

EXERCÍCIOS

**Passo
a.
Passo**

MICROSOFT

LIÇÃO 1 PARTE 1

RESUMO

Olhe para a tela de seu computador. Quando você digita algo no seu teclado nada ocorre, até que você pressione "CR, RETURN", que se localiza à sua direita. Quando você digitar o "zero", não use jamais a letra "O". O computador possui a tecla correta para o número, o mesmo acontecendo quando for digitar o número 1 (não use a letra L). Se você se confundir e quiser modificar o que digitou anteriormente, pressione a tecla de retrocesso, que também está à sua direita.

A seguir um exemplo de como é a memória do computador .

```
0F00 21 00 45 4D 11 00 0E CD 1F 0F B9 CA 08 0F 77 00
0F10 80 47 23 18 7A B3 CA 29 0F CD 1F 0F C3 0F 0F DB
0F20 00 E6 40 CA 1F 0F DB 01 C9 AF B8 C2 2F 0F 76 FB
0F32 C3 30 0F 00 00 00 00 00 00 00 00 00 00 00 00
```

A seguir um exemplo de como é, a linguagem de máquina (Assembler) em um programa,

```
FF00 D8
FF01 58
FF02 AD 7F
FF04 8C 12 D0
FF07 A9 A7
FF09 8D 11 D0
FF0C 8D 13 D0
FF0F C9 DF
FF11 FD 13
FF13 C9 9B
FF15 FD 03
FF17 C8
FF18 10 0F
FF1A A9 DC
```

INTERPRETADOR: O Interpretador é um tradutor. Seu trabalho é traduzir a linguagem BASIC, para uma linguagem inteligível pelo computador para poder executar os comandos e instruções por ele traduzidos.

BASIC significa: BEGINER'S ALL PURPOSE SYMBOLIC INSTRUCTION CODE.

BASIC usa palavras em inglês. A seguir temos exemplos de comandos e instruções:

LICÃO 1 PARTE 2

RESUMO

Usando PRINT em Aritmética.

Digit: PRINT 10+6

0 computador imprime: 16

Entre aspas os números são tratados como se fossem letras.

```
Digit:          PRINT "8+9"
```

0 computador imprime: 8+9

Aritmeticamente em ordem de operação:

A. * MULTIPLICAÇÃO

B. / DIVISÃO

C. + ADIÇÃO

D. - SUBTRAÇÃO

Exemplo: 10 - 50 / 2 '4 8 * 3

1º passo: $10 - 25 + 8 \times 3$

2º passo: $10 - 25 + 24$

3º passo: $-15 \quad + 24$

40 passo: 9

Quando se usa um parêntesis, a operação dentro destes é feita primeiramente :

Exemplo: $3 * \{ 2 + 5 \}$

1º passo: $3 * 7$.

2º passo: 2

Quando parêntesis são usados :

Primeiro: MULTIPLICAÇÃO e DIVISÃO entre parentesis são executadas,

Segundo: ADIÇÃO e SUBTRAÇÃO entre parentesis são executadas.

Terceiro: MULTIPLICAÇÃO e DIVISÃO fora de parentesis são executadas,

Quarto: ADIÇÃO e SUBTRAÇÃO fora de parentesis são executadas.

Exemplo: $\{ 1 + 4 * 3 \} + 2 * \{ 3 + 3 \}$

10 passo: $\{ 1 + 12 \} + 2 * \{ 3 + 3 \}$

2º passo: $\{ 13 \} + 2 * \{ 3 + 3 \}$

3º passo: $\{ 13 \} + 2 * \{ 6 \}$

4º passo: 13 + 12

5º passo: 25

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1

PROBLEMAS PRÁTICOS 1,2

Vamos praticar usando comando PRINT. Digite cada um dos seguintes comandos e pressione RETURN/CR, e veja o que o computador responde.

OBS: A tecla * é a direita da tecla zero. Pressione a tecla SHIFT e segure-a enquanto digita o *.

Se você acha que aprendeu as regras bem a contento, tente os próximos exemplos. Se resolvê-los passe adiante:

```
PRINT 4 / 2 + 6
PRINT 3 * 2 + 1
PRINT 10 / 5 + 4 - 5
PRINT 8 * 2 - 6 * 2
PRINT { 3 + 6 } / { 3 + 3 }
PRINT { 10 * 4 - 4 } / { 2 * 8 - 4 }
PRINT 2 * { 6 * 2 } - 20
PRINT 25 + { 5 * 5 } / 5
PRINT { 6 * 4 + 4 } / { 3 + 4 } - 4
PRINT 100 / { 10 + 10 } * { 2 * 4 } - 1
```

RUN TESTE 1

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1

LICAO 2 PARTE 1

RESUMO

Este é um programa:

```
10 PRINT "A"
```

20 PRINT

30 PRINT "B"

Esta linhas não são um programa:

PRINT "A"

PRINT

PRINT "B"

Instruções em um programa devem conter números de linha. Os números de linha podem ser incrementados, mas não podem ultrapassar 64.000 linhas.

É uma boa idéia numerar as linhas de 10 em 10, assim deixa-se espaço para adicionar outras linhas que se queira inserir.

O COMANDO RUN

O comando RUN executa o programa começando pela linha de menor valor numérico e trabalha até a linha de maior valor. O RUN não limpa da memória o programa, só o executa.

O COMANDO LIST

O comando LIST lista todo e ou parte de um programa em ordem numérica de -
linhas. Alguns exemplos:

LIST lista todo o programa

LIST 10 lista a linha de número 10

LIST 1,100 lista as linhas de número 1 a 100

LIST 100, lista a linha de número 100 e todas a seguir

[illegible]

LICÃO 2 PARTE 2

RESUMO

Para corrigir uma linha em um programa:

A. Digite o número da linha e pressione RETURN (CR); isto remove a linha.

B. Re-digite a linha corretamente e pressione RETURN (CR)

COMANDO NEW

O comando NEW é utilizado para remover um programa inteiro.

O COMANDO DEL

O comando DEL é utilizado para cancelar linhas de um programa. