

000 NF NEXT SANS FOR	002 SN ERREUR DE SYNTAXE
004 RG RETURN SANS GOTO	006 OD FIN DE DATA POUR READ
008 FC FONCTION ILLEGALE	010 OV DEPASSEMENT DE CAPACITE
012 OM DEPASSEMENT DE MEMOIRE	014 UL LIGNE INCONNUE
016 BS FAUX PARAMETRES	018 DD TABLEAU MAL DIMENSIONNE
020 /0 DIVISION PAR 0	022 ID COM. DIRECTE INTERDITE
024 TM FAUTE DE FRAPPE	026 OS TROP DE VARIABLES
028 LS CHAINE TROP LONGUE	030 ST CHAINE TROP COMPLEXE
032 CN NE PEUT CONTINUER	034 UF (FUTURS DEVELOPEMENTS)
036 FD FAUTES DE DATAS	038 AD FICHIER DEJA OUVERT
040 DN NUMERO DU DRIVE	042 IO ERREUR D ENTRE/SORTIE
044 FM MODE FICHIER ERRONE	046 NO FICHIER NON OUVERT
048 IE ENTRE POST EOF	050 DS ORDRE DIRECT
128 NR LECTEUR PAS PRET	130 SK DEFAUT DISQUE
132 WP PROTEGE A L ECRITURE	134 RT TYPE D ENREGISTREMENT
136 RF FICHIER INCONNU	138 CC BOUCLE SANS FIN
140 LD DATAS PERDUS	142 BT BOOT
144 IV DIRECTORY NON ACCESSIBLE	146 FD DIRECTORY PLEIN
148 OF DIQUE PLEIN	150 FS SPECIFICATION DE FICHIER
152 PT PROTECTION	154 PE LECTURE POST EOF
156 FF FICHE NON EXISTANT	158 FE FICHIER EXISTANT
160 NF N EXISTE PAS	162 TF PLUS DE 10 FICHIERS
164 PR PARAMETRES	

RUN:

SERT A TRANSFERER ET A EXECUTER DIRECTEMENT UN PROGRAMME.

RUN'PROG1' LOAD LE PROGRAMME ET L EXECUTE DIRECTEMENT.

RUN'3:FICHE.BAK' VA LOADER ET EXECUTER DIRECTEMENT LE FICHIER FICHE DU LECTEUR 3

BACKUP:

LA COMMANDE 'BACKUP' PERMET DE REALISER UNE COPIE COMPLETE DE DISQUETTE.
 BACKUP 3 TO 4,2,80 PERMET DONC DE CREER UNE COPIE DU DISQUE 3 SUR LE DISQUE 4
 (2 FACES,80 PISTES).

LA SOURCE ET LA DESTINATION PEU VENT ETRE LE MEME LECTEUR... LES DISQUETTES D
 EVRONT ETRE ALTERNES AUX ORDRES DU DRAGON.

BOOT:

LA COMMANDE BOOT PERMET LE CHAR-GEMENT D UN NOUVEAU SYSTEME D EXPLOITATION OU LA
 NGAGE A PARTIR DE L ADRESSE 9728.

BOOT SERT SI UN SEUL LECTEUR ESTBRANCHE,BOOT N (N=1,2,3 OU4) SI IL Y A PLUSIEURS
 LECTEURS.

L EXECUTION EST OBTENUE PAR
 EXEC 9730

CHAIN:

SERT A LOADER ET EXECUTER UN DESPROGRAMMES SANS DETRUIRE LES VA-RIABLES EN MEMOI
 RE.UTILISE SI VOUSVOULEZ FAIRE TOURNEE PLUSIEURS PROGRAMMES SUR LES MEMES DATAS.
 CHAIN 'PROG2' LOAD ET RUN LE PROGRAMME 2 DU LECTEUR1 SANS REMET-TRE LES VARIABLE
 S A 0.

