MISS,NEST,VERSION,EDITION,
     2     MSDI_1,MSDI_2,MSDI_3,ACALIB,TYPE,FTYPE,BAF,VALUE,TDATAI,
     3     TDATAR,PGROWTH,PTNO)
C
C     TRANSFER DATA TO THE EXECUTE SUBROUTINE VARIABLES AND ARRAYS
C
      DO I=1,30
         INDS(I)=0
      ENDDO
      ISTAGE=STAGE
      IBHAGE=BHAGE
      IF(EVEN) THEN
        INDS(4)=1
      ENDIF
```

```
      DO I=1,NTREES1
        TREENO(I)=I                              !  TEMPORARY TREE NUMBERS
        SPECIES(I)=TDATAI(I,1)
        USER(I)=TDATAI(I,3)
        DBH1(I)=TDATAR(I,1)
        HT1(I)=TDATAR(I,2)
        CR1(I)=TDATAR(I,3)
        EXPAN1(I)=TDATAR(I,4)
      ENDDO
C
C     INITIALIZE SELECTED VARIABLES TO ZERO BEFORE RUNNING EXECUTE DLL
C
      CYCLG=0
      DO I=1,NTREES1
        SCR1(I)=0.0
        MGEXP(I)=0.0
        DO J=1,3
           BRCNT(I,J)=0
        ENDDO
        DO J=1,40
           BRHT(I,J)=0
           BRDIA(I,J)=0
           JCORE(I,J)=0
        ENDDO
      ENDDO
      BABT=0.0
      DO I=1,5
        PN(I)=0.0
        YSF(I)=0.0
        BART(I)=0.0
        YST(I)=0.0
      ENDDO
      RVARS(1)=SITE_1
      RVARS(2)=SITE_2
      RVARS(3)=MSDI_1
      RVARS(4)=MSDI_2
      RVARS(5)=MSDI_3
C
C     SET DEFAULT VALUES FOR SELECTED VARIABLES BEFORE RUNNING EXECUTE DLL
C
C     IN THE FOLLOWING EXAMPLE, THE STAND IS GROWN ONE CYCLE, THINNED,
C     GROWN FOR TWO CYCLES, THINNED AGAIN, AND GROWN FOR AN ADDITIONAL
C     CYCLE
C
      INDS(9)=1                                  !  USE LIMIT ON MAXIMUM SDI
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C     ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C     FOR THE NEXT CALL TO EXECUTE
C
      DO I=1,NTREES2
        DBH1(I)=DBH2(I)
        HT1(I)=HT2(I)
        CR1(I)=CR2(I)
        SCR1(I)=SCR2(I)
        EXPAN1(I)=EXPAN2(I)
      ENDDO
```

```
      NTREES1=NTREES2
C
C     SET THINNING VARIABLES
C
C     REMOVE ALL TREES WITH A DBH > 10.0"
C
      DO I=1,NTREES1
         BABT=BABT+0.005454154*DBH(I)*DBH1(I)*EXPAN1(I)/FLOAT(NPTS)
         IF(DBH1(I) .GT. 10.0) THEN
            IF(VERSION .EQ. 1)THEN
               IF(SPECIES(I) .EQ. 202 .OR. SPECIES(I) .EQ. 15 .OR.
     1            SPECIES(I) .EQ. 17 .OR. SPECIES(I) .EQ. 122 .OR.
     2            SPECIES(I) .EQ. 117 .OR. SPECIES(I) EQ. 81) THEN
                  INDS(13)=1
               ELSE
                  INDS(13)=0
               ENDIF
            ELSE
               IF(SPECIES(I) .EQ. 202 .OR. SPECIES(I) .EQ. 17 .OR.
     1            SPECIES(I) .EQ. 263) THEN
                  INDS(13)=1
               ELSE
                  INDS(13)=0
               ENDIF
            ENDIF
            MGEXP(I)=EXPAN(I)
            EXPAN(I)=0.0
            BART(1)=BART(1)
     1              +0.005454154*DBH1(I)*DBH1(I)*MGEXP1(I)/FLOAT(NPTS)
         ENDIF
      ENDDO
      IF(BART(1) .GT. 0.0) THEN
         INDS(7)=1                              !  THINNING INDICATOR
         YST(1)=5.0*FLOAT(CYCLG)
      ELSE
         INDS(7)=0
         BABT=0.0
         YST(1)=0.0
      ENDIF
      DO J=1,2
         IF(J .GT. 1) THEN
            INDS(13)=0
         ENDIF
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C        ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C        FOR THE NEXT CALL TO EXECUTE
C
         DO I=1,NTREES2
           DBH1(I)=DBH2(I)
           HT1(I)=HT2(I)
           CR1(I)=CR2(I)
           SCR1(I)=SCR2(I)
           EXPAN1(I)=EXPAN2(I)
         ENDDO
         NTREES1=NTREES2
      ENDDO
```

```
C
C     SET THINNING VARIABLES
C
C     MOVE THE VARIABLES FOR THE FIRST THINNING INTO THE SECOND
C     POSITION AND REINITIALIZE APPROPRIATE VARIABLES
C
      IF(BART(1) .GT. 0.0) THEN
         DO I=5,2,-1
            YST(I)=YST(I-1)
            BART(I)=BART(I-1)
         ENDDO
         BART(1)=0.0
         BABT=0.0
      ENDIF
C
C     REMOVE ALL HARDWOOD TREES
C
      INDS(13)=0                  !  HARDWOODS ARE NOT A MAJOR CONIFER SPECIES
      DO I=1,NTREES1
         BABT=BABT+0.005454154*DBH1(I)*DBH1(I)*EXPAN1(I)/FLOAT(NPTS)
         MGEXP(I)=0.0
         IF(SPECIES(I) .GT. 300) THEN
            MGEXP(I)=EXPAN1(I)
            EXPAN(I)=0.0
            BART(1)=BART(1)
     1             +0.005454154*DBH1(I)*DBH(I)*MGEXP(I)/FLOAT(NPTS)
         ENDIF
      ENDDO
      IF(BART(1) .GT. 0.0) THEN
         INDS(7)=1                                !  THINNING INDICATOR
         YST(1)=5.0*FLOAT(CYCLG)
      ELSE
         INDS(7)=0
         BABT=0.0
         YST(1)=0.0
      ENDIF
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C     ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C     FOR THE ENDING VALUES
C
      DO I=1,NTREES2
        DBH1(I)=DBH2(I)
        HT1(I)=HT2(I)
        CR1(I)=CR2(I)
        SCR1(I)=SCR2(I)
        EXPAN1(I)=EXPAN2(I)
      ENDDO
      NTREES1=NTREES2
C
C     OUTPUT RESULTS FROM THE RUN
C
      IF(IERROR .EQ. 1) THEN
C
C        OUTPUT INFORMATION ABOUT ERRORS IF THEY OCCURED
C
         WRITE(LU20,1700)
```

```
 1700    FORMAT(' STAND LEVEL ERRORS')
         DO I=1,34
            WRITE(LU20, 1750) I,SERROR(I)
 1750       FORMAT(I2,', ',I1)
         ENDDO
         WRITE(LU20,1800)
 1800    FORMAT(' STAND LEVEL WARNINGS')
         DO I=1,9
            WRITE(LU20, 1850) I, SWARNING(I)
 1850       FORMAT(I2,', ',I1)
         ENDDO
         WRITE(LU20,1900)
 1900    FORMAT(' TREE LEVEL ERRORS AND A WARNING')
         DO I=1,NTREES1
            WRITE(LU20,1950) I,TERROR(I,1),TERROR(I,2),TERROR(I,3),
     1                       TERROR(I,4),TERROR(I,5),TERROR(I,6),
     2                       TWARNING(I)
 1950       FORMAT(I4,', ',6(I1,', '),I1)
         ENDDO
      ELSE
C
C        OUTPUTS TREE INFORMATION TO THE TREE LIST FILE
C
         DO I = 1,NTREES1
            IF(EXPAN1(I) .GT. 0.000000001) THEN
               WRITE(LU20,2000) CYCLG,PTNO(I),I,SPECIES(I),USER(I),
     1                          DBH1(I),HT1(I),CR1(I),EXPAN1(I),MORTEXP(I),
     2                          DGRO(I),HGRO(I)
 2000          FORMAT(I4,', ',I3,', ',I4,', ',I3,', ',I3,', ',F6.1,', ',
     1                F6.1,', ',F5.2,', ',F8.2,', ',F8.2,', ',F8.2,', ',
     2                F8.2)
            ENDIF
         ENDDO
      ENDIF
   99 CLOSE(LU10)
      CLOSE(LU20)
      STOP
      END
C
C*******************************************************************
      INTEGER*4 FUNCTION IYN(IA)
C  Reads a single character, returns 0 if not 'Y','N',or ' '
C  Returns 1 if 'Y','y', 2 if 'N','n'.  If ' ', returns value of IA.
C      INCLUDE "LUS.INC"
C
      IMPLICIT NONE
      INTEGER*4 IA
      CHARACTER ANS*1
C
      READ(*,'(A1)')ANS
      IF(SCAN('yYy',ANS(1:1)).GT. 0)THEN
        IYN = 1
      ELSEIF (SCAN('nNn',ANS(1:1)).GT.0)THEN
        IYN = 2
      ELSEIF (ANS(1:1) .EQ. ' ')THEN
        IYN = IA
      ELSE
        IYN = 0
        WRITE(*,1000)
 1000   FORMAT(14X,'*** Please enter only "Y" or "N" ***')
      ENDIF
      RETURN
      END
```

Example #3

