



## Programma Scrittore

```

1 PRINT"(CLR) "
5 POKES3281,0:POKES3280,0:POKES5,0:POKES6,160:DIM L(400)
6 DIM G(400),C$(400),B(400)
7 INPUT "(RED) (DOWN) NOME (WHIT) ":I$
10 OPEN "1.8.0.18.Z8-CHR$(0)
11 PRINT"(CLR) (DOWN) (RED) FLOW CHART BY ACCOMAZZI VITTORIO FOR MC"
12 PRINT"(RED)
20 GET#1,AS:GET#1,BS:PRINT"(DOWN) (DOWN) (DOWN) PROGRAMMA (RED) ==>(WHIT) ":I$
30 P=P+1:GET#1,OS:GET#1,AS:GET#1,BS:L(P)=256*ASC(BS+Z8)+ASC(AS+Z8)
31 PRINT"(HOME) (DOWN) (DOWN) (DOWN) (DOWN) (DOWN) (DOWN) (DOWN) (DOWN) (DOWN) (RED) STO LE
GGENDO":(WHIT)":L(P),":UL=0
32 NC=NC-1:IFXC>0THEN(XC-NC)=L(P):IFNC<>0 THEN 32
33 NC=0:XC=0
39 GET#1,AS:X=ASC(AS+Z8)
40 IF X=0 THEN 30
50 IF X=139 THEN GOSUB500:UL=UL+1:UL:GOTO 110
51 IF X=144 THEN GOSUB500:UL=UL+1:CS(P)=" STOP ":G(P)=-1:CS(P+1)="V"
52 IF X=138 THEN GOSUB500:UL=UL+1:CS(P)=" RUN ":G(P)=-1:CS(P+1)="V"
53 IF X=128 THEN GOSUB500:UL=UL+1:CS(P)=" END ":G(P)=-1:CS(P+1)="V"
54 IF X=162 THEN GOSUB500:UL=UL+1:CS(P)=" NEW ":G(P)=-1:CS(P+1)="V"
55 IF X=145 THEN GOSUB500:UL=UL+1:CS(P)="V":GOTO 161
60 IF X=141 THEN GOSUB500:UL=UL+1:CS(P)="GOSUB":GOTO80:REM GOSUB
61 IF X=142 THEN GOSUB 500:UL=UL+1:UL:CS(P+1)="V":CS(P)="RETURN":REM RETURN
62 IF X=129 THEN GOSUB 500:UL=UL+1:UL:G(P)=-1:CS(P)="FOR":IFCS(P+1)="V"
63 IF X=130 THEN GOSUB500:UL=UL+1:UL:GOTO 175:REM NEXT
70 IF X=137 THEN GOSUB500:UL=UL+1:UL:GOTO 80:REM GOTO
71 IF ST=66 THEN CLOSE1:GOTO200
72 GOTO39
80 GET#1,AS:X=ASC(AS+Z8):IF X<48 OR X>57 THEN 80
81 GS=GS+AS
82 IF CS(P+1)=" THEN CS(P+1)="V"
90 GET#1,AS:X=ASC(AS+Z8):IF X<48 OR X>57 THEN G(P)=VAL(G$):G$="":GOTO40
100 GS=GS+AS:GOTO90
110:REM IF
111 CS(P)="":IF CS(P+1)=" THEN CS(P+1)="V"
113 GET#1,AS:X=ASC(AS+Z8):IFX=167THEN130
114 IF AS=" THEN 113
129 GOSUB700:CS(P)=CS(P)+AS:GOTO113
130 GET#1,AS:X=ASC(AS+Z8):IFX=32 THEN 130
131 IF X<48 OR X>57 THEN 40
140 GS=GS+AS
150 GET#1,AS:X=ASC(AS+Z8):IF X<48 OR X>57 THEN G(P)=VAL(G$):G$="":GOTO40
160 GS=GS+AS:GOTO150
161:REM ON
162 VS="":GET#1,AS:X=ASC(AS+Z8):IFAS=" THEN 162
163 IFAS<> THEN GOSUB700:VS=VS+AS
164 GET#1,AS:X=ASC(AS+Z8):IFX>137 AND X<141 THEN 163
165 IFX=137 THENI$="GOTO"
166 IFX=141 THENI$="GOSUB"
167 CS(P)="ON "+VS+" "+I$:GOSUB500
168 GET#1,AS:X=ASC(AS+Z8):IFAS=" THEN 168
170 IFAS=" THENG(P)=VAL(G$):GS="":Z=Z+1:CS(P)=VS+"+STR$(Z):GOSUB500:GOTO168
171 IFX<48ORX>57THENG(P)=VAL(G$):GS="":Z=Z+1:CS(P)=VS+"+STR$(Z):GOTO40
172 GS=GS+AS:CS(P+1)="V":GOTO168
175:REM NEXT
176:REM
177 CS(P+1)="V":GET#1,AS:X=ASC(AS+Z8):IFX=32THEN176
179 IF AS=" THEN 187
180 IF X=0 THEN 187
181 IF AS=" THEN 187
185 CS=CS+AS:CS(P+1)="V":GOTO176
187 FOR=PTO1STEP-1:IFG(O)=-1ANDCS(O)=""FOR THEN GOSUB195:GOTO189
188 NEXT:CS="":GOTO40
189 IF AS=" THEN 191
190 GOTO40
191 GOSUB 500:GET#1,AS
193 FOR=OTO1STEP-1:IFG(K)=-1ANDCS(K)=""FOR THEN GOSUB 196:GOTO40
194 NEXTK:CS="":GOTO40
195 CS(O)="+STR$(UL):CS(P)="+STR$(UL):G(O)=G(P)=L(O):RETURN
196 CS(K)="+STR$(UL):CS(P)="+STR$(UL):G(K)=G(P)=L(K):RETURN
200:REM
201 PRINT"(CLR) (DOWN) (DOWN) (DOWN) (WHIT)
202 GET AS:IF AS="" THEN 202
203 OPEN "8.8.8.80:DATI.S.W"
204 PRINT"(CLR) (DOWN) (DOWN) (DOWN) (WHIT)
ATTENDERE PREGO
220 P=P-1
230 FOR T=1TOP:IFG(T)="" THEN 260
260 NEXTT
265 NEXTL
300:REM BLOCCHI
301 REM
302 PP=0:IFCS(1)="" THEN PP=1
310 FOR=1TOP:IF CS(F)="" AND LEN(NS)<40 THEN321
311 IF NS<>"" THEN PRINT#8,NS
312 NS=STR$(L(F))
313 IFCS(F)<>"V"AND CS(F)<>"AND MIDS(C$(F),1,1)<"*THENS=STR$(L(F))+"+CS(F)
314 IFMIDS(C$(F),1,1)=""*ANDG(F)=OTHERNS=STR$(L(F))+"+*FOR"
315 IFMIDS(C$(F),1,1)=""*ANDG(F)<OTHERNS=STR$(L(F))+"+*NEXT"
316 PP=PP+1:IF PP>173 THEN CLOSE8:PRINT"(CLR) (DOWN) (DOWN) PROGRAMMA TROPPO
LUNGO":END
320 B(F)=PP:GOTO340
321 CS=STR$(L(F))
322 IFCS(F)<>"V"ANDCS(F)<>"ANDMIDS(C$(F),1,1)<"*THENS=STR$(L(F))+"+CS(F)
323 IFMIDS(C$(F),1,1)=""*ANDG(F)=OTHERNS=STR$(L(F))+"+*FOR"
324 IFMIDS(C$(F),1,1)=""*ANDG(F)<OTHERNS=STR$(L(F))+"+*NEXT"
330 NS=NS+CS:B(F)=PP
340 NEXT F:PRINT#8,NS:PRINT#8,"**FINE**"
351:REM
352:REM
353:REM
354:REM
355:REM
356:REM
357:REM
358:REM
359:REM
360 FOR=1TOP:IFG(F)=GTHEN410
360 FOR=1TOP:IFG(F)=L(G)ANDFC>GTHEN391
391 IF MIDS(C$(F),1,1)=""*AND CS(F)<>CS(G) THEN 390
400 PRINT#8,B(F):PRINT#8,B(G)
410 NEXT F:PRINT#8,Z55:PRINT#8,255
420:REM
430:REM
431:REM
432:REM
433:REM
434:REM
435:REM
436:REM
437:REM
438:REM
439:REM
440:REM
441:REM
442:REM
443:REM
444:REM
445:REM
446:REM
447:REM
448:REM
449:REM
450:REM
451:REM
452:REM
453:REM
454:REM
455:REM
456:REM
457:REM
458:REM
