

Routine EDIT programmi Basic Applesoft

di Marco Merler - Gardolo (TN)

Una delle maggiori lacune riscontrabili nel Basic Applesoft è senz'altro la mancanza di un buon metodo di EDITING per la correzione degli errori. Correggere una riga diventa più difficile che fare un programma complesso fra copiare, correggere, aggiungere, copiare ancora, cancellare... Per non parlare di quando la riga è molto lunga o contiene delle stringhe o peggio ancora di quando è necessario inserire un comando all'interno di una riga. Molte volte è più conveniente riscrivere addirittura la riga per risparmiare tempo!!!

Ecco quindi una breve routine in ASSEMBLER che facilita notevolmente il lavoro di edit in un programma consentendo di risparmiare una notevole quantità di tempo. Se aggiungiamo che resta disponibile il vecchio sistema e che la routine "ruba" solo 512 byte di memoria RAM allora non si capisce veramente come mai non ci

abbia pensato la Apple!!! Misteri dell'informatica ...

Istruzioni per il caricamento

Per utilizzare la routine è necessario entrare in monitor con l'istruzione CALL - 151 e caricarla byte per byte ad iniziare dalla locazione \$9400 come indicato in figura 1. Consiglio di caricare i byte a gruppi di 8 o 16 e controllare, mano a mano, che i dati inseriti siano esatti. Naturalmente questo lavoro si deve fare una sola volta. Terminata questa operazione è necessario salvare la routine, in linguaggio macchina con l'istruzione:

BSAVE EDIT. OBJ0, A\$9400, L1FF

Quindi bisogna caricare il programma di inizializzazione di figura 2 (pag. 102) con lo stesso sistema, ad iniziare dalla locazione \$341, e salvarlo con l'istruzione:

BSAVE EDINIT. OBJ0, A\$340, L\$80

Infine è necessario caricare il programma BASIC di figura 3 (pag. 102) che esegue un controllo sull'esattezza dei dati inseriti, e provvede all'inizializzazione della routi-

I minifloppy con i programmi per Apple II

Presso la redazione sono disponibili i minifloppy con alcuni dei programmi pubblicati nella rubrica di software per Apple II. Il prezzo è di 15.000 lire IVA compresa per ciascun minifloppy. Per l'ordinazione inviare l'importo (a mezzo assegno, c/c o vaglia postale) alla Technimedia srl, Via Valsolda 135, 00141 Roma

Minifloppy disponibili

codice	programma	MC n.
DA2/01	Motomuro	24
DA2/02	&DEBUG	28
DA3/03	EDIT Applesoft + INPUT senza INPUT	29

```

0000: 7 *
0001: 8 ROUTINE DELL'INTERPRETE USATE NEL PROGRAMMA
0002: 9 *
0003: 10 RDK EDU #FDDC :ROUTINE STANDARD DI INPUT
FC18: 11 OS EDU #FC18 :ARRIETRA IL CURSORE DI 1 POSIZIONE
FBFD: 12 VIDOUT EDU #FBFD :FA L'OUTPUT SU VIDEO DEL CARATTERE
0001: 13 ICONTENUTO NELL'ACCUMULATORE
FDE0: 14 COUT EDU #FDED :ROUTINE STANDARD DI OUTPUT
FFB4: 15 ADV EDU #FFB4 :FA AVANZARE IL CURSORE DI 1 POSIZ.
D067: 16 FRNUM EDU #D067 :VALUTA UNA FORMULA NUMERICA
E752: 17 GETADR EDU #E752
0000: 18 *
0001: 19 *LOCAZIONI RAM USATE DAL PROGRAMMA E DAL BASIC*
0002: 20 *
0003: 21 LINUM EDU #50 :CONTIENE IL NUMERO DI RIGA TROVATO
0004: 22 LDA GETADR
0005: 23 BUFF EDU #66 :USATO COME MEMORIA DI LAVORO
0006: 24 :INSIEME A #07 E #08
0240: 25 BUFFER EDU #240 :INIZIO DEL BUFFER DELLA RIGA
0B19: 26 NCHAR EDU #19 :NUMERO CARATTERI DELLA RIGA
0003: 27 #SM EDU #30 :POINTER DELLA ROUTINE DI INPUT
0000: 28 :STANDARD DEL BASIC
00F9: 29 YSAV1 EDU #F9 :MEMORIA PROVU. PER IL REGISTRO Y
0024: 30 CH EDU #24 :INDIRIZZO ORIZZONTALE DEL CURSORE
0025: 31 CV EDU #25 :INDIRIZZO VERTICALE DEL CURSORE
0341: 32 MESSAGE EDU #341 :INIZIO DEL MESSAGGIO DI LINE NOT FOUND
-----
NEXT OBJECT FILE NAME IS EDIT00C.OBJ0
9400: 33 ORG #9400
*****
9400: 34 *INIZIO ROUTINE*
9401: 35 #1 RICERCA PARAMETRI*
9402: 36 #2 RICERCA RIGA*
9403: 37 #3 COPIATURA RIGA IN BUFFER ASCII*
9404: 38 #4 NEMLINE RIGA IN BUFFER ASCII*
9405: 39 #5 RIGAINES LDY #800 :EMETTE IL MESSAGGIO
9406: 40 #6 NEMLINE LDA MESSAGE.Y :DI LINE NOT FOUND SE IL PROGRAMMA
9407: 41 #7 JSR COUT :E' FINITO SENZA
9408: 42 #8 INY :TROVARE LA RIGA RICHIESTA
9409: 43 #9 CPY #17
9410: 44 #10 BNE NEMLINE
9411: 45 #11 RTS
9412: 46 #12 #3 COPIATURA RIGA IN BUFFER ASCII*
9413: 47 TROVATO LDA #CHINIF :MODIFICA IL PUNTATORE DELLA
9414: 48 STA #36 :ROUTINE DI OUTPUT DEL
9415: 49 LDA #CHINIF :MONITOR PER USARE LA ROUTINE
9416: 50 STA #37
9417: 51 JSR #30EA
9418: 52 LDA #30EA
9419: 53 LDA #30EA
9420: 54 LDA #30EA
9421: 55 LDA #30EA
9422: 56 LDA #30EA
9423: 57 LDA #30EA
9424: 58 LDA #30EA
9425: 59 LDA #30EA
9426: 60 LDA #30EA
9427: 61 LDA #30EA
9428: 62 LDA #30EA
9429: 63 LDA #30EA
9430: 64 LDA #30EA
9431: 65 LDA #30EA
9432: 66 LDA #30EA
9433: 67 LDA #30EA
9434: 68 LDA #30EA
9435: 69 LDA #30EA
9436: 70 LDA #30EA
9437: 71 LDA #30EA
9438: 72 LDA #30EA
9439: 73 LDA #30EA
9440: 74 LDA #30EA
9441: 75 LDA #30EA
9442: 76 LDA #30EA
9443: 77 LDA #30EA
9444: 78 LDA #30EA
9445: 79 LDA #30EA
9446: 80 LDA #30EA
9447: 81 LDA #30EA
9448: 82 LDA #30EA
9449: 83 LDA #30EA
9450: 84 LDA #30EA
9451: 85 LDA #30EA
9452: 86 LDA #30EA
9453: 87 LDA #30EA
9454: 88 LDA #30EA
9455: 89 LDA #30EA
9456: 90 LDA #30EA
9457: 91 LDA #30EA
9458: 92 LDA #30EA
9459: 93 LDA #30EA
9460: 94 LDA #30EA
9461: 95 LDA #30EA
9462: 96 LDA #30EA
9463: 97 LDA #30EA
9464: 98 LDA #30EA
9465: 99 LDA #30EA
9466: 100 LDA #30EA
9467: 101 LDA #30EA
9468: 102 LDA #30EA
9469: 103 LDA #30EA
9470: 104 LDA #30EA
9471: 105 LDA #30EA
9472: 106 LDA #30EA
9473: 107 LDA #30EA
9474: 108 LDA #30EA
9475: 109 LDA #30EA
9476: 110 LDA #30EA
9477: 111 LDA #30EA
9478: 112 LDA #30EA
9479: 113 LDA #30EA
9480: 114 LDA #30EA
9481: 115 LDA #30EA
9482: 116 LDA #30EA
9483: 117 LDA #30EA
9484: 118 LDA #30EA
9485: 119 LDA #30EA
9486: 120 LDA #30EA
9487: 121 LDA #30EA
9488: 122 LDA #30EA
9489: 123 LDA #30EA
9490: 124 LDA #30EA
9491: 125 LDA #30EA
9492: 126 LDA #30EA
9493: 127 LDA #30EA
9494: 128 LDA #30EA
9495: 129 LDA #30EA
9496: 130 LDA #30EA
9497: 131 LDA #30EA
9498: 132 LDA #30EA
9499: 133 LDA #30EA
9500: 134 LDA #30EA
9501: 135 LDA #30EA
9502: 136 LDA #30EA
9503: 137 LDA #30EA
9504: 138 LDA #30EA
9505: 139 LDA #30EA
9506: 140 LDA #30EA
9507: 141 LDA #30EA
9508: 142 LDA #30EA
9509: 143 LDA #30EA
9510: 144 LDA #30EA
9511: 145 LDA #30EA
9512: 146 LDA #30EA
9513: 147 LDA #30EA
9514: 148 LDA #30EA
9515: 149 LDA #30EA
9516: 150 LDA #30EA
9517: 151 LDA #30EA
9518: 152 LDA #30EA
9519: 153 LDA #30EA
9520: 154 LDA #30EA
9521: 155 LDA #30EA
9522: 156 LDA #30EA
9523: 157 LDA #30EA
9524: 158 LDA #30EA
9525: 159 LDA #30EA
9526: 160 LDA #30EA
9527: 161 LDA #30EA
9528: 162 LDA #30EA
9529: 163 LDA #30EA
9530: 164 LDA #30EA
9531: 165 LDA #30EA
9532: 166 LDA #30EA
9533: 167 LDA #30EA
9534: 168 LDA #30EA
9535: 169 LDA #30EA
9536: 170 LDA #30EA
9537: 171 LDA #30EA
9538: 172 LDA #30EA
9539: 173 LDA #30EA
9540: 174 LDA #30EA
9541: 175 LDA #30EA
9542: 176 LDA #30EA
9543: 177 LDA #30EA
9544: 178 LDA #30EA
9545: 179 LDA #30EA
9546: 180 LDA #30EA
9547: 181 LDA #30EA
9548: 182 LDA #30EA
9549: 183 LDA #30EA
9550: 184 LDA #30EA
9551: 185 LDA #30EA
9552: 186 LDA #30EA
9553: 187 LDA #30EA
9554: 188 LDA #30EA
9555: 189 LDA #30EA
9556: 190 LDA #30EA
9557: 191 LDA #30EA
9558: 192 LDA #30EA
9559: 193 LDA #30EA
9560: 194 LDA #30EA
9561: 195 LDA #30EA
9562: 196 LDA #30EA
9563: 197 LDA #30EA
9564: 198 LDA #30EA
9565: 199 LDA #30EA
9566: 200 LDA #30EA
9567: 201 LDA #30EA
9568: 202 LDA #30EA
9569: 203 LDA #30EA
9570: 204 LDA #30EA
9571: 205 LDA #30EA
9572: 206 LDA #30EA
9573: 207 LDA #30EA
9574: 208 LDA #30EA
9575: 209 LDA #30EA
9576: 210 LDA #30EA
9577: 211 LDA #30EA
9578: 212 LDA #30EA
9579: 213 LDA #30EA
9580: 214 LDA #30EA
9581: 215 LDA #30EA
9582: 216 LDA #30EA
9583: 217 LDA #30EA
9584: 218 LDA #30EA
9585: 219 LDA #30EA
9586: 220 LDA #30EA
9587: 221 LDA #30EA
9588: 222 LDA #30EA
9589: 223 LDA #30EA
9590: 224 LDA #30EA
9591: 225 LDA #30EA
9592: 226 LDA #30EA
9593: 227 LDA #30EA
9594: 228 LDA #30EA
9595: 229 LDA #30EA
9596: 230 LDA #30EA
9597: 231 LDA #30EA
9598: 232 LDA #30EA
9599: 233 LDA #30EA
9600: 234 LDA #30EA
9601: 235 LDA #30EA
9602: 236 LDA #30EA
9603: 237 LDA #30EA
9604: 238 LDA #30EA
9605: 239 LDA #30EA
9606: 240 LDA #30EA
9607: 241 LDA #30EA
9608: 242 LDA #30EA
9609: 243 LDA #30EA
9610: 244 LDA #30EA
9611: 245 LDA #30EA
9612: 246 LDA #30EA
9613: 247 LDA #30EA
9614: 248 LDA #30EA
9615: 249 LDA #30EA
9616: 250 LDA #30EA
9617: 251 LDA #30EA
9618: 252 LDA #30EA
9619: 253 LDA #30EA
9620: 254 LDA #30EA
9621: 255 LDA #30EA
9622: 256 LDA #30EA
9623: 257 LDA #30EA
9624: 258 LDA #30EA
9625: 259 LDA #30EA
9626: 260 LDA #30EA
9627: 261 LDA #30EA
9628: 262 LDA #30EA
9629: 263 LDA #30EA
9630: 264 LDA #30EA
9631: 265 LDA #30EA
9632: 266 LDA #30EA
9633: 267 LDA #30EA
9634: 268 LDA #30EA
9635: 269 LDA #30EA
9636: 270 LDA #30EA
9637: 271 LDA #30EA
9638: 272 LDA #30EA
9639: 273 LDA #30EA
9640: 274 LDA #30EA
9641: 275 LDA #30EA
9642: 276 LDA #30EA
9643: 277 LDA #30EA
9644: 278 LDA #30EA
9645: 279 LDA #30EA
9646: 280 LDA #30EA
9647: 281 LDA #30EA
9648: 282 LDA #30EA
9649: 283 LDA #30EA
9650: 284 LDA #30EA
9651: 285 LDA #30EA
9652: 286 LDA #30EA
9653: 287 LDA #30EA
9654: 288 LDA #30EA
9655: 289 LDA #30EA
9656: 290 LDA #30EA
9657: 291 LDA #30EA
9658: 292 LDA #30EA
9659: 293 LDA #30EA
9660: 294 LDA #30EA
9661: 295 LDA #30EA
9662: 296 LDA #30EA
9663: 297 LDA #30EA
9664: 298 LDA #30EA
9665: 299 LDA #30EA
9666: 300 LDA #30EA
9667: 301 LDA #30EA
9668: 302 LDA #30EA
9669: 303 LDA #30EA
9670: 304 LDA #30EA
9671: 305 LDA #30EA
9672: 306 LDA #30EA
9673: 307 LDA #30EA
9674: 308 LDA #30EA
9675: 309 LDA #30EA
9676: 310 LDA #30EA
9677: 311 LDA #30EA
9678: 312 LDA #30EA
9679: 313 LDA #30EA
9680: 314 LDA #30EA
9681: 315 LDA #30EA
9682: 316 LDA #30EA
9683: 317 LDA #30EA
9684: 318 LDA #30EA
9685: 319 LDA #30EA
9686: 320 LDA #30EA
9687: 321 LDA #30EA
9688: 322 LDA #30EA
9689: 323 LDA #30EA
9690: 324 LDA #30EA
9691: 325 LDA #30EA
9692: 326 LDA #30EA
9693: 327 LDA #30EA
9694: 328 LDA #30EA
9695: 329 LDA #30EA
9696: 330 LDA #30EA
9697: 331 LDA #30EA
9698: 332 LDA #30EA
9699: 333 LDA #30EA
9700: 334 LDA #30EA
9701: 335 LDA #30EA
9702: 336 LDA #30EA
9703: 337 LDA #30EA
9704: 338 LDA #30EA
9705: 339 LDA #30EA
9706: 340 LDA #30EA
9707: 341 LDA #30EA
9708: 342 LDA #30EA
9709: 343 LDA #30EA
9710: 344 LDA #30EA
9711: 345 LDA #30EA
9712: 346 LDA #30EA
9713: 347 LDA #30EA
9714: 348 LDA #30EA
9715: 349 LDA #30EA
9716: 350 LDA #30EA
9717: 351 LDA #30EA
9718: 352 LDA #30EA
9719: 353 LDA #30EA
9720: 354 LDA #30EA
9721: 355 LDA #30EA
9722: 356 LDA #30EA
9723: 357 LDA #30EA
9724: 358 LDA #30EA
9725: 359 LDA #30EA
9726: 360 LDA #30EA
9727: 361 LDA #30EA
9728: 362 LDA #30EA
9729: 363 LDA #30EA
9730: 364 LDA #30EA
9731: 365 LDA #30EA
9732: 366 LDA #30EA
9733: 367 LDA #30EA
9734: 368 LDA #30EA
9735: 369 LDA #30EA
9736: 370 LDA #30EA
9737: 371 LDA #30EA
9738: 372 LDA #30EA
9739: 373 LDA #30EA
9740: 374 LDA #30EA
9741: 375 LDA #30EA
9742: 376 LDA #30EA
9743: 377 LDA #30EA
9744: 378 LDA #30EA
9745: 379 LDA #30EA
9746: 380 LDA #30EA
9747: 381 LDA #30EA
9748: 382 LDA #30EA
9749: 383 LDA #30EA
9750: 384 LDA #30EA
9751: 385 LDA #30EA
9752: 386 LDA #30EA
9753: 387 LDA #30EA
9754: 388 LDA #30EA
9755: 389 LDA #30EA
9756: 390 LDA #30EA
9757: 391 LDA #30EA
9758: 392 LDA #30EA
9759: 393 LDA #30EA
9760: 394 LDA #30EA
9761: 395 LDA #30EA
9762: 396 LDA #30EA
9763: 397 LDA #30EA
9764: 398 LDA #30EA
9765: 399 LDA #30EA
9766: 400 LDA #30EA
9767: 401 LDA #30EA
9768: 402 LDA #30EA
9769: 403 LDA #30EA
9770: 404 LDA #30EA
9771: 405 LDA #30EA
9772: 406 LDA #30EA
9773: 407 LDA #30EA
9774: 408 LDA #30EA
9775: 409 LDA #30EA
9776: 410 LDA #30EA
9777: 411 LDA #30EA
9778: 412 LDA #30EA
9779: 413 LDA #30EA
9780: 414 LDA #30EA
9781: 415 LDA #30EA
9782: 416 LDA #30EA
9783: 417 LDA #30EA
9784: 418 LDA #30EA
9785: 419 LDA #30EA
9786: 420 LDA #30EA
9787: 421 LDA #30EA
9788: 422 LDA #30EA
9789: 423 LDA #30EA
9790: 424 LDA #30EA
9791: 425 LDA #30EA
9792: 426 LDA #30EA
9793: 427 LDA #30EA
9794: 428 LDA #30EA
9795: 429 LDA #30EA
9796: 430 LDA #30EA
9797: 431 LDA #30EA
9798: 432 LDA #30EA
9799: 433 LDA #30EA
9800: 434 LDA #30EA
9801: 435 LDA #30EA
9802: 436 LDA #30EA
9803: 437 LDA #30EA
9804: 438 LDA #30EA
9805: 439 LDA #30EA
9806: 440 LDA #30EA
9807: 441 LDA #30EA
9808: 442 LDA #30EA
9809: 443 LDA #30EA
9810: 444 LDA #30EA
9811: 445 LDA #30EA
9812: 446 LDA #30EA
9813: 447 LDA #30EA
9814: 448 LDA #30EA
9815: 449 LDA #30EA
9816: 450 LDA #30EA
9817: 451 LDA #30EA
9818: 452 LDA #30EA
9819: 453 LDA #30EA
9820: 454 LDA #30EA
9821: 455 LDA #30EA
9822: 456 LDA #30EA
9823: 457 LDA #30EA
9824: 458 LDA #30EA
9825: 459 LDA #30EA
9826: 460 LDA #30EA
9827: 461 LDA #30EA
9828: 462 LDA #30EA
9829: 463 LDA #30EA
9830: 464 LDA #30EA
9831: 465 LDA #30EA
9832: 466 LDA #30EA
9833: 467 LDA #30EA
9834: 468 LDA #30EA
9835: 469 LDA #30EA
9836: 470 LDA #30EA
9837: 471 LDA #30EA
9838: 472 LDA #30EA
9839: 473 LDA #30EA
9840: 474 LDA #30EA
9841: 475 LDA #30EA
9842: 476 LDA #30EA
9843: 477 LDA #30EA
9844: 478 LDA #30EA
9845: 479 LDA #30EA
9846: 480 LDA #30EA
9847: 481 LDA #30EA
9848: 482 LDA #30EA
9849: 483 LDA #30EA
9850: 484 LDA #30EA
9851: 485 LDA #30EA
9852: 486 LDA #30EA
9853: 487 LDA #30EA
9854: 488 LDA #30EA
9855: 489 LDA #30EA
9856: 490 LDA #30EA
9857: 491 LDA #30EA
9858: 492 LDA #30EA
9859: 493 LDA #30EA
9860: 494 LDA #30EA
9861: 495 LDA #30EA
9862: 496 LDA #30EA
9863: 497 LDA #30EA
9864: 498 LDA #30EA
9865: 499 LDA #30EA
9866: 500 LDA #30EA
9867: 501 LDA #30EA
9868: 502 LDA #30EA
9869: 503 LDA #30EA
9870: 504 LDA #30EA
9871: 505 LDA #30EA
9872: 506 LDA #30EA
9873: 507 LDA #30EA
9874: 508 LDA #30EA
9875: 509 LDA #30EA
9876: 510 LDA #30EA
9877: 511 LDA #30EA
9878: 512 LDA #30EA
9879: 513 LDA #30EA
9880: 514 LDA #30EA
9881: 515 LDA #30EA
9882: 516 LDA #30EA
9883: 517 LDA #30EA
9884: 518 LDA #30EA
9885: 519 LDA #30EA
9886: 520 LDA #30EA
9887: 521 LDA #30EA
9888: 522 LDA #30EA
9889: 523 LDA #30EA
9890: 524 LDA #30EA
9891: 525 LDA #30EA
9892: 526 LDA #30EA
9893: 527 LDA #30EA
9894: 528 LDA #30EA
9895: 529 LDA #30EA
9896: 530 LDA #30EA
9897: 531 LDA #30EA
9898: 532 LDA #30EA
9899: 533 LDA #30EA
9900: 534 LDA #30EA
9901: 535 LDA #30EA
9902: 536 LDA #30EA
9903: 537 LDA #30EA
9904: 538 LDA #30EA
9905: 539 LDA #30EA
9906: 540 LDA #30EA
9907: 541 LDA #30EA
9908: 542 LDA #30EA
9909: 543 LDA #30EA
9910: 544 LDA #30EA
9911: 545 LDA #30EA
9912: 546 LDA #30EA
9913: 547 LDA #30EA
9914: 548 LDA #30EA
9915: 549 LDA #30EA
9916: 550 LDA #30EA
9917: 551 LDA #30EA
9918: 552 LDA #30EA
9919: 553 LDA #30EA
9920: 554 LDA #30EA
9921: 555 LDA #30EA
9922: 556 LDA #30EA
9923: 557 LDA #30EA
9924: 558 LDA #30EA
9925: 559 LDA #30EA
9926: 560 LDA #30EA
9927: 561 LDA #30EA
9928: 562 LDA #30EA
9929: 563 LDA #30EA
9930: 564 LDA #30EA
9931: 565 LDA #30EA
9932: 566 LDA #30EA
9933: 567 LDA #30EA
9934: 568 LDA #30EA
9935: 569 LDA #30EA
9936: 570 LDA #30EA
9937: 571 LDA #30EA
9938: 572 LDA #30EA
9939: 573 LDA #30EA
9940: 574 LDA #30EA
9941: 575 LDA #30EA
9942: 576 LDA #30EA
9943: 577 LDA #30EA
9944: 578 LDA #30EA
9945: 579 LDA #30EA
9946: 580 LDA #30EA
9947: 581 LDA #30EA
9948: 582 LDA #30EA
9949: 583 LDA #30EA
9950: 584 LDA #30EA
9951: 585 LDA #30EA
9952: 586 LDA #30EA
9953: 587 LDA #30EA
9954: 588 LDA #30EA
9955: 589 LDA #30EA
9956: 590 LDA #30EA
9957: 591 LDA #30EA
9958: 592 LDA #30EA
9959: 593 LDA #30EA
9960: 594 LDA #30EA
9961: 595 LDA #30EA
9962: 596 LDA #30EA
9963: 597 LDA #30EA
9964: 598 LDA #30EA
9965: 599 LDA #30EA
9966: 600 LDA #30EA
9967: 601 LDA #30EA
9968: 602 LDA #30EA
9969: 603 LDA #30EA
9970: 604 LDA #30EA
9971: 605 LDA #30EA
9972: 606 LDA #30EA
9973: 607 LDA #30EA
9974: 608 LDA #30EA
9975: 609 LDA #30EA
9976: 610 LDA #30EA
9977: 611 LDA #30EA
9978: 612 LDA #30EA
9979: 613 LDA #30EA
9980: 614 LDA #30EA
9981: 615 LDA #30EA
9982: 616 LDA #30EA
9983: 617 LDA #30EA
9984: 618 LDA #30EA
9985: 619 LDA #30EA
9986: 620 LDA #30EA
9987: 621 LDA #30EA
9988: 622 LDA #30EA
9989: 623 LDA #30EA
9990: 624 LDA #30EA
9991: 625 LDA #30EA
9992: 626 LDA #30EA
9993: 627 LDA #30EA
9994: 628 LDA #30EA
9995: 629 LDA #30EA
9996: 630 LDA #30EA
9997: 631 LDA #30EA
9998: 632 LDA #30EA
9999: 633 LDA #30EA
10000: 634 LDA #30EA
10001: 635 LDA #30EA
10002: 636 LDA #30EA
10003: 637 LDA #30EA
10004: 638 LDA #30EA
10005: 639 LDA #30EA
10006: 640 LDA #30EA
10007: 641 LDA #30EA
10008: 642 LDA #30EA
10009: 643 LDA #30EA
10010: 644 LDA #30EA
10011: 645 LDA #30EA
10012: 646 LDA #30EA
10013: 647 LDA #30EA
10014: 648 LDA #30EA
10015: 649 LDA #30EA
10016: 650 LDA #30EA
10017: 651 LDA #30EA
10018: 652 LDA #30EA
10019: 653 LDA #30EA
10020: 654 LDA #30EA
10021: 655 LDA #30EA
10022: 656 LDA #30EA
10023: 657 LDA #30EA
10024: 658 LDA #30EA
10025: 659 LDA #30EA
10026: 660 LDA #30EA
10027: 661 LDA #30EA
10028: 662 LDA #30EA
10029: 663 LDA #30EA
10030: 664 LDA #30EA
10031: 665 LDA #30EA
10032: 666 LDA #30EA
10033: 667 LDA #30EA
10034: 668 LDA #30EA
10035: 669 LDA #30EA
10036: 670 LDA #30EA
10037: 671 LDA #30EA
10038: 672 LDA #30EA
10039: 673 LDA #30EA
10040: 674 LDA #30EA
10041: 675 LDA #30EA
10042: 676 LDA #30EA
10043: 677 LDA #30EA
10044: 678 LDA #30EA
10045: 679 LDA #30EA
10046: 680 LDA #30EA
10047: 681 LDA #30EA
10048: 682 LDA #30EA
10049: 683 LDA #30EA
10050: 684 LDA #30EA
10051: 685 LDA #30EA
10052: 686 LDA #30EA
10053: 687 LDA #30EA
10054: 688 LDA #30EA
10055: 689 LDA #30EA
10056: 690 LDA #30EA
10057: 691 LDA #30EA
10058: 692 LDA #30EA
10059: 693 LDA #30EA
10060: 694 LDA #30EA
10061: 695 LDA #30EA
10062: 696 LDA #30EA
10063: 697 LDA #30EA
10064: 698 LDA #30EA
10065: 699 LDA #30EA
10066: 700 LDA #30EA
10067: 701 LDA #30EA
10068: 702 LDA #30EA
10069: 703 LDA #30EA
10070: 704 LDA #30EA
10071: 705 LDA #30EA
10072: 706 LDA #30EA
10073: 707 LDA #30EA
10074: 708 LDA #30EA
10075: 709 LDA #30EA
10076: 710 LDA #30EA
10077: 711 LDA #30EA
10078: 712 LDA #30EA
10079: 713 LDA #30EA
10080: 714 LDA #30EA
10081: 715 LDA #30EA
10082: 716 LDA #30EA
10083
```