```
C     THIS VERSION OF TESTDLL IS AN EXAMPLE OF APPLYING PRUNING USING
C     THE EXECUTE SUBROUTINE IN THE ORGRUN.DLL

C
      PROGRAM TESTDLL
C
      IMPLICIT NONE
C
C
      INTEGER*4  LU10,LU20,L,LEN,IANS,IYN,I,J
      INTEGER*4  NPTS,NTREES1,STAGE,BHAGE,BIG6,OTHER,MISS,NEST,VERSION,
     1           EDITION,FTYPE(5),TDATAI(2000,3),PTNO(2000)
      INTEGER*4  CYCLG,ISTAGE,IBHAGE,TREENO(2000),SPECIES(2000),
     1           USER(2000),NPR(2000),PRAGE(2000,3),BRCNT(2000,3),
     2           BRHT(2000,40),BRDIA(2000,40),JCORE(2000,40),SERROR(35),
     3           TERROR(2000,6),SWARNING(9),TWARNING(2000)
      INTEGER*4  INDS(30),IERROR,NTREES2
      INTEGER*4  TSP
      REAL*4     SITE_1,SITE_2,MSDI_1,MSDI_2,MSDI_3,ACALIB(3,18),BAF(5),
     1           VALUE(5),TDATAR(2000,4),PGROWTH(2000)
      REAL*4     DBH1(2000),HT1(2000),CR1(2000),EXPAN1(2000),SCR1(2000),
     1           PN(5),YSF(5),BABT,BART(5),YST(5),PRLH(2000,3),
     2           PRDBH(2000,3),PRHT(2000,3),PRCR(2000,3),PREXP(2000,3),
     3           OLD,MGEXP(2000),DGRO(2000),HGRO(2000),CRCHNG(2000),
     4           SCRCHNG(2000),MORTEXP(2000),STOR(30), DBH2(2000),
     5           HT2(2000),CR2(2000),EXPAN2(2000),SCR2(2000),RVARS(30)
      REAL*4     MAXLH,TCR,MDBH,MAXTCR,X,DIFF
      CHARACTER  ANS1*1
      CHARACTER  DNAME*128,SNAME*128,ONAME*128
      CHARACTER  TITLE*20,TYPE*1(5)
      LOGICAL    HERE
      LOGICAL*2  EVEN,RAD
C
C     IMPORT THE EXECUTE SUBROUTINE FOR THE ORGRUN.DLL
C
      DLL_IMPORT EXECUTE
C
C     IMPORT THE READDATA SUBROUTINE FROM THE READINP.DLL FOR
C     READING ORGANON .INP FILES
C
      DLL_IMPORT READDATA
C
C
      LU10=10
      LU20=20
C
C     DETERMINE THE NAME OF THE ORGANON CREATED .INP FILE TO OPEN
C
   10 WRITE(*,1000)
 1000 FORMAT(' Enter tree file name [ NO extension (i.e., ',
     1       'C:MYFILE) ]',/' ---> '\)
      READ(*,'(A128)') DNAME
      L = ICHAR(DNAME(1:1))
      IF ( L .EQ. 17 .OR. L .EQ. 27 ) GOTO 99
      LEN = LEN_TRIM(DNAME)
      IF ( LEN .GT. 124 ) LEN = 124
      SNAME=DNAME(1:LEN)//'.INP'
      INQUIRE(FILE=SNAME,EXIST=HERE)
      IF(HERE) THEN
```