459:REM
460:REM
461:REM
462:REM
463:REM
464:REM
465:REM
466:REM
467:REM
468:REM
469:REM
470:REM
471:REM
472:REM
473:REM
474:REM
475:REM
476:REM
477:REM
478:REM
479:REM
480:REM
481:REM
482:REM
483:REM
484:REM
485:REM
486:REM
487:REM
488:REM
489:REM
490:REM
491:REM
492:REM
493:REM
494:REM
495:REM
496:REM
497:REM
498:REM
499:REM
500:REM
501:REM
502:REM
503:REM
504:REM
505:REM
506:REM
507:REM
508:REM
509:REM
510:REM
511:REM
512:REM
513:REM
514:REM
515:REM
516:REM
517:REM
518:REM
519:REM
520:REM
521:REM
522:REM
523:REM
524:REM
525:REM
526:REM
527:REM
528:REM
529:REM
530:REM
531:REM
532:REM
533:REM
534:REM
535:REM
536:REM
537:REM
538:REM
539:REM
540:REM
541:REM
542:REM
543:REM
544:REM
545:REM
546:REM
547:REM
548:REM
549:REM
550:REM
551:REM
552:REM
553:REM
554:REM
555:REM
556:REM
557:REM
558:REM
559:REM
560:REM
561:REM
562:REM
563:REM
564:REM
565:REM
566:REM
567:REM
568:REM
569:REM
570:REM
571:REM
572:REM
573:REM
574:REM
575:REM
576:REM
577:REM
578:REM
579:REM
580:REM
581:REM
582:REM
583:REM
584:REM
585:REM
586:REM
587:REM
588:REM
589:REM
590:REM
591:REM
592:REM
593:REM
594:REM
595:REM
596:REM
597:REM
598:REM
599:REM
600:REM
601:REM
602:REM
603:REM
604:REM
605:REM
606:REM
607:REM
608:REM
609:REM
610:REM
611:REM
612:REM
613:REM
614:REM
615:REM
616:REM
617:REM
618:REM
619:REM
620:REM
621:REM
622:REM
623:REM
624:REM
625:REM
626:REM
627:REM
628:REM
629:REM
630:REM
631:REM
632:REM
633:REM
634:REM
635:REM
636:REM
637:REM
638:REM
639:REM
640:REM
641:REM
642:REM
643:REM
644:REM
645:REM
646:REM
647:REM
648:REM
649:REM
650:REM
651:REM
652:REM
653:REM
654:REM
655:REM
656:REM
657:REM
658:REM
659:REM
660:REM
661:REM
662:REM
663:REM
664:REM
665:REM
666:REM
667:REM
668:REM
669:REM
670:REM
671:REM
672:REM
673:REM
674:REM
675:REM
676:REM
677:REM
678:REM
679:REM
680:REM
681:REM
682:REM
683:REM
684:REM
685:REM
686:REM
687:REM
688:REM
689:REM
690:REM
691:REM
692:REM
693:REM
694:REM
695:REM
696:REM
697:REM
698:REM
699:REM
700:REM
701:REM
702:REM
703:REM
704:REM
705:REM
706:REM
707:REM
708:REM
709:REM
710:REM
711:REM
712:REM
713:REM
714:REM
715:REM
716:REM
717:REM
718:REM
719:REM
720:REM
721:REM
722:REM
723:REM
724:REM
725:REM
726:REM
727:REM
728:REM
729:REM
730:REM
731:REM
732:REM
733:REM
734:REM
735:REM
736:REM
737:REM
738:REM
739:REM
740:REM
741:REM
742:REM
743:REM
744:REM
745:REM
746:REM
747:REM
748:REM
749:REM
750:REM
751:REM
752:REM
753:REM
754:REM
755:REM
756:REM
757:REM
758:REM
759:REM
760:REM
761:REM
762:REM
763:REM
764:REM
765:REM
766:REM
767:REM
768:REM
769:REM
770:REM
771:REM
772:REM
773:REM
774:REM
775:REM
776:REM
777:REM
778:REM
779:REM
780:REM
781:REM
782:REM
783:REM
784:REM
785:REM
786:REM
787:REM
788:REM
789:REM
790:REM
791:REM
792:REM
793:REM
794:REM
795:REM
796:REM
797:REM
798:REM
799:REM
800:REM
801:REM
802:REM
803:REM
804:REM
805:REM
806:REM
807:REM
808:REM
809:REM
810:REM
811:REM
812:REM
813:REM
814:REM
815:REM
816:REM
817:REM
818:REM
819:REM
820:REM
821:REM
822:REM
823:REM
824:REM
825:REM
826:REM
827:REM
828:REM
829:REM
830:REM
831:REM
832:REM
833:REM
834:REM
835:REM
836:REM
837:REM
838:REM
839:REM
840:REM
841:REM
842:REM
843:REM
844:REM
845:REM
846:REM
847:REM
848:REM
849:REM
850:REM
851:REM
852:REM
853:REM
854:REM
855:REM
856:REM
857:REM
858:REM
859:REM
860:REM
861:REM
862:REM
863:REM
864:REM
865:REM
866:REM
867:REM
868:REM
869:REM
870:REM
871:REM
872:REM
873:REM
874:REM
875:REM
876:REM
877:REM
878:REM
879:REM
880:REM
881:REM
882:REM
883:REM
884:REM
885:REM
886:REM
887:REM
888:REM
889:REM
890:REM
891:REM
892:REM
893:REM
894:REM
895:REM
896:REM
897:REM
898:REM
899:REM
900:REM
901:REM
902:REM
903:REM
904:REM
905:REM
906:REM
907:REM
908:REM
909:REM
910:REM
911:REM
912:REM
913:REM
914:REM
915:REM
916:REM
917:REM
918:REM
919:REM
920:REM
921:REM
922:REM
923:REM
924:REM
925:REM
926:REM
927:REM
928:REM
929:REM
930:REM
931:REM
932:REM
933:REM
934:REM
935:REM
936:REM
937:REM
938:REM
939:REM
940:REM
941:REM
942:REM
943:REM
944:REM
945:REM
946:REM
947:REM
948:REM
949:REM
950:REM
951:REM
952:REM
953:REM
954:REM
955:REM
956:REM
957:REM
958:REM
959:REM
960:REM
961:REM
962:REM
963:REM
964:REM
965:REM
966:REM
967:REM
968:REM
969:REM
970:REM
971:REM
972:REM
973:REM
974:REM
975:REM
976:REM
977:REM
978:REM
979:REM
980:REM
981:REM
982:REM
983:REM
984:REM
985:REM
986:REM
987:REM
988:REM
989:REM
990:REM
991:REM
992:REM
993:REM
994:REM
995:REM
996:REM
997:REM
998:REM
999:REM
1000:REM