ne. Attenzione: il controllo avviene sommando le varie locazioni di memoria della routine. Può perciò succedere, come per la "famosa" prova del 9, che due errori si annullino a vicenda.

A questo punto si dà il run e, se non succede niente, la routine è già pronta per l'uso, altrimenti il programma emette un messaggio d'errore, e la routine deve essere corretta o ricaricata. A questo punto si possono cancellare le righe dal numero 20 al numero 40 e salvare il programma Basic con il nome p.e. di EDIT.

SAVE EDIT

Se tutto è andato bene, per attivare la routine è adesso sufficiente battere:

RUN EDIT

È possibile naturalmente inserire le istruzioni Basic sopracitate nel programma che il DOS esegue al bootstrap in modo che l'Apple II le esegua automaticamente all'accensione.

Istruzioni per l'uso della routine

Per editare una riga è sufficiente battere l'istruzione:

& NN

dove NN è il numero di riga da correggere. A questo punto la riga appare sul video, come per un normale LIST, tranne per i caratteri di controllo che sono scritti in negativo, e con il cursore che si posiziona

sul primo carattere. Per muovere il cursore e posizionarlo nel punto voluto, si usano le solite frecce e i caratteri <CTRL A> e <CTRL Z> rispettivamente per salire e per scendere. Dopo essersi posizionati sul punto desiderato, è possibile manipolare la riga con i seguenti comandi:

CHANGE: per sostituire uno o più caratteri all'interno della riga è sufficiente semplicemente posizionarsi sul carattere errato e ribattere quello corretto. Attenzione: se una volta giunti a fine riga si continua ad inserire dei caratteri, questi continuano a sostituirsi all'ultimo premuto.

INSERT: per inserire dei caratteri bisogna posizionarsi sul punto dove si desidera fare l'inserimento e premere <ESC> per entrare nel modo Insert.

Attenzione: nel modo Insert, l'unico carattere di controllo cursore disponibile è la freccia indietro che cancella l'ultimo carattere inserito.

Per uscire dal modo Insert e tornare nel modo Edit normale premere nuovamente <ESC>.

DELETE: per cancellare un carattere è sufficiente posizionarsi sul carattere da cancellare e premere <CTRL D>

RESTORE: premendo <CTRL R> si cancellano tutte le modifiche effettuate, e la riga viene riportata nel buffer di edit

nella sua forma originaria pronta per essere corretta. Premere questo tasto quindi, quando si sono fatte delle modifiche non volute.