```
         CLOSE(LU10)
      ELSE
         WRITE(*,1100)DNAME
 1100    FORMAT(
     1   ' FILE DOES NOT EXIST!'/1X,A124/' Either re-enter',
     2   ' a data file name (check DRIVE) or enter <CTRL-Q>'/
     3   ' to restart ORGANON to enter or edit a new data file.')
         L=65
         GO TO 10
      ENDIF
C
C     OPEN OUTPUT FILE
C
   20 WRITE(*,1200)
 1200 FORMAT(1X,'Enter output file name ---> '\)
   30 READ(*,'(A128)')ONAME
      IF(ONAME .EQ. ' ')THEN
           WRITE(*,1300)
 1300      FORMAT(14X,'*** ERROR--REENTER ---> '\)
           GO TO 30
      ENDIF
      LEN = LEN_TRIM(ONAME) - 1
      DO I = 1,LEN
        IF(ONAME(I:I) .EQ. ' ' .AND. ONAME(I+1:I+1) .NE. ' ') THEN
          WRITE(*,1400)
 1400     FORMAT(14X,'INVALID FILE NAME - REENTER ---> '\)
          GO TO 30
        ENDIF
      ENDDO
      INQUIRE(FILE=ONAME,EXIST=HERE)
      IF(HERE)THEN
        IF(LEN .GT. 64) THEN
           WRITE(*,1500)ONAME
 1500      FORMAT(1X,A128/' FILE EXISTS! -- OVERWRITE? (Y) --->'\)
        ELSE
           WRITE(*,1600)ONAME
 1600   FORMAT('FILE "',A64,'" EXISTS!'/,' OVERWRITE? (Y) --->'\)
        ENDIF
   40   IANS = IYN(1)
        IF(IANS .EQ. 2)GO TO 20
        IF(IANS .EQ. 0)THEN
          WRITE(*,1300)
          GO TO 40
        ENDIF
      ENDIF
      OPEN(LU20,FILE=ONAME,STATUS='UNKNOWN')
C
C     READ TREE DATA FROM THE ORGANON CREATED .INP FILE USING THE
C     READDATA SUBROUTINE FROM THE READINP.DLL
C
      CALL READDATA(SNAME,TITLE,NPTS,NTREES1,SITE_1,SITE_2,EVEN,RAD,
     1     STAGE,BHAGE,BIG6,OTHER,MISS,NEST,VERSION,EDITION,
     2     MSDI_1,MSDI_2,MSDI_3,ACALIB,TYPE,FTYPE,BAF,VALUE,TDATAI,
     3     TDATAR,PGROWTH,PTNO)
C
C     TRANSFER DATA TO THE EXECUTE SUBROUTINE VARIABLES AND ARRAYS
C
      DO I=1,30
         INDS(I)=0
      ENDDO
      ISTAGE=STAGE
      IBHAGE=BHAGE
      IF(EVEN) THEN
```

```
        INDS(4)=1
      ENDIF
      DO I=1,NTREES1
        TREENO(I)=I                              !  TEMPORARY TREE NUMBERS
        SPECIES(I)=TDATAI(I,1)
        USER(I)=TDATAI(I,3)
        DBH1(I)=TDATAR(I,1)
        HT1(I)=TDATAR(I,2)
        CR1(I)=TDATAR(I,3)
        EXPAN1(I)=TDATAR(I,4)
      ENDDO
C
C     INITIALIZE SELECTED VARIABLES TO ZERO BEFORE RUNNING EXECUTE DLL
C
      CYCLG=0
      DO I=1,NTREES1
        SCR1(I)=0.0
        MGEXP(I)=0.0
        DO J=1,3
           BRCNT(I,J)=0
        ENDDO
        DO J=1,40
           BRHT(I,J)=0
           BRDIA(I,J)=0
           JCORE(I,J)=0
        ENDDO
      ENDDO
      BABT=0.0
      DO I=1,5
        PN(I)=0.0
        YSF(I)=0.0
        BART(I)=0.0
        YST(I)=0.0
      ENDDO
      RVARS(1)=SITE_1
      RVARS(2)=SITE_2
      RVARS(3)=MSDI_1
      RVARS(4)=MSDI_2
      RVARS(5)=MSDI_3
C
C     IN THE FOLLOWING EXAMPLE, THE STAND IS GROWN ONE CYCLE, PRUNED,
C     GROWN FOR TWO CYCLES, PRUNED AGAIN, AND GROWN FOR AN ADDITIONAL
C     CYCLE
C
      INDS(9)=1                                  !  USE LIMIT ON MAXIMUM SDI
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C     ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C     FOR THE NEXT CALL TO EXECUTE
C
      DO I=1,NTREES2
        DBH1(I)=DBH2(I)
        HT1(I)=HT2(I)
        CR1(I)=CR2(I)
        SCR1(I)=SCR2(I)
        EXPAN1(I)=EXPAN2(I)
      ENDDO
```