```

## Programma Installatore

```

5 PRINT"(CLR)(RED)FLOW CHART BY ACCOMAZZI VIT
TORIO FOR MC":POKE53281,0:POKE53280,0
6 PRINT"(DOWN)(DOWN)(DOWN)(DOWN)(DOWN)(DOWN)(
DOWN)(DOWN)(WHT) MI STO INSTALLAN
DO "
7 OPEN 8.8.8,"@0:L.M..P.W":PRINT#8,CHR$(0):CH
R$(192):
10 FORT=49152 TO 49731 :READA:PRINT#8,CHR$(A)
::NEXT
20 FORT=49732 TO 49919 :PRINT#8,CHR$(0)::NEXT
30 FORT=49920 TO 50187 :READA:PRINT#8,CHR$(A)
)::NEXT T:CLOSE8:END
10001 DATA 32 , 179 , 192 , 32 , 129
10002 DATA 192 , 32 , 106 , 192 , 32
10003 DATA 169 , 192 , 169 , 80 , 133
10004 DATA 2 , 166 , 20 , 208 , 3
10005 DATA 76 , 35 , 192 , 169 , 194
10006 DATA 160 , 0 , 145 , 251 , 32
10007 DATA 117 , 192 , 202 , 208 , 244
10008 DATA 165 , 21 , 208 , 3 , 76
10009 DATA 128 , 192 , 162 , 0 , 169
10010 DATA 194 , 160 , 0 , 145 , 251
10011 DATA 32 , 117 , 192 , 202 , 208
10012 DATA 244 , 198 , 21 , 208 , 240
10013 DATA 96 , 32 , 179 , 192 , 32
10014 DATA 129 , 192 , 32 , 106 , 192
10015 DATA 32 , 169 , 192 , 160 , 0
10016 DATA 169 , 195 , 145 , 251 , 200
10017 DATA 196 , 20 , 208 , 249 , 96
10018 DATA 32 , 179 , 192 , 169 , 32
10019 DATA 160 , 0 , 145 , 251 , 200
10020 DATA 208 , 251 , 230 , 252 , 166
10021 DATA 252 , 224 , 160 , 208 , 243
10022 DATA 96 , 32 , 169 , 192 , 165
10023 DATA 20 , 133 , 2 , 32 , 117
10024 DATA 192 , 96 , 165 , 251 , 24
10025 DATA 101 , 2 , 133 , 251 , 144
10026 DATA 2 , 230 , 252 , 96 , 32
10027 DATA 169 , 192 , 169 , 80 , 133
10028 DATA 2 , 166 , 20 , 208 , 3
10029 DATA 76 , 149 , 192 , 32 , 117
10030 DATA 192 , 202 , 208 , 250 , 165
10031 DATA 21 , 208 , 3 , 76 , 128
10032 DATA 192 , 162 , 0 , 32 , 117
10033 DATA 192 , 202 , 208 , 250 , 198
10034 DATA 21 , 208 , 244 , 96 , 32
10035 DATA 253 , 174 , 32 , 138 , 173
10036 DATA 32 , 247 , 183 , 96 , 169
10037 DATA 48 , 133 , 252 , 169 , 0
10038 DATA 133 , 251 , 96 , 32 , 179
10039 DATA 192 , 32 , 129 , 192 , 32
10040 DATA 106 , 192 , 32 , 169 , 192
10041 DATA 169 , 80 , 133 , 2 , 166
10042 DATA 20 , 208 , 3 , 76 , 225
10043 DATA 192 , 160 , 0 , 177 , 251
10044 DATA 201 , 195 , 208 , 42 , 32
10045 DATA 117 , 192 , 202 , 208 , 242
10046 DATA 165 , 21 , 208 , 7 , 169
10047 DATA 0 , 133 , 2 , 76 , 128
10048 DATA 192 , 162 , 0 , 160 , 0
10049 DATA 177 , 251 , 201 , 195 , 208
10050 DATA 25 , 32 , 117 , 192 , 202
10051 DATA 208 , 242 , 198 , 21 , 208
10052 DATA 238 , 169 , 0 , 133 , 2
10053 DATA 96 , 201 , 32 , 208 , 3
10054 DATA 76 , 219 , 192 , 133 , 2
10055 DATA 96 , 201 , 32 , 208 , 3
10056 DATA 76 , 246 , 192 , 133 , 2
10057 DATA 96 , 32 , 179 , 192 , 32
10058 DATA 129 , 192 , 162 , 4 , 32
10059 DATA 201 , 255 , 160 , 0 , 177
10060 DATA 251 , 32 , 210 , 255 , 200
10061 DATA 192 , 80 , 208 , 246 , 96
10062 DATA 0 , 133 , 251 , 32 , 253
10063 DATA 174 , 32 , 138 , 173 , 32
10064 DATA 247 , 183 , 166 , 20 , 169
10065 DATA 80 , 133 , 2 , 32 , 77
10066 DATA 192 , 202 , 224 , 0 , 208
10067 DATA 248 , 32 , 253 , 174 , 32
10068 DATA 138 , 173 , 32 , 247 , 183
10069 DATA 165 , 20 , 133 , 2 , 32
10070 DATA 77 , 192 , 32 , 253 , 174
10071 DATA 32 , 138 , 173 , 32 , 247
10072 DATA 183 , 169 , 80 , 133 , 2
10073 DATA 166 , 20 , 202 , 160 , 0
10074 DATA 177 , 251 , 201 , 194 , 208
10075 DATA 5 , 169 , 255 , 133 , 2
10076 DATA 96 , 32 , 77 , 192 , 224
10077 DATA 0 , 208 , 235 , 169 , 0
10078 DATA 133 , 2 , 96 , 169 , 2
10079 DATA 141 , 255 , 207 , 141 , 254