END: per terminare l'edit e riportare la riga corretta nel programma Basic premere <RETURN>.

Come funziona

Come non è necessario sapere come funziona al suo interno un computer per usarlo, così alla maggior parte degli utilizzatori di questa routine non interesserà minimamente sapere come funziona.

Tuttavia per chi avesse intenzione di trascorrere alcune notti in bianco a studiare il listato sorgente pubblicato in queste pagine (poveretti!!) per scoprire come funziona la routine, e magari modificarla, dirò che essa è composta essenzialmente di 3 parti:

1) preleva la riga da correggere dal programma Basic in memoria, e la porta nel "buffer" di correzione situato a partire dalla locazione \$240, trasformandola opportunamente in una stringa di caratteri ASCII. Ricordo infatti che in memoria le istruzioni Basic sono "tokenizzate", cioè sostituite da un numero maggiore di 127, come riportato dall'appendice F del manuale di riferimento Applesoft, ed è perciò necessario provvedere alla loro decodificazione (\$946A - \$9496).

```

94E1:20 BC 95 152 JSR DELETE ;PRECEDENTEMENTE INSERITO
94E1:A9 80 153 LDA #80
94F0:C9 A0 154 MZ? BMT #80
94F2:38 E9 155 BPI INSERT
94F4:B5 08 156 STA BUFF+2
94F6:E6 19 157 INC NCHAR ;SPOSTA IN AVANTI TUTTI I CARATTERI
94F8:E6 19 158 INC NCHAR
94FA:B6 09 159 STX BUFF+3 ;FINO AL FINE RIGA PER FAR POSTO
94FC:A4 09 160 LDY BUFF+3 ;AL CARATTERE DA INSERIRE
94FE:B9 40 02 161 OTHER LDA BUFFER,Y
9501:A8 162 PHA
9502:A5 08 163 LDA BUFF+2
9504:99 40 02 164 STA BUFFER,Y
9507:68 165 PLA
9508:B5 08 166 STA BUFF+2
950A:C8 167 INY
950B:C4 19 168 CPY NCHAR
950D:F8 EF 169 BEQ OTHER
950F:98 ED 170 ECC OTHER
9511:E8 171 INX
9512:20 67 95 172 JSR LIST ;SPOSTA IN AVANTI IL CURSORE
9515:C6 19 173 DEC NCHAR ;STAMPA SU VIDEO LA RIGA COSI'
9517:D8 C4 174 BNE INSERT ;MODIFICATA
9519: 175 *ROUTINE METTE CHAR IN BUFF*
951B:86 09 176 CHINEF STX BUFF+3 ;SUBROUTINE CHE METTE NEL BUFFER IL
951D:A6 19 177 LDX NCHAR ;CARATTERE PRESENTE NELL'ACCUMULATORE

951D:20 25 95 178 JSR CHINEF1 ;AGGIORNANDO IL CONTATORE DI CARATTERI

9520:B6 19 179 STX NCHAR ;E LASCIANDO INALTERATO IL REGISTRO X

9522:A6 09 180 LDX BUFF+3
9524:68 181 RTG
9525:F8 80 182 CHINEF1 ORA #80
9527:9D 40 02 183 STA BUFFER,X ;IDEM COME SOPRA TRANNE CHE MODIFICA
952A:E8 184 INX ;IL REGISTRO X
952B:68 185 RTS
952C:20 50 95 186 END JSR RESTORE ;ROUTINE CHE PROVVUDE ALL'INSERIMENTO
DELLA
952F:A9 3F 187 LDA #>INPUT ;DELLA RIGA CORRETTA NEL PROGRAMMA
9531:B5 38 188 STA KSWL ;BASIC MODIFICANDO IL PUNTIATORE DELLA

9533:A9 95 189 LDA #<INPUT ;ROUTINE DI INPUT (#38-#39) PERCHE'
9535:B5 39 190 STA KSWL+1 ;ACCETTI L'INPUT DALLA SEGUENTE
9537:20 EA 03 191 JSR #3EA
953A:A9 80 192 LDA #80
953C:B5 06 193 STA BUFF
953E:68 194 RTS
953F:86 07 195 INPUT STX BUFF+1 ;ROUTINE CHE INVIA ALL'INTERPRETE
9541:A6 06 196 LDX BUFF ;BASIC I CARATTERI COMPONENTI LA RIGA

9543:80 40 02 197 LDA BUFFER,X ;QUANDO LA RIGA E' FINITA
9546:C9 80 198 CMP #80 ;SPEDISCE UN CR E RIPRISTINA I
9548:D8 00 199 BNE NONCR ;PUNTIATORI DELLA ROUTINE DI INPUT
954A:A9 18 200 LDA #18 ;ALLA ROUTINE CHE LEGGE LA TASTIERA
954C:B5 38 201 STA KSWL
954E:A9 FD 202 LDA #FD
9550:B5 39 203 STA KSWL+1
9552:20 EA 03 204 JSR #3EA
9555:A9 80 205 LDA #80
9557:E8 206 NONCR INX
9558:B6 06 207 STX BUFF
955A:A6 07 208 LDX BUFF+1
955C:68 209 RTS
955D:A9 11 210 RESTORE LDA #17 ;PORTA IL CURSORE A CAPO DELLA 17 RIGA

955F:B5 25 211 STA CV
9561:A9 80 212 LDA #80
9563:B5 24 213 STA CH

9565:AA 214 TAX
9566:68 215 RTS
9567: 216 *LIST* ;STAMPA SU VIDEO IL CONTENUTO DEL BUFFER
9567:86 88 217 LIST STX BUFF+2 ;LASCIANDO INALTERATO IL REGISTRO X
    
```

```

9569:20 50 95 218 JSR RESTORE ;E LA POSIZIONE DEL CURSORE
956C:20 62 FC 219 JSR #FC2
956F:80 40 02 220 LOOP1 LDA BUFFER,X
9572:F8 0C 221 BEQ ENDL ;CONTROLLA LA FINE DELLA RIGA
9574:C9 A0 222 CMP #80
9576:18 02 223 BPL VIDEO
9578:29 7F 224 AND #7F
957A:20 FD FB 225 VIDEO JSR VIDEOUT ;CONTROLLO E IN POSITIVO GLI ALTRI
957D:E8 226 INX
957E:D8 EF 227 BNE LOOP1
9580:20 9C FC 228 ENDL JSR #FC9C
9583:20 18 FC 229 BX JSR BS ;RIPORTA IL CURSORE AL SUO POSTO
9586:1C A 230 DEX
9587:1E 08 231 CPX BUFF+2
9589:D8 FB 232 BNE BX
958B:A8 233 RTS
958C:86 09 234 DELETE STX BUFF+3 ;GESTISCE LA CANCELLAZIONE DI UN
958E:1E 19 235 CPX NCHAR ;CARATTERE DAL BUFFER SPOSTANDO
9591:F0 1F 236 BEQ LAST ;ALL'INSIETRO TUTTI I CARATTERI RESTAN
TI
9592:A4 19 237 LDY NCHAR
9594:A9 80 238 LDA #80
9596:A5 08 239 STA BUFF+2
9598:B6 09 240 LOPS LDA BUFFER,Y
959B:A8 241 PHA
959C:A5 08 242 LDA BUFF+2
959E:99 40 02 243 STA BUFFER,Y
95A1:68 244 PLA
95A2:B5 08 245 STA BUFF+2
95A4:88 246 DEY
95A5:F0 84 247 BEQ DECN
95A7:C4 09 248 CPY BUFF+3
95A9:B8 ED 249 BCS LOOPS
95AB:C4 19 250 DECH NCHAR
95AD:20 67 95 251 EDIT3 JSR LIST ;STAMPA LA RIGA COSI' OTTENUTA
95B0:68 252 RTS
95B1:A9 A0 253 LAST LDA #A0
95B3:9D 40 02 254 STA BUFFER,X ;SE IL CARATTERE E' L'ULTIMO INVECE
95B6:D8 F5 255 BNE EDIT3 ;DI CANCELLARLO LO SOSTITUISCE
95B8:E4 19 256 ADVANCE CPX NCHAR ;CON UNO SPAZIO
95BA:F0 04 257 BEQ RTS3 ;GESTISCE L'AVANZAMENTO DEL CURSORE
95BC:20 FA FB 258 JSR ADV ;DI UNA POSIZIONE
95BF:E8 259 INX ;CONTROLLANDO LA FINE DELLA RIGA
95C1:68 260 RTG3 RTS #A'-64
95C3:18 08 261 BACK CPX #80
95C5:F8 FB 262 BEQ RTS3 ;GESTISCE L'ARRETRAMENTO DEL CURSORE
95C8:20 10 FC 263 JSR BS ;DI UNA POSIZIONE CONTROLLANDO
95CB:CA 264 DEX ;L'INIZIO RIGA
95CD:A8 265 RTS
95CA:A8 28 266 UP LDY #40 ;SPOSTA IL CURSORE VERSO L'ALTO DI UNA
RIGA
95CC:2C C1 95 267 MZ5 JSR BACK ;RIGA, SE NON E' POSSIBILE LO PORTA
95CF:B8 268 DEY ;A INIZIATORICA
95D0:D8 FA 269 BNE MZ5
95D2:68 270 RTS
95D3:18 28 271 DOWN LDY #40 ;SPOSTA IL CURSORE VERSO IL BASSO DI
95D5:20 B8 95 272 MZ6 JSR ADVANCE ;UNA RIGA SE NON E' POSSIBILE LO PORTA
A FINE RIGA
95D8:88 273 DEY
95D9:D8 FA 274 BNE MZ6
95DB:68 275 RTS
95DC:C9 95 276 MOVIMENTI CMP #21+128 ;CONTROLLA SE SONO STATI PREVENTI I
95DE:F8 08 277 BEQ ADVANCE ;TASTI RELATIVI AI VARI MOVIMENTI DEL
CURSORE E LI ESEGUE DI CONSEGUENZA
95E0:C9 88 278 CMP #88+128
95E2:F8 00 279 BEQ BACK
95E4:C9 81 280 CMP #A'-64
95E6:F0 E2 281 BEQ UP
95E8:C9 9A 282 CMP #Z'-64
95EA:F0 E7 283 BEQ DOWN
95EC:C9 84 284 CMP #D'-64
95EE:F8 9C 285 BEQ DELETE
95F0:68 286 RTS
    
```



```

9400- 20 67 DD 20 52 E7 A9 00
9408- 85 19 A5 67 85 06 A5 68
9410- 85 07 A0 02 B1 06 C5 50
9418- D0 07 C8 B1 06 C5 51 F0
9420- 22 A0 00 B1 06 85 08 C8
9428- B1 06 F0 09 85 07 A5 08
9430- 85 06 4C 12 94 A0 00 B9
9438- 41 03 20 ED FD C8 C0 17
9440- D0 F5 60 A9 19 85 36 A9
9448- 95 85 37 20 EA 03 A6 50
9450- A5 51 20 24 ED A9 F0 85
9458- 36 A9 FD 85 37 20 EA 03
9460- A0 04 84 F9 B1 06 F0 39
9468- 10 2E 29 7F AA A9 CE 85
9470- 81 A9 D0 85 B2 A0 00 B1
9478- 81 10 03 CA 30 09 E6 81
9480- D0 02 E6 82 4C 77 94 A9
9488- A0 20 19 95 C8 B1 B1 48
9490- 20 19 95 68 10 F6 A9 A0
9498- 20 19 95 A4 F9 C8 4C 62
94A0- 94 A6 19 9D 40 02 20 5D
94A8- 95 20 42 FC C6 19 20 67
94B0- 95 F0 06 20 25 95 20 67
94B8- 95 20 0C FD C9 9B F0 1D
94C0- C9 92 D0 03 4C 06 94 C9
94C8- 8D F0 61 48 20 DC 95 68
94D0- C9 A0 30 E5 E4 19 D0 DB
94D8- 9D 40 02 F0 D9 20 0C FD
94E0- C9 9B F0 D5 C9 8B D0 08
94E8- 20 C1 95 20 8C 95 A9 80
94F0- C9 A0 30 E9 85 08 E6 19
94F8- E6 19 86 09 A4 09 B9 40
9500- 02 48 A5 08 99 40 02 68
9508- 85 08 C8 C4 19 F0 EF 90
9510- ED E8 20 67 95 C6 19 D0
9518- C4 86 09 A6 19 20 25 95
9520- 86 19 A6 09 60 09 80 9D
9528- 40 02 E8 60 20 5D 95 A9
9530- 3F 85 38 A9 95 85 39 20
9538- EA 03 A9 00 85 06 60 B6
9540- 07 A6 06 BD 40 02 C9 00
9548- D0 0D A9 1B 85 38 A9 FD
9550- 85 39 20 EA 03 A9 8D E8
9558- 86 06 A6 07 60 A9 11 85
9560- 25 A9 00 85 24 AA 60 86
9568- 08 20 5D 95 20 62 FC BD
9570- 40 02 F0 0C C9 A0 10 02
9578- 29 7F 20 FD FB E8 D0 EF
9580- 20 9C FC 20 10 FC CA E4
9588- 08 D0 F8 60 86 09 E4 19
9590- F0 1F A4 19 A9 00 85 08
9598- B9 40 02 48 A5 08 99 40
95A0- 02 68 85 08 88 F0 04 C4
95A8- 09 B0 ED C6 19 20 67 95
95B0- 60 A9 A0 9D 40 02 D0 F5
95B8- E4 19 F0 04 20 F4 FB E8
95C0- 60 E0 00 F0 FB 20 10 FC
95C8- CA 60 A0 2B 20 C1 95 8B
95D0- D0 FA 60 A0 2B 20 B8 95
95D8- 8B D0 FA 60 C9 95 F0 DB
95E0- C9 8B F0 DD C9 81 F0 E2
95E8- C9 9A F0 E7 C9 84 F0 9C
95F0- 60