```
      NTREES1=NTREES2
C
C     SET PRUNING VARIABLES
C
C
C     THE FOLLOWING PRUNING HAS A TARGET CROWN RATIO AFTER PRUNING OF 0.5,
C     A MAXIMUM LIFT HEIGHT OF 16', A MINIMUM DBH OF 4" AND TARGETS JUST
C     DOUGLAS-FIR
C
      INDS(6)=1                              !  PRUNING INDICATOR
      MAXLH=16.0                             !  SET MAX. LIFT HEIGHT FOR PRUNING
      TSP=202                                !  SET TARGET SPECIES FOR PRUNING
      TCR=0.5                               !  SET TARGET CROWN RATIO FOR PRUNING
      MDBH=4.0                              !  SET MINIMUM DBH FOR PRUNED TREES
      DO I=1,NTREES
         IF(EXPAN1(I) .LE. 0.0) CYCLE
         IF(SPECIES(I) .NE. TSP) CYCLE
         IF(DBH1(I) .LT. MDBH) CYCLE
         MAXTCR=1.0-MAXLH/HT1(I)              !  DETERMINE MAXIMUM CROWN RATIO
         X=TCR
         IF(MAXTCR .GT. X) X=MAXTCR
         DIFF=(CR1(I)-X)*HT1(I)
         IF(DIFF .GE. 0.1) THEN
            IF(SCR1(I) .LE. CR1(I)) THEN
               SCR1(I)=CR1(I)
            ENDIF
            CR(I)=X
         ENDIF
         NPR(I)=1
         PRAGE(I,1)=IBHAGE
         PRDBH(I,1)=DBH1(I)
         PRHT(I,1)=HT1(I)
         PRCR(I,1)=CR1(I)
         PREXP(I,1)=EXPAN1(I)
      ENDDO
      DO J=1,2
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C        ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C        FOR THE NEXT CALL TO EXECUTE
C
         DO I=1,NTREES2
           DBH1(I)=DBH2(I)
           HT1(I)=HT2(I)
           CR1(I)=CR2(I)
           SCR1(I)=SCR2(I)
           EXPAN1(I)=EXPAN2(I)
         ENDDO
         NTREES1=NTREES2
      ENDDO
C
C     SET PRUNING VARIABLES
C
C
C     THE FOLLOWING PRUNING HAS A TARGET CROWN RATIO AFTER PRUNING OF 0.3,
C     A MAXIMUM LIFT HEIGHT OF 32', A MINIMUM DBH OF 10" AND TARGETS JUST
C     DOUGLAS-FIR
```