```

## Nota

I codici di controllo nei listati sono riportati in forma «esplicita», in conseguenza dell'impiego della stampante Star NL-10 e relativa interfaccia per Commodore. Ovviamente, nella digitazione del programma è necessario usare i consueti tasti che corrispondono alle indicazioni fra parentesi: ad esempio cursore destro per (RGHT), CTRL-3 per (RED) eccetera.

```

(CLR) = ␣
(HOME) = ␣
(DOWN) = ␣
(UP) = ␣
(RGHT) = ␣
(LEFT) = ␣
(BLK) = ␣
(WHT) = ␣
(RED) = ␣
(CYN) = ␣
(PUR) = ␣
(GRN) = ␣
(BLU) = ␣
(YEL) = ␣
(RVS) = ␣
(OFF) = ␣
(ORNG) = ␣
(BRN) = ␣
(LRED) = ␣
(GRY1) = ␣
(GRY2) = ␣
(LGRN) = ␣
(LBLU) = ␣
(GRY3) = ␣
(SWLC) = ␣

```

```

10080 DATA 207 , 173 , 0 , 220 , 201
10081 DATA 126 , 208 , 15 , 206 , 255
10082 DATA 207 , 173 , 255 , 207 , 201
10083 DATA 1 , 208 , 67 , 169 , 155
10084 DATA 141 , 255 , 207 , 201 , 125
10085 DATA 208 , 15 , 238 , 255 , 207
10086 DATA 173 , 255 , 207 , 201 , 155
10087 DATA 208 , 48 , 169 , 1 , 141
10088 DATA 255 , 207 , 201 , 119 , 208
10089 DATA 15 , 238 , 254 , 207 , 173
10090 DATA 254 , 207 , 201 , 41 , 208
10091 DATA 29 , 169 , 0 , 141 , 254
10092 DATA 207 , 201 , 123 , 208 , 15
10093 DATA 206 , 254 , 207 , 173 , 254
10094 DATA 207 , 201 , 0 , 208 , 10
10095 DATA 169 , 41 , 141 , 254 , 207
10096 DATA 201 , 111 , 208 , 173 , 96
10097 DATA 32 , 237 , 192 , 76 , 140
10098 DATA 193 , 120 , 169 , 243 , 141
10099 DATA 20 , 3 , 169 , 193 , 141
10100 DATA 21 , 3 , 88 , 96 , 238
10101 DATA 241 , 207 , 173 , 241 , 207
10102 DATA 201 , 9 , 208 , 42 , 169
10103 DATA 0 , 141 , 241 , 207 , 169
10104 DATA 192 , 133 , 253 , 169 , 7
10105 DATA 133 , 254 , 160 , 0 , 177
10106 DATA 253 , 141 , 240 , 207 , 200
10107 DATA 177 , 253 , 136 , 145 , 253
10108 DATA 200 , 192 , 39 , 208 , 245
10109 DATA 173 , 240 , 207 , 145 , 253
10110 DATA 165 , 254 , 201 , 219 , 208
10111 DATA 3 , 76 , 49 , 234 , 169
10112 DATA 219 , 133 , 254 , 76 , 10
10113 DATA 194 , 24 , 160 , 0 , 162
10114 DATA 0 , 32 , 240 , 255 , 96
10115 DATA 114 , 254 , 0 , 158 , 0
10116 DATA 191 , 0 , 255 , 0 , 255
20001 DATA 169 , 0 , 141 , 255 , 207
20002 DATA 32 , 198 , 195 , 234 , 173
20003 DATA 0 , 220 , 201 , 126 , 208
20004 DATA 13 , 169 , 80 , 133 , 2
20005 DATA 32 , 186 , 195 , 32 , 237
20006 DATA 195 , 76 , 129 , 195 , 201
20007 DATA 125 , 208 , 13 , 169 , 80
20008 DATA 133 , 2 , 32 , 174 , 195
20009 DATA 32 , 207 , 195 , 76 , 129
20010 DATA 195 , 201 , 119 , 208 , 35
20011 DATA 238 , 255 , 207 , 173 , 255
20012 DATA 207 , 201 , 41 , 208 , 15
20013 DATA 169 , 0 , 141 , 255 , 207
20014 DATA 169 , 40 , 133 , 2 , 32
20015 DATA 186 , 195 , 76 , 129 , 195
20016 DATA 169 , 1 , 133 , 2 , 32
20017 DATA 174 , 195 , 76 , 129 , 195
20018 DATA 201 , 123 , 208 , 35 , 206
20019 DATA 255 , 207 , 173 , 255 , 207
20020 DATA 201 , 255 , 208 , 15 , 169
20021 DATA 40 , 141 , 255 , 207 , 169
20022 DATA 40 , 133 , 2 , 32 , 174
20023 DATA 195 , 76 , 129 , 195 , 169
20024 DATA 1 , 133 , 2 , 32 , 186
20025 DATA 195 , 76 , 129 , 195 , 201
20026 DATA 111 , 208 , 137 , 96 , 24
20027 DATA 162 , 0 , 160 , 0 , 32
20028 DATA 240 , 255 , 169 , 80 , 133
20029 DATA 2 , 162 , 0 , 160 , 0
20030 DATA 177 , 251 , 32 , 210 , 255
20031 DATA 200 , 192 , 40 , 208 , 246
20032 DATA 32 , 174 , 195 , 232 , 224
20033 DATA 24 , 208 , 236 , 162 , 24
20034 DATA 32 , 186 , 195 , 202 , 208
20035 DATA 250 , 76 , 9 , 195 , 165
20036 DATA 251 , 24 , 101 , 2 , 133