CHAIN 'PROG' JOINT LE PROGRAMME 3 ET L'EXECUTE A PARTIR DE LIGNE 36.
CHAIN '2:PROG' PERMET DE REALISER L INSTRUCTION CHAIN SUR LE PROGRAMME 'PROG' DU LECTEUR 2.

QUAND ON UTILISE CHAIN AVEC DES PROGRAMMES UTILISANT DES CHAINES DE VARIABLE, IL EST CONSEILLE DE CREER UN REPERTOIRE DE CELLES CI AU DEPART. POUR CELA, UTILISEZ LA FONCTION FREQ DE LA FACON SUIVANTE:

10 ZZ=FREQ'REPERTOIRE
20 REM PROGRAMME CHAINE

CLOSE:

CETTE COMMANDE EST UTILISEE POUR FERMER UN FICHIER OUVERT PAR LES INSTRUCTIONS FWR, FTE, FREAD, FLREAD. 10 FICHIERS PEUVENT ETRE OUVERTS EN MEME TEMPS. DONC IL PEUT ETRE NECESSAIRE D EN FERMER UN A UN MOMENT DONNE. LE POINTEUR (VOIR ALOC) SERA REMIS CLOSE FERME TOUT LES FICHIERS.

CLOSE 2 FERME LES FICHIERS SUR LE LECTEUR 2.

COPY:

CETTE COMMANDE EST UTILISEE POUR DOUBLER UN FICHIER. LA COPIE PEUT ETRE SUR UN DEUXIEME DISQUE OU SUR UN SEUL (SI UN SEUL LECTEUR).

COPY 'NOM.BAS' TO 'NOM1.BAS' VA DONNER UNE COPIE DE NOM APPELE NOM1 SUR LE MEME LECTEUR.

COPY '1:NOM.BAS' TO '2:NOM1.BAS' VA COPIER NOM SUR LE LECTEUR 2.

CREATE:

CETTE COMMANDE SERT A CREER DES FICHIERS DE DONNEES.

CREATE 'NOM' CREE UN FICHIER DE TYPE NOM.DAT DE LONGUEUR 0 BYTE

CREATE 'NOM',80 CREE UN FICHIER DUNE LONGUEUR DE 80 BYTES.

DIR:

LA COMMANDE DIR DONNE LA LISTE DES FICHIERS SUR LE DISQUE, LA LONGUEUR DE CEUX-CI ET LA PLACE RESTANT SUR LE DISQUE.

DIR LISTE LE PREMIER LECTEUR

DIR 2 LE DISQUE SUR LE LECTEUR 2

BAS SIGNIFIE BASIC

BAK SIGNIFIE COPIE

BIN SIGNIFIE MACHINE

DAT SIGNIFIE DONNEES

DRIVE:

LA COMMANDE DRIVE PERMET DE DEFINIR LE LECTEUR UTILISE. SI DRIVE N EST PAS PRECISE, LE DERNIER LECTEUR UTILISE EST PRIS COMME REFERENCE.

A L ALLUMAGE, DRIVE EST MIS SUR 1 PAR DEFAUT.

DSKINIT:

DSKINIT EST UTILISE POUR L INITIALISATION OU FORMATAGE. ELLE EST LE PREALABLE A TOUTES OPERATIONS SUR UN DISQUE NEUF.

DSKINIT FORMATE UN DISQUE SUR LE LECTEUR 1, A 40 PISTES.

DSKINIT 3,2,80 FORMATE UN DISQUE SUR LE LECTEUR 3, EN DOUBLE FACE ET SUR 80 PISTES S/FACE.

EOF:

CETTE FONCTION VERIFIE SI LE POINTEUR DE LECTURE EST A LA FIN D UN FICHIER C.A.D SI TOUT LES ENREGISTREMENTS D UN FICHIER ONT ETE LUS.