```
C
      MAXLH=32.0                          !  SET MAX. LIFT HEIGHT FOR PRUNING
      TSP=202                             !  SET TARGET SPECIES FOR PRUNING
      TCR=0.3                             !  SET TARGET CROWN RATIO FOR PRUNING
      MDBH=10.0                           !  SET MINIMUM DBH FOR PRUNED TREES
      DO I=1,NTREES1
         IF(EXPAN(I) .LE. 0.0) CYCLE
         IF(SPECIES(I) .NE. TSP) CYCLE
         IF(DBH1(I) .LT. MDBH) CYCLE         !  DETERMINE MAXIMUM CROWN RATIO
         MAXTCR=1.0-MAXLH/HT(I)
         X=TCR
         IF(MAXTCR .GT. X) X=MAXTCR
         DIFF=(CR1(I)-X)*HT1(I)
         IF(DIFF .GE. 0.1) THEN
            IF(SCR1(I) .LE. CR1(I)) THEN
               SCR1(I)=CR1(I)
            ENDIF
            CR1(I)=X
         ENDIF
         NPR(I)=1
         PRAGE(I,2)=IBHAGE
         PRDBH(I,2)=DBH1(I)
         PRHT(I,2)=HT1(I)
         PRCR(I,2)=CR1(I)
         PREXP(I,2)=EXPAN1(I)
      ENDDO
      CALL EXECUTE(CYCLG,VERSION,NPTS,NTREES1,ISTAGE,IBHAGE,TREENO,PTNO,
     1             SPECIES,USER,INDS,DBH1,HT1,CR1,SCR1,EXPAN1,MGEXP,
     2             RVARS,ACALIB,PN,YSF,BABT,BART,YST,NPR,PRAGE,PRLH,
     3             PRDBH,PRHT,PRCR,PREXP,BRCNT,BRHT,BRDIA,JCORE,SERROR,
     4             TERROR,SWARNING,TWARNING,IERROR,DGRO,HGRO,CRCHNG,
     5             SCRCHNG,MORTEXP,NTREES2,DBH2,HT2,CR2,SCR2,EXPAN2,
     6             STOR)
C
C     ADD GROWTH AND CHANGE TO BEGINNING OF GROWTH CYCLE VALUES
C     FOR THE ENDING VALUES
C
      DO I=1,NTREES2
        DBH1(I)=DBH2(I)
        HT1(I)=HT2(I)
        CR1(I)=CR2(I)
        SCR1(I)=SCR2(I)
        EXPAN1(I)=EXPAN2(I)
      ENDDO
      NTREES1=NTREES2
C
C     OUTPUT RESULTS FROM THE RUN
C
      IF(IERROR .EQ. 1) THEN
C
C        OUTPUTS INFORMATION ABOUT ERRORS IF THEY OCCURED
C
         WRITE(LU20,1700)
 1700    FORMAT(' STAND LEVEL ERRORS')
         DO I=1,34
            WRITE(LU20, 1750) I,SERROR(I)
 1750       FORMAT(I2,', ',I1)
         ENDDO
         WRITE(LU20,1800)
 1800    FORMAT(' STAND LEVEL WARNINGS')
         DO I=1,9
            WRITE(LU20, 1850) I,SWARNING(I)
 1850       FORMAT(I2,', ',I1)
```

```
         ENDDO
         WRITE(LU20,1900)
 1900    FORMAT(' TREE LEVEL ERRORS AND A WARNING')
         DO I=1,NTREES1
            WRITE(LU20,1950) I,TERROR(I,1),TERROR(I,2),TERROR(I,3),
     1                       TERROR(I,4),TERROR(I,5),TERROR(I,6),
     2                       TWARNING(I)
 1950       FORMAT(I4,', ',6(I1,', '),I1)
         ENDDO
      ELSE
C
C        OUTPUTS TREES TO THE TREE LIST FILE
C
         DO I = 1,NTREES1
            IF(EXPAN1(I) .GT. 0.000000001) THEN
               WRITE(LU20,2000) CYCLG,PTNO(I),I,SPECIES(I),USER(I),
     1                          DBH1(I),HT1(I),CR1(I),EXPAN1(I),MORTEXP(I),
     2                          DGRO(I),HGRO(I)
 2000          FORMAT(I4,', ',I3,', ',I4,', ',I3,', ',I3,', ',F6.1,', ',
     1                F6.1,', ',F5.2,', ',F8.2,', ',F8.2,', ',F8.2,', ',
     2                F8.2)
            ENDIF
         ENDDO
      ENDIF
   99 CLOSE(LU10)
      CLOSE(LU20)
      STOP
      END
C
C**********************************************************************
      INTEGER*4 FUNCTION IYN(IA)
C
C  Reads a single character, returns 0 if not 'Y','N',or ' ',
C  Returns 1 if 'Y','y', 2 if 'N','n'.  If ' ', returns value of IA.
C      INCLUDE "LUS.INC"
C
      IMPLICIT NONE
      INTEGER*4 IA
      CHARACTER ANS*1
C
      READ(*,'(A1)')ANS
      IF(SCAN('yYy',ANS(1:1)).GT. 0)THEN
        IYN = 1
      ELSEIF (SCAN('nNn',ANS(1:1)).GT.0)THEN
        IYN = 2
      ELSEIF (ANS(1:1) .EQ. ' ')THEN
        IYN = IA
      ELSE
        IYN = 0
        WRITE(*,1000)
 1000   FORMAT(14X,'*** Please enter only "Y" or "N" ***')
      ENDIF
      RETURN
      END
```