20037 DATA 251 , 144 , 14 , 230 , 252
20038 DATA 96 , 165 , 251 , 56 , 229
20039 DATA 2 , 133 , 251 , 176 , 2
20040 DATA 198 , 252 , 96 , 169 , 48
20041 DATA 133 , 252 , 169 , 0 , 133
20042 DATA 251 , 96 , 165 , 252 , 56
20043 DATA 201 , 150 , 144 , 239 , 165
20044 DATA 251 , 201 , 48 , 144 , 233
20045 DATA 32 , 198 , 195 , 174 , 255
20046 DATA 207 , 169 , 1 , 133 , 2
20047 DATA 32 , 174 , 195 , 202 , 208
20048 DATA 250 , 96 , 24 , 165 , 252
20049 DATA 201 , 48 , 176 , 209 , 169
20050 DATA 150 , 133 , 252 , 169 , 48
20051 DATA 133 , 251 , 174 , 255 , 207
20052 DATA 169 , 1 , 133 , 2 , 32
20053 DATA 174 , 195 , 202 , 208 , 250
20054 DATA 96 , 0 , 255

```

traccerà in memoria il grafico che è possibile esplorare usando il joystick porta 2.

Decrizione dei programmi:

**Installatore:** apre un file programma (linea 7) e scrive all'interno di questo i dati contenuti nelle linee 10001-20054 con tre loop per rilocarli nelle apposite locazioni.

**Scrittore:** variabili: G(400) linea richiamata; C\$(400) messaggio del token B(400) blocco dove risiede la linea corrispondente; L(400) contiene tutti i numeri di linea.

Dimensiona le matrici e chiede il nome del programma da leggere (linea 7) in seguito apre il file prende il byte alto e basso di locazione del file (linea 20), il puntatore di linea e il byte alto e basso della linea (linea 31). Poi inizia a prendere i token Basic, nelle linee 50-70 confronta se sono quelli da ricercare e, per ogni valore, modifica le apposte variabili. Non mi inoltro nella descrizione di ogni routine, per ogni token, perché sarebbe troppo lungo e noioso, mi limito a evidenziare le parti più importanti.

**220-260** Se una linea richiama un'altra essa deve stare all'inizio di un blocchetto

(con tale termine intendo varie linee collegate tra loro sequenzialmente che non vengono mai eseguite separatamente, visivamente esse si troveranno su una stessa linea del diagramma separate da uno spazio).

**302-330** Concatena le varie linee in blocchetti e mette il valore nella matrice B (400).

**360-410** Ricerca e scrive sul file le varie connessioni tra i blocchetti.

**450-470** Crea le varie connessioni sequenziali tra i programmi.

**501-520** Ristabilisce le variabili quando una linea ha più di un comando che influisce sul diagramma di flusso.

**700-1000** Riconosce i vari operatori di stringa e matematici.

### Letto

**Variabili:** X1(175) e X2(175) spazio del blocchetto dal margine destro e sinistro; g (100) elenco dei blocchetti che chiamano un medesimo blocchetto; q1 (175) e q2(175) blocchetti che si richiamano.

Questo programma legge le routine in l.m nella linea 1, dimensiona le variabili e inizia a leggere (205-220) le stringhe che

dovranno essere all'interno del blocchetto. Alla linea 230-240 legge tutti i richiami che vi sono tra i blocchetti, nelle linee 250-270 legge i richiami sequenziali e li traccia. Di seguito, il programma traccia tutti i collegamenti non sequenziali e procede come segue:

1) **320-360** per ogni blocchetto, cerca da chi è chiamato e lo mette nel vettore g (100);

2) **15101-15106** calcola il più alto e il più basso blocchetto che si dovrà collegare con la linea verticale;

3) **15132-15234** traccia le linee; in fine permette uno scrolling con una routine completamente in l.m e chiede se si vuole stampare; routine utilizzate in linguaggio macchina.