X=EOF('NOM') DONNE A X LA VALEUR 0 SI LE POINTEUR EST A LA FIN, OU 1 AUTREMENT. LES PARENTHESES SONT IMPORTANTES

FLREAD:

EST UTILISE POUR LIRE UNE CHAÎNE D UN FICHIER.VIRGULES ET POINTS NE SONT PAS DES
SIGNAUX DE FIN.
FREAD'NOM'/VAR. LIT LA VARIABLE D UN FICHIER A PARTIR DE LA POSITION DU POINTEUR
(DEBUT FICHIER AU DEPART).LE POINTEUR EST REMIS A LA FICHE SUIVANT CELLE CI.
FREAD'NOM',FROM 100,FOR 60;VAR.LIT A PARTIR DU BYTE 100.APRES LECTURE LE POINT
EUR SERA A 160.
UN FICHIER RESTE OUVERT

FREAD:

CETTE ORDRE EST UTILISE POUR LA LECTURE DES DONNEES D UN FICHIER
FREAD'FICHE'/LISTE VAR. LIT LES VARIABLES D UN FICHIER,A PARTIR DU POINTEUR(AU D
EPART SI DEBUT).LE POINTEUR SE DECALE D UN ORAN A CHAQUE FOIS.
FREAD'FICHE',FROM30,FOR50;LISTE VAR. LIT LES VARIABLES DE LA POSITION 30 A 80.
LES VARIABLES SONT TERMINEES PAR VIRGULES ET POINT.

FREE:

FREE DONNE LE NOMBRE DE BYTES LIBRES SUR LE DISQUE.
PRINT FREE DONNE LA PLACE LIBRE SUR LE LECTEUR 1
PRINT FREE 2 DONNE LE NOMBRE DE BYTES LIBRES SUR LE LECTEUR 2
FREE PEUT ETRE INCLUS DANS LES PROGRAMMES (EX:X=FREE1).

FWRITE:

FWRITE EST UTILISE POUR ECRIRE DES DONNEES SUR UN FICHIER.
FWRITE'FICHE'/LISTE VAR. ECRIT LA VARIABLE DANS UN FICHIER.SI LE FICHIER N AS
PAS ETE OUVERT, IL SERA CREE AUTOMATIQUEMENT.
FWRITE'FICHE',FROM 10,FOR 30;LISTE VAR. ECRIT DU BYTE 10 A 40.LE FICHIER DOIT AV
OIR ETE OUVERT ET DOIT AVOIR 9 BYTES DE LONG.

KILL:

CETTE INSTRUCTION PERMET DE DETRUIRE UN PROGRAMME DU DIRECTORY.
KILL'NOM.BAS' DETRUIT LE PROGRAMME BASIC 'NOM'.
KILL'NOM.BIN' DETRUIT LE FICHIER ASSEMBLEUR NOMME 'NOM'.
KILL'2'NOM.BAS'DETRUIT LE FICHIER 'NOM' SUR LE LECTEUR 2.

LOAD:

LOAD PERMET DE TRANSFERER UN PROGRAMME EN MEMOIRE.
LOAD'NOM' PERMET DE LIRE LE PROGRAMME BASIC 'NOM' SUR LE LECTEUR 1
LOAD'NOM.BIN' PERMET DE LIRE UN PROGRAMME EN MACHINE.LE CHARGEMENT SE FAIT A L A
DRESSE DE SAUVETAGE PAR DEFAUT.POUR EXECUTER,IL FAUT TAPER EXEC POINT D ENTRE.
LOAD'NOM.BIN',1000 LOAD LE PROGRAMME MACHINE,A PARTIR DE L ADRESSE 1000.SI 'NOM'
AVAIT ETE SAUVE SOUS LA FORME 'NOM',DEPART,FIN,ENTRE, LE DEPART SERA A 1000 +
ENTRE-DEPART.
LOAD'NOM.BAK' LOAD LE PROGRAMME BAK.
LOAD'2'NOM' LOAD LE PROGRAMME DE NOM 'NOM' DU LECTEUR 2.