```

Figura 1 - Codice oggetto del programma EDIT.

```

10 HIMEM: 37887: PRINT CHR$(4)"BLOAD EDIT.OBJ0": PRINT CHR$(4)"BLOAD
  EDINIT.OBJ0": TT = 0
20 FOR X = 37888 TO 38384: TT = TT + PEEK(X): NEXT X: IF TT < > 60004 THEN
  PRINT "ERRORE IN EDIT.OBJ0": CHR$(7): END
25 TT = 0
30 FOR X = 833 TO 939: TT = TT + PEEK(X): NEXT X: IF TT < > 18724 THEN
  PRINT "ERRORE IN EDINIT.OBJ0": CHR$(7): END
40 PRINT "ROUTINE O.K."
50 CALL 912
60 NEW

```

Figura 3

```

0340- 60 8D CC C9 CE C5 A0 CE
0348- CF D4 A0 C6 CF D5 CE C4
0350- A0 C5 D2 D2 CF D2 87 8D
0358- 8D C1 D0 D0 CC C5 D3 CF
0360- C6 D4 A0 C5 C4 C9 D4 CF
0368- D2 A0 D2 CF D5 D4 C9 CE
0370- C5 A0 B1 AE B0 8D A8 C3
0378- A9 A0 B1 B9 B8 B3 A0 C2
0380- D9 A0 CD C5 D2 CC C5 D2
0388- A0 CD C1 D2 C3 CF 8D 00
0390- A9 4C 8D F5 03 A9 94 8D
0398- F7 03 A9 00 8D F6 03 A8
03A0- B9 58 03 F0 06 20 ED FD
03A8- C8 D0 F5 60

```

Figura 2

2) provvede alla correzione della riga nel buffer seguendo i comandi dell'operatore indicati sopra.

3) quando l'operatore preme <return> riporta la riga corretta nel programma Basic sostituendola alla precedente.

Per evitare una gestione del video complessa la routine modifica esclusivamente il buffer e provvede a ristamparlo, completamente, dopo ogni modifica. Viene inoltre mantenuto un cursore interno che indi-

AVVERTENZE

1) I caratteri di controllo vengono stampati sul video in negativo. Possono venire cancellati o sostituiti con caratteri normali, ma *né nel modo normale, né nel modo insert possono essere inseriti caratteri di controllo nella linea da editare*. La routine li rifiuterà automaticamente.

2) Per motivi che è difficile spiegare, è impossibile cancellare il primo carattere con il tasto <CTRL D>. Pertanto posizionandosi con il cursore sul primo carattere, e premendo tale tasto *verrà cancellato il secondo carattere*. È però possibile cancellare il primo carattere, sostituendo ad esso uno spazio.

3) Con questa versione non è possibile editare righe di più di 240 caratteri. Per superare tale limite è sufficiente dare il comando POKÉ 38238,16. In ogni modo però è necessario ricordarsi di non superare i 255 caratteri, perché la routine non esegue alcun controllo sulla lunghezza, e i caratteri in più andrebbero persi con risultati imprevedibili.

ca il carattere sul quale è posizionato il cursore video ed è rappresentato dal registro X. Tutte le modifiche (Change, Insert, Delete etc..) vengono eseguite sul carattere indicato da tale registro. La breve routine situata a partire dalla locazione \$340 serve a stampare i messaggi di errore, di copyright e a collegare la routine al simbolo &.

Detto questo penso che il listato dovrebbe essere comprensibile, per la maggior parte delle istruzioni, a chiunque abbia un minimo di conoscenza del linguaggio macchina del 6502.

L'input è meglio senza INPUT

Se avete provato ad usare la INPUT del Basic per creare un file EXEC o per inserire una lista di nomi e numeri in un programma di archivio, per esempio una rubrica, vi sarà certamente capitato di veder apparire la scritta EXTRA IGNORED alla pressione del tasto di Return. Questo succede quando la riga digitata contiene una virgola, che la funzione INPUT utilizza come separatore tra gli argomenti.

Per ovviare all'inconveniente si deve fare ricorso ad una complicata e pesante gestione dell'input da tastiera mediante l'uso della GET e il successivo concatenamento alla stringa in uso.

La routine che vi presentiamo consente, con una semplice CALL, di prelevare da tastiera una riga qualsiasi e di assegnarla ad una variabile stringa a piacere.

La routine è lunga più di 200 byte e questo ci impedisce di metterla nella solita pagina \$3 a partire dalla locazione 768; così si è deciso di metterla in una zona di memoria che nessun programma andrà mai a sporcare: tra il DOS e i suoi BUFFER.

In realtà tra il DOS e i Buffer ci sono solo sette byte liberi, ma basta spostare i puntatori al buffer di 255 indietro per creare lo spazio necessario alla nostra routine. Il DOS, per trovare l'inizio del buffer, legge il valore contenuto nelle locazioni \$9D00 e \$9D01, la prima locazione contiene la parte bassa, la seconda la parte alta. La parte bassa non ci interessa in quanto la nostra routine entra esattamente in una pagina, ci basta quindi decrementare di uno la parte alta (locazione \$9D01) per far indietreggiare di 255 byte i BUFFER del DOS. Una volta cambiato il valore si farà un JMP \$3D3 per eseguire un COLD-START del DOS e fargli accettare la modifica.

Non è purtroppo possibile lanciare in esecuzione una routine che si sovrappone ai buffer del DOS prima che si sia riservato lo spazio opportuno. Perciò la routine viene caricata a partire dalla locazione \$3FED e una piccola routine iniziale provvede a spostare i puntatori dei buffer e trasferire il resto della routine nella zona da \$9C00 a \$9CFF. Attenzione al fatto che il JMP \$3D3, che effettua il Coldstart del DOS, effettua anche un New con la perdita dei programmi in BASIC.

Una volta lanciata la routine apparentemente non accade nulla, ma se durante un programma in BASIC effettuiamo una CALL 39936, A\$, l'Apple si predispose ad accettare una riga in INPUT e a depositarla nella variabile A\$ (o un'altra a piacere). La riga può essere composta da qualsiasi