**49237** pulisce la memoria.

**49213** colonna, riga, lunghezza: crea una linea orizzontale.

**49152** colonna, riga, lunghezza: crea una linea verticale.

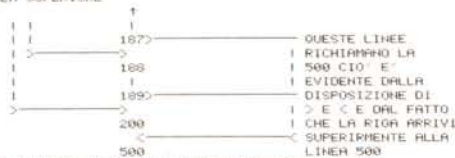
**49340** colonna, riga, lunghezza: controlla se è possibile tracciare una linea verticale e, se sì, unisce alle altre.

**49433** riga, stampa la riga.

**49920** scrolling di tutta la pagina, premere fire per uscire.

### Come si legge l'output generato da Flow chart.

QUESTO TRATTINO INDICA CHE SI ARRIVA SEQUENZIALMENTE DALLA LINEA SUPERIORE



NON VI E' ALCUNA DIFFERENZA SE LA RIGA E' DESTRA O SINISTRA DELLA LINEA INTERESSATA

ESEMPIO DI BLOCCHETTO :

1 5 6 7 10 11 12 20

< INDICA CHE LE LINEE SONO SEMPRE ESEGUITE SEQUENZIALMENTE >

CON GLI IF THEN VI POSSONO ESSERE 3 CASI :

1 510 : UL=0  
< INDICA CHE NELLA LINEA 510 VI E' IF UL=0 THEN LINEA\_INDICATA, SE LA CONDIZIONE NON SI E' VERIFICATA PROCEDE SEQUENZIALMENTE >

2 502 : UC=139 AND UL>0  
< INDICA CHE DOPO IL THEN NON VI E' UN COMANDO CHE ALTERA IL DIAGRAMMA DI FLUSSO, PUO' ESSERCI PERCIO' UN CALCOLO O UN PRINT >

3 316 : PP>173  
316 : END  
320  
< INDICA CHE DOPO IL THEN VI E' UN COMANDO CHE ALTERA IL DIAGRAMMA DI FLUSSO IN QUESTO CASO SE E' VERA LA CONDIZIONE SI LEGGE IL COMANDO CON LO STESSO NUMERO DI LINEA SE E' FALSA SI SEGUE LA RIGA >

IL COMANDO GOTO VIENE RAPPRESENTATO COME SEGUE

55 :>

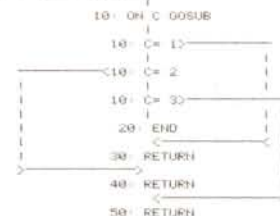
INDICA IN CHE ALLA LINEA 55 VI E' UN SALTO INCONDIZIONATO, LO SI PUO' INCONTRARE ANCHE ALLA FINE DI UN BLOCCHETTO ESEMPIO :

40 50 55 :>

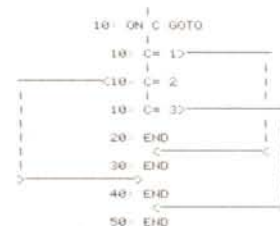
OPPURE



IL COMANDO ON GOSUB E' RAPPRESENTATO :



SE LE RIGHE PARTISSERO TUTTE DALLO STESSO BLOCCHETTO NON AVREBBE VISIVAMENTE LA CORRISPONDENZA VALORE-LINEA. IL COMANDO ON GOTO DIFFERISCE DAL PRECEDENTE SOLO PER NON AVERE IL COLLEGAMENTO SEQUENZIALE DOPO LA SERIE DI BLOCCHETTI DELLA STESSA LINEA ESEMPIO :



IL METODO CON CUI HO SCHEMATIZZATO QUESTI COMANDI E' STATO DETERMINATO DALLA PRATICITA' E DALLA MASSIMA SEMPLIFICAZIONE. FLOW CHART INOLTRE EVIDENZIA I COMANDI CHE PORTANO AD UNA INTERRUZIONE FORZATA, I FOR NEXT E I GOSUB

# Flower's love

di Daniele Noris - Bergamo

Questo è un programma adattato al Commodore 64 da una precedente ver-

sione per l'Apple IIe. Lo scopo del programma è quello di spiegare il significato dei fiori, cosa che può essere molto utile quando si vuole dimostrare i propri sentimenti verso gli altri (amore, odio). Il programma è dotato di un menu principale, dove noi potremo scegliere varie opzioni:

1) nel caso in cui hai ricevuto dei fiori, devi introdurre che tipo di fiore hai ricevuto, e saprai subito il significato del

fiore (o quello che voleva dirti chi li ha spediti).

2) Nel caso in cui devi spedire dei fiori, devi dire cosa provi ed il computer te lo dirà.

3) Altro caso: puoi visionare l'elenco completo dei fiori contenuti nell'archivio.

Il programma è semplice e di facile comprensione perciò altri commenti non sono necessari.