LOC:

CETTE FONCTION EST UTILISEE POUR TROUVER LA POSITION DU POINTEUR.
LOC'FICHE' RETOURNE LE NUMERO DU DERNIER BYTE LU.POUR UN READ,IL S AGIT DU DEFAU
T POUR FROM.

LOF:

LOF DONNE LA LONGUEUR EN BYTES DUN FICHIER.
PRINT LOF'NOM.BIN' VA DONNER LA LONGUEUR DU FICHIER NOM.
LOF PEUT ETRE UTILISE DANS LES PROGRAMMES SOUS LA FORME X=LOF' NOM.BAS'.LA NENT
ION BAS OU BIN EST OBLIGATOIRE.

MERGE:

EST UTILISE POUR RELIER UN PROGRAMME SUR DISQUE AVEC UN PROGRAMME EN MEMOIRE. LE FICHIER SUR DISQUE RESTE INTACT. IL EST NECESSAIRE DE RENUMEROTER AVANT D'EXECUTER CETTE OPERATION.

EN EFFET, SINON, LES LIGNES SERONT MELANGEES.
LE PROGRAMME OBTENU PEUT ETRE RESAUE.

PROTECT:

PERMET DE PROTEGER UN PROGRAMME, SAUF DE DSKINIT.
PROTECTON'NOM.BAS' PROTEGE LE FICHIER 'NOM'.
PROTECTOFF'NOM.BAS' LEVE LA PROTECTION.
LA PROTECTION EST SIGNALÉE PAR UN P EN INVERSION.

RENAME:

EST UTILISEE POUR RENOMMER UN DE VOS PROGRAMMES SUR LE DISQUE. PEUT AUSSI TRANSFERER UN PROGRAMME D'UN DISQUE A UN AUTRE.
RENAME'ANC.BIN' TO 'NOU.BIN' DONNE AU FICHIER ANC LE NOM NOU.
LA MENTION D'ESPECE(BIN,BAS...) EST OBLIGATOIRE.

SAVE:

SERT A TRANSFERER DANS LE SENS MEMOIRE/DISQUE.
SAVE'PROG' CREE UN FICHIER SUR LE DISQUE APPELE 'PROG.BAS'. SI CE NOM EXISTE DEJA, LE PROGRAMME EST APPELE 'PROG.BAK'. SI CELUI CI EXISTE DEJA, IL EST DETRUIT.
SAVE'CODE',2000,5001,4000 SAUVE UN FICHIER MACHINE DEBUTANT A 2000, SE FINISSANT A 5000 ET EXECUTE A L'ADRESSE 4000. BAK MARCHE COMME PRECEDEMENT.
LE NUMERO DU LECTEUR A UTILISER DOIT ETRE PRECISE SI PLUS D'UN LECTEUR EST UTILISE. LE NUMERO DU LECTEUR EST PRECISE DE LA FACON SUIVANTE:

SAVE'2:NOM'
CECI SAUVERAI LE PROGRAMME BASIC NOM SUR LE LECTEUR 2.

SREAD:

UTILISE POUR LIRE LES DONNEES CONTENU SUR UN SECTEUR DU DISQUE. LA DONNEE EST DE 256 BYTES ET SERA LU EN DEUX VARIABLES (A# ET B# PAR EXEMPLE) DE 128 BYTES.
SREAD2,6,10,A#,B# LIT LES 256 BYTES DU 10 SECTEUR DE LA 6 EME. PISTE DU LECTEUR 2.
LES 128 PREMIERS BYTES SONT DANS A#, LES 128 AUTRES DANS B#.
IL PEUT ETRE NECESSAIRE DE PLACER DES CHR\$(0) POUR OBTENIR DES TRANCHES DE 128 CARACTERES.