carattere ed essere lunga al massimo 240 caratteri, un Bip ci segnala gli ultimi dieci caratteri. La tastiera è tipo macchina da scrivere, cioè in minuscolo, e con lo shift si ottengono le maiuscole. I caratteri speciali che si trovano sopra la P, la N e la M si ottengono premendo CTRL + SHIFT + il

```

3FED      1      ORG #3FED
3FED      2      OBJ #3FED
3FED      3      ;
3FED      4      ; ROUTINE DI INPUT
3FED      5      ; BY VALTER DI DIO
3FED      6      ;
3FED      7      LOCO EPZ #E
3FED      8      DIFF EPZ #0D
3FED      9      CH EPZ #24
3FED     10      BASL EPZ #2B
3FED     11      LNPRG EPZ #75
3FED     12      NXTINL EPZ #7F
3FED     13      NXTINH EPZ #80
3FED     14      VARL EPZ #85
3FED     15      VARH EPZ #86
3FED     16      NWLNH EPZ #87
3FED     17      NWLNH EPZ #88
3FED     18      OCHRH EPZ #88
3FED     19      OCHRH EPZ #89
3FED     20      ;
3FED     21      GETCHR EQU #B1
3FED     22      NXTCHR EQU #B7
3FED     23      ;
3FED     24      IN EQU #200
3FED     25      SHIFT EQU #C063
3FED     26      BUFFER EQU #9D00
3FED     27      ;
3FED     28      DOS EQU #3D3
3FED     29      LET EQU #DA7B
3FED     30      COMMA EQU #DEBE
3FED     31      RECLN EQU #D533
3FED     32      FINDV EQU #DFE3
3FED     33      STRNG EQU #E3ED
3FED     34      VALUE EQU #E73D
3FED     35      KEYIN EQU #FD1B
3FED     36      COUT EQU #FDED
3FED     37      CROUT EQU #FDBE
3FED     38      CLREQL EQU #FC9C
3FED     39      BELL EQU #FF3A
3FED     40      ;
3FED     41      LDX #B0
3FED     42      LOOP LDA #4000, X
3FED     43      STA #9C00, X
3FED     44      DEX
3FED     45      BNE LOOP
3FED     46      LDA #99B
3FED     47      STA BUFFER+1
3FED     48      JMP DOS
3FED     49      ;
3FED     50      ORG #9C00
3FED     51      OBJ #4000
3FED     52      ;
3FED     53      JSR COMMA
3FED     54      JSR GETLN
3FED     55      JSR RECLN
3FED     56      LDA IN
3FED     57      STA LNPRG
3FED     58      STX NXTINL
3FED     59      STY NXTINH
3FED     60      JSR FINDV
3FED     61      STA VARL
3FED     62      STY VARH
3FED     63      LDA OCHRH
3FED     64      LDY OCHRH
3FED     65      STA NWLNH
3FED     66      STY NWLNH
3FED     67      LDY NXTINL
3FED     68      STX OCHRH
3FED     69      STY OCHRH
3FED     70      JSR GETCHR
3FED     71      LDA OCHRH
3FED     72      LDY OCHRH
3FED     73      STA LOCO
3FED     74      STA DIFF
3FED     75      JSR STRNG
3FED     76      JSR VALUE
3FED     77      JSR LET
3FED     78      JSR NXTCHR
3FED     79      LDA OCHRH
3FED     80      LDY OCHRH
3FED     81      STA NXTINL
3FED     82      STY NXTINH
3FED     83      LDY NWLNH
3FED     84      STA OCHRH
3FED     85      STY OCHRH
3FED     86      END      RTS

9C51 BD0002 87 NOTCR LDA IN, X
9C54 20EDFD 88 JSR COUT
9C57 C988 89 CMP #98B
9C59 F01D 90 BEQ BCKSPC
9C5B C998 91 CMP #99B
9C5D F00A 92 BEQ CANCELL
9C5F E0E5 93 CPX #E5
9C61 D003 94 BNE NOTCR1
9C63 203AFF 95 JSR BELL
9C66 EB 96 NOTCR1 INX
9C67 D013 97 BNE NXTCH
9C69 A9DC 98 CANCEL LDA #"\
9C6B 20EDFD 99 JSR COUT
9C6E 20BEFD 100 GETLNZ JSR CROUT
9C71 A9BF 101 GETLN LDA #"?
9C73 20EDFD 102 JSR COUT
9C76 A201 103 LDX #B1
9C78 8A 104 BCKSPC TXA
9C79 F0F3 105 BEQ GETLNZ
9C7B CA 106 DEX
9C7C A424 107 NXTCH LDY CH
9C7E B128 108 LDA (BASL), Y
9C80 48 109 PHA
9C81 C9E0 110 CMP #E0
9C83 9002 111 BCC NOINV
9C85 291F 112 AND #B1F
9C87 293F 113 NOINV AND #3F
9C89 0940 114 ORA #40
9C8B 9128 115 STA (BASL), Y
9C8D 68 116 PLA
9C8E 20A49C 117 JSR GETCH
9C91 C995 118 CMP #995
9C93 D002 119 BNE ADDINP
9C95 B128 120 LDA (BASL), Y
9C97 9D0002 121 ADDINP STA IN, X
9C9A C9BD 122 CMP #9BD
9C9C D0B3 123 BNE NOTCR
9C9E 209FCF 124 JSR CLREQL
9CA1 4CBEFD 125 JMP CROUT
9CA4 201BFD 126 GETCH JSR KEYIN
9CA7 C9C0 127 CMP #C0
9CA9 B039 128 BCS MINUS
9CAB 90A0 129 CMP #9A0
9CAD 9001 130 BCC CTRL
9CAF 60 131 RTS
9CB0 2C63C0 132 CTRL BIT SHIFT
9CB3 1001 133 BPL RTS1+1
9CB5 60 134 RTS1 RTS
9CB6 C985 135 CMP #985
9CB8 D002 136 BNE C1
9CBA A9FD 137 LDA #9FD
9CBC C98F 138 C1 CMP #98F
9CBE D002 139 BNE C2
9CC0 A9FC 140 LDA #9FC
9CC2 C995 141 C2 CMP #995
9CC4 D002 142 BNE C3
9CC6 A9E0 143 LDA #9E0
9CC8 C989 144 C3 CMP #989
9CCA D002 145 BNE C4
9CCC A9FE 146 LDA #9FE
9CCE C981 147 C4 CMP #981
9CD0 D002 148 BNE C5
9CD2 A9FB 149 LDA #9FB
9CD4 C980 150 C5 CMP #980
9CD6 F009 151 BEQ CONV
9CDB C99E 152 CMP #99E
9CDA F005 153 BEQ CONV
9CDC C99D 154 CMP #99D
9CDE F001 155 BEQ CONV
9CE0 60 156 RTS
9CE1 0940 157 CONV ORA #40
9CE3 60 158 RTS
9CE4 C9DE 159 MINUS CMP #"\
9CE6 F004 160 BEQ CONV1
9CE8 C9DD 161 CMP #"?
9CEA D005 162 BNE RTS2+1
9CEC 0910 163 CONV1 ORA #B10
9CEE 29EF 164 AND #9EF
9CF0 60 165 RTS2 RTS
9CF1 C9C0 166 CMP #9C
9CF3 D003 167 BNE RTS3+1
9CF5 0910 168 ORA #B10
9CF7 60 169 RTS3 RTS
9CF8 2C63C0 170 BIT SHIFT
9CFB 1002 171 BPL RTS4
9CFD 0920 172 ORA #B20
9CFF 60 173 RTS4 RTS
3FED     86      END      RTS
    
```

Figura 4 - Sorgente LISA del programma di INPUT. Se non si dispone di un assembler caricare i codici della seconda colonna a partire dalla locazione \$3FED e salvare con il nome di INPUT. PLUS, AS 3FED, LS 115.

```

10 DIM A$(100):I = 0
20 D$ = CHR$(4): REM CTRL D
30 HOME : TEXT
40 IF PEEK(39937) + PEEK(399
38) = 412 THEN 60
50 PRINT D$;"BRUN INPUT.OBJ"
60 I = I + 1
70 CALL 39936,A$(I)
110 IF A$(I) < > "" GOTO 60
120 PRINT
130 INPUT "WHAT FILE NAME? ";I$
140 PRINT D$;"OPEN ";I$
150 PRINT D$;"DELETE ";I$
160 PRINT D$;"OPEN ";I$
170 PRINT D$;"WRITE ";I$
190 FOR J = 1 TO I - 1
200 PRINT A$(J)
210 NEXT J
220 PRINT D$;"CLOSE ";I$
    
```

Figura 5 - Dump esadecimale della routine semplificata per chi non dispone della EPROM delle minuscole. Caricare a partire dalla locazione \$300 e salvare con BSAVE INPUT. PLUS, AS 300, LSBO.

```

0300- 20 BE DE JSR #DEBE
0303- 20 2C D5 JSR #D52C
0306- AD 00 02 LDA #0200
0309- B5 75 STA #75
030B- 86 7F STX #7F
030D- 84 B0 STY #B0
030F- 20 E3 DF JSR #DFE3
0312- 85 85 STA #85
0314- 84 86 STY #86
0316- A5 BB LDA #BB
0318- A4 B9 LDY #B9
031A- 85 87 STA #87
031C- 84 BB STY #BB
031E- A6 7F LDX #7F
0320- A4 B0 LDY #B0
0322- 86 BB STX #BB
0324- 84 B9 STY #B9
0326- 20 B1 00 JSR #00B1
0329- A5 BB LDA #BB
032B- A4 B9 LDY #B9
032D- 85 0E STA #0E
032F- 85 0D STX #0D
0331- 20 ED E3 JSR #E3ED
0334- 20 3D E7 JSR #E73D
0337- 20 7B DA JSR #DA7B
033A- 20 B7 00 JSR #00B7
033D- A5 BB LDA #BB
033F- A4 B9 LDY #B9
0341- 85 7F STA #7F
0343- B4 B0 STY #B0
0345- A5 87 LDA #87
0347- A4 BB LDY #BB
0349- 85 BB STA #BB
034B- 84 B9 STY #B9
034D- 60 RTS
    
```

Figura 6 - Programma in Basic per la creazione di file EXEC. La routine di INPUT. PLUS deve già essere in memoria!

tasto desiderato; nello stesso modo si ottengono le lettere accentate. La pressione del CTRL X annulla la riga scritta e ripropone il prompt senza tornare al programma. Solo il tasto di RETURN consente di uscire dalla routine. Per le modifiche si usano le due frecce come al solito. Nella locazione 40050 si trova il codice ASCII del prompt (di default è un punto interrogativo) che viene presentato dalla routine di input, se non lo si vuole si può fare una POKE 40050,128; qualunque altro valore verrà interpretato come prompt e stampato in testa al rigo.

Chi non avesse (ancora?) la nostra EPROM Apple-Minus per le minuscole può fare a meno di una gran parte di programma; la routine semplificata (vedi figura 5) entra comodamente nella pagina tre.

In figura 6 trovate il listato in Basic di un programmino che serve per creare dei File EXEC, quei file cioè che vengono eseguiti dall'Apple come se il loro contenuto fosse direttamente battuto da un operatore sulla tastiera del computer.