## Flow Chart

```

0 rem *****
1 rem ** programma tradotto da : **
2 rem ** **
3 rem ** daniele noris **
4 rem ** **
5 rem *****
6 rem * versione originale; apple II *
7 rem * di m. magnani f. veronese *
8 rem *****
9 rem * 23 luglio 1985 *
10 poke$3280,0;poke$3281,0;print"(clr)(yel)",chr$(14)
11 print"(rva)(down)(down)(right)(right)(right)(right)(right)(right)(right)(right)
(rght)(rght) BITELD CON I FIORI (off)"
12 print"(down)(down)(right)(right)(right)(right)(right)(right)(right)(right)(rgrn)
(l:red)O (orange)W (ired)E (brn)R (wht)S (gry)3L (blu)O (lgrn)V (wht)E"
13 print"(gry)3(down)(down)(down)(right)(right)(right)(right)(right)(right)(right)
(programma tradotto da:"print"(down)(right)(right)(right)(right)(right)(right)(rgh)
(rght)(grn)Daniele Noris"
14 print"(gry)3(down)(right)(right)(right)(right)(right)(right)(right)(right)(right)
(int)(down)(right)(right)(right)(right)di: M.Magnani e F.Veronese
15 print"(blu)(down)(down)(down)(right)(right)(right)(right)(right)(right)(right)
(rght)(rght)23 luglio 1985"
16 print"(down)(down)(right)(right)(right)(right)(right)(right)(orange)Premi la barra per cont
inuare"poke198,0;wait198,1
20 print"(clr)(yel)(down)(down)(right)(right)(right)(right)(right)(right)(right)(right)
(right)(right)(right)(right)*** M E N U ***"
30 printtab(4)"(down)(down)(down)(blu)(rva) 1 (off) --- (gry)3Hai ricevuto un f
iore"
40 printtab(4)"(down)(down)(blu)(rva) 2 (off) --- (gry)3Devi spedire un fiore"
50 printtab(4)"(down)(down)(blu)(rva) 3 (off) --- (gry)3Elenco fiori"
60 printtab(4)"(down)(down)(blu)(rva) 4 (off) --- (gry)3Elenco messaggi"
70 printtab(4)"(down)(down)(blu)(rva) 5 (off) --- (gry)3Fine"
80 input"(down)(down)(right)(right)(right)(right)(yel)Quale ----> (gry)3)a
120 onsgoto130,370,270,490,1060
130 input"(clr)(down)(down)(down)(down)(gry)3Che fiore hai ricevuto ?":f$
140 restore
150 reada$,b$
160 ifb$="end"thengoto250
170 ifb$<>i$then150
180 print:print
190 printf$(rght)(rght)Significa":print"(down)(right)(right)(yel)<":a$;">"
200 reada$,b$
210 ifb$="end"then350
220 ifb$<>i$then200
230 print"(down)(right)(right)(gry)3o anche(right)(yel) <":a$;">"
240 goto200
250 print"(gry)3(down)Nessun fiore con quel nome":input"Vuoi vedere la lista dei
fiori ?":i$
260 ifi$="n"then20
270 print"(clr)"
280 restore
290 i=i+1
300 reada$,b$
305 fora=0to300:nexta
310 ifi=5ori=7thengoto340
312 ifi=8ori=10thengoto340
314 ifi=33ori=34thengoto340
316 ifi=37ori=39thengoto340
318 ifi=41ori=45thengoto340
320 ifi=20ori=40thenprint"(down)(down)(right)(right)(gry)3Premi la barra per conti
nuare"poke198,0;wait198,1;print"(clr)"
330 print tab(8);b$
340 i=i+1;if i<48thengoto300
350 print"(down)(right)(right)(gry)3premi la barra per tornare al menu":poke198,0
;wait198,1
360 goto20
370 print"(clr)(down)(down)(gry)3Cosa vuoi dire ":input"(down)con un fiore ?":m$
380 restore
390 reada$,b$
400 ifa$="end"thengoto460
410 ifa$<>m$thengoto390
420 print:print
430 print"(yel)<":m$;">";(gry)3..."print"(down)... ditelo con un fiore del tip
o"
435 print"(down)(down)(rva)(yel)<":b$;">"
440 print"(down)(right)(gry)3Premi la barra per tornare al menu":poke198,0;wait
198,1
450 goto20
460 print"(gry)3(down)Questo messaggio non e' contemplato."
465 input"(gry)3(down)Vuoi vedere i possibili messaggi ?":k$
470 ifleft$(k$,1)="-a"thenprint"(clr)":goto490
480 goto20
490 print"(clr)"
500 restore
510 fori=1to47
520 reada$,b$
525 fora=0to300:nexta
530 ifi=20ori=40thenprint"Un tasto per continuare":poke198,0;wait198,1;print"(cl
r)"
540 print tab(10);a$
550 nexti
560 print"Un tasto qualsiasi per tornare al menu":poke198,0;wait198,1
570 goto20
580 data abbandono.anemone
590 data amicizia.disinteressata.glicine
600 data amore fisico.orchidea
610 data amore piacente.narciso
620 data amore platonico.acacia
630 data sentimento puro.acacia
640 data preferenza.anemone
650 data candore.anemone
660 data speranza.biancospino
670 data connessione.bucaneve
680 data comprensione.bucaneve
690 data dispiacere.calendula
700 data sacrificio.camelia
710 data riconoscenza e affetto.dalia
720 data tenerezza e perenne fedelta'.edera
730 data solitudine.eric
740 data delicatezza.fiordaliso
750 data richiesta di matrimonio.fiori d'arancio
760 data compiacenza.fiori di pesco
770 data fedelta'.garofano bianco
780 data errore.garofano rosso
790 data adegno.garofano giallo
800 data sensazione.garofano rosato
810 data benevolenza.giacinto
820 data sciocchezza.geranio
830 data purezza.giglio
840 data pulizia.ginestra
850 data diffidenza.lavanda
860 data bellezza superba.magnolia
870 data ritorno di felicita'.maghetto
880 data freddezza.ortensia
890 data stordimento.papavero
900 data dubbio.papavero
910 data sorpresa.papavero
920 data credulita' e speranza.primula
930 data silenzio.rosa bianca
940 data infedelta'.rosa gialla
950 data vergogna.rosa gialla
960 data freschezza.rosa rosa
970 data tenerezza.rosa rosa
980 data passione.rosa rossa
990 data innamoramento totale.rosa rossa
1000 data messaggio d'amore.tulipano
1010 data sei sempre nella mia mente.viola del pensiero
1020 data pudore.viola mammola
1030 data modestia.viola mammola
1040 data adegno.viola gialla
1050 data end.end
1055 rem** routine finale **
1060 print"(clr)"
1070 fori=0to10:print.nexti:printtab(8)"(yel)":
1090 h$="a r r i v e d e r c i l"
1100 forf=1to1en(h$):printmid$(h$,f,1):;fort=ito40;nextf
1110 poke198,0;wait198,1;print"(clr)"

```



## Elenco del software disponibile su cassetta o minifloppy

Per ovviare alle difficoltà incontrate da molti lettori nella digitazione dei listati pubblicati nelle varie rubriche di software sulla rivista, MCmicrocomputer mette a disposizione i programmi più significativi direttamente su supporto magnetico. Riepiloghiamo qui sotto i programmi disponibili per le varie macchine, ricordando che i titoli non sono previsti per computer diversi da quelli indicati. Il numero della rivista su cui viene descritto ciascun programma è riportato nell'apposita colonna; consigliamo gli interessati di procurarsi i relativi numeri arretrati, eventualmente rivolgendosi al nostro Servizio Arretrati utilizzando il tagliando pubblicato in fondo alla rivista.