SWRITE:

SERT A ECRIRE UNE DONNEE SUR UN SEUL SECTEUR. LA DONNEE EST PORTEE PAR 2 VARIABLES DE 128 BYTES.
SWRITE5,7,9,X#,Y# ECRIT SUR LE LECTEUR 5, 9 EME. SECTEUR ET 7 EME PISTE UNE DONNEE PORTEE PAR X# & Y#.
IL FAUT REMPLIR L'ESPACE PAR DES CHR\$(0) POUR OBTENIR DES ZONES DE 128 BYTES.

VERIFY:

LA COMMANDE VERIFY ON ET VERIFY OFF PERMETTENT DE PASSER DE LA PROCEDURE VERIFICATION OUVERTE A LA PROCEDURE VERIFICATION FERME.
A L'OUVERTURE, CETTE COMMANDE EST EN POSITION OUVERTE.(ON)

HIMEM:

CETTE FONCTION DONNE LA PLUS HAUTE ADRESSE MEMOIRE UTILISABLE EN BASIC. ELLE EST MODIFIEE PAR L'USAGE DE CLEAR.
PRINT HIMEM RETOURNE 32766 SAUF SI CLEAR A ETE UTILISE.

PRE#:

DONNE LA PLACE ENCORE DISPONIBLE POUR LES VARIABLES. CE N'EST PAS UNE VARIABLE MAIS UNE VALEUR NUMERIQUE. SON UTILISATION FORCE LA CREATION DU DICTIONNAIRE.
PRINT FRES RETOURNE LE NOMBRE D'OCTETS LIBRES ALLOUES AUX VARIABLES APRES QUE LES VARIABLES DEJA MISES EN PLACE AIENT ETE STOCKEES.

ERROR GOTO:

CETTE COMMANDE EST UN CONTROLE EN MODE DIRECT D'UNE LIGNE SI UNE ERREUR EST DETECTEE.

APRES QUE LA LIGNE PORTANT GOTO 5000 AIT ETE EXECUTEE, SI UNE ERREUR EST DETECTEE LE CONTROLE PASSE A LA LIGNE 5000

ERR:

QUAND UNE ERREUR EST APPARUE, LA FONCTION ERR VA DONNER LE CODE D'ERREUR CORRESPONDANT.

AINSI APRES UN I/O ERROR, TAPER PRINT ERR VA RETOURNER LA VALEUR 42 (VOIR LISTE DES ERREURS).

ERL:

QUAND UNE ERREUR ARRIVE, L'USAGE DE LA FONCTION ERL VA DONNER LE NUMERO DE LA LIGNE OU S'EST PRODUITE L'ERREUR.

AINSI APRES UNE ERREUR EN LIGNE 50, TAPER PRINT ERL RETOURNE LA VALEUR 50

BEEP:

LA COMMANDE BEEP DONNE UN SON DE TYPE 'BEEP' DE BONNE QUALITE.

BEEP DONNE 1 BEEP.

BEEP 20 DONNE 20 BEEP.

WAIT:

CETTE FONCTION SUSPEND L'EXECUTION DU PROGRAMME ET REMPLACE DONC LES BOUCLES DE TYPE FOR...NEXT.

WAIT 5000 VA DONNER UNE PAUSE DE 5 SECONDES.

EN EFFET UNE UNITE CORRESPOND A LA MILLISECONDE.

SWAP:

PERMET L'ECHANGE DES VALEURS DES VARIABLES.

SWAP X,Y DONNE A X LA VALEUR DE Y ET VICE-VERSA.

AUTO:

PERMET DE NUMEROTER AUTOMATIQUEMENT VOS LIGNES DE PROGRAMME.

AUTO PRODUIT UN NUMEROTAGE DE 10 EN 10 A PARTIR DE 100

AUTO 50,5 PRODUIT UN NUMEROTAGE DE 5 EN 5 A PARTIR DE 50.

ON SORT DE CE MODE EN APPUYANT SUR BREAK ET ON VALIDE LA LIGNE PAR ENTER.