Per l'ordinazione inviare l'importo (a mezzo assegno, c/c o vaglia postale) alla Technimedia srl, Via Carlo Perrier 9, 00157 Roma.

Codice	Titolo programma	MC n.	Prezzo	Codice	Titolo programma	MC n.	Prezzo
<b>APPLE II</b>				<b>COMMODORE 64</b>			
DA2/06	Miniset + Leva-DOS	37	15000	C64/06	Bilancio familiare	35	17000
DA2/07	27 programmi grafici	38	30000	C64/19	Sistema Enalotto	47	17000
DA2/08	Adventure Editor	38	15000	C64/21	RTTY	48	17000
DA2/09	Animazione Funzioni	42	15000	C64/22	Mescola	49	17000
DA2/12	Routine grafiche estese	44	15000	C64/23	Oihello	51	17000
DA2/13	Scroll 300 linee	46	15000	C64/24	Voters	51	17000
DA2/14	Assembler in Basic	50	15000	C64/25	Flashtape	50/51	17000
DA2/15	G-Basic II	53	15000	C64/26	Cross Reference	53	17000
DA2/16	Disk Editor	54	15000	C64/27	Flib	54	17000
DA2/17	Latino	57	15000	C64/28	Boz's Adventure	57	17000
DA2/18	Battaglia	61	15000	D64/11	Anno Domini	57	15000
DA2/19	Catalogo	64	15000	D64/12	The Disk Editor	54/6/7	15000
DA2/20	Apple Puzzle II	65	15000	D64/13	Boz's Adventure	57	15000
DA2/21	Precisione Multipla	66	15000	D64/14	Link-64	57	30000
DA2/22	Sistema 2 + Toto 5.3 IIGS	68	15000	D64/15	New Char 2.2	58	15000
DA2/23	Operazione Apokalypsis	71	30000	D64/16	Music 64	59	15000
DA2/24	Classifiche di Formula 1	72	15000	D64/17	TRX-MEM	59	15000
DA2/25	Programmabile RPN	73	15000	D64/18	WOS + WBasic	60	15000
<b>COMMODORE AMIGA</b>				D64/19	Strange Basic + Dracula	63	15000
DAM/01	F-15	63	15000	D64/20	File Rescue	64	15000
DAM/02	Gest. liste programmi	64	15000	D64/21	La Casa	64	15000
DAM/03	Studio di Funzioni	66	15000	D64/22	Digital Voice	65	15000
DAM/04	Math Pack	68	15000	D64/23	Vita 3D	65	15000
DAM/05	Redcode & Mars (Core Wars)	68	15000	D64/24	Corso di Linguistica	66	15000
DAM/06	Life	69	15000	D64/25	Archiplus	66	15000
DAM/07	Rubrica Telefonica	70	15000	D64/26	Math Pack Plus	66	15000
DAM/08	Piramidi	70	15000	D64/27	Scroll + Multitask + Classifica	67	15000
DAM/09	Regolazione dei colori	71	15000	D64/28	Calend. Perpetuo + Effetto Telecamera	68	15000
DAM/10	Analitica	71	15000	D64/29	Listing Plus + Utility Data	69	15000
DAM/11	Grafici	72	15000	D64/30	TWS Basic	70	15000
DAM/12	Traduttore	73	15000	D64/31	Trucchi e routine per programmatori	71	15000
<b>MS-DOS</b>				D64/32	Flow-Chart + Flower's Love	73	15000
DMS/01	Plotter + Morse	67	15000	<b>MSX</b>			
DMS/02	Melole + Spawn	68	15000	CMX/04	Monitor Disassembler	45	17000
DMS/03	Pretty + Scritte scorrevoli + Compute	69	15000	CMX/05	Video Art	46	17000
DMS/04	Emulatore CGA per Hercules	70	15000	CMX/13	Shape Editor	54	17000
DMS/05	Turbo Directory	71	15000	CMX/14	Labyrinth 3D	55	17000
DMS/06	Math-Tool S	72	15000	CMX/15	Fred	56	17000
DMS/07	Bioritmi + Routine	72	15000	CMX/16	Il tesoro dei pirati	57	17000
DMS/08	Salvavideo + Scritte scorrevoli + PG151	73	15000	CMX/17	Omino	58	17000
<b>COMMODORE 128</b>				CMX/18	Toto 13	60	17000
C28/01	MMCalc	53	17000	CMX/19	Painter	62	17000
C28/02	Mega Bank 128	56	17000	CMX/20	MSX Bank	63	17000
D28/01	MMCalc	53	15000	CMX/21	Grafica 3D + Hard Copy	65	17000
D28/02	Hardcopy 128	55	15000	CMX/22	Mini dBase MSX	71	17000
D28/03	Sheett	57	15000	DMX/01	Toto 13	60	15000
D28/04	Star Quest	58	15000	DMX/02	Painter	62	15000
D28/05	Family Budget	60	15000	DMX/03	MSX Bank	63	15000
D28/06	La casa stregata	61	15000	DMX/04	Grafica 3D + Hard Copy	65	15000
D28/07	Strutture 80/33	63	15000	DMX/05	Easy Disk	66	15000
D28/08	Bas-80 V2.0a	64	15000	DMX/06	Classifiche	67	15000
D28/09	Paint 80 1.0	65	15000	DMX/07	Magic Paint	67	15000
D28/10	Bas-80 V2.11	66	15000	DMX/08	Autogest	68	15000
D28/11	Calendario Perpetuo + Montecarlo	67	15000	DMX/09	Compiatore v. 1.01	69	15000
D28/12	Disegna Circuiti	68	15000	DMX/10	Diskmap	70	15000
D28/13	Mark's Data Base	70	15000	DMX/11	Mini dBase MSX	71	15000
D28/14	Label Disk + Disk-Editor + Dem DOS	71	15000	DMX/12	Grafica in Turbo Pascal	72	15000
D28/15	Pulldown 128/HR + Menu' + Drawer	72	15000	DMX/13	Math Pack Plus 3.20	73	15000
D28/16	Prospettive	73	15000	Nota: l'iniziale del codice e' C per le cassette, D per i floppy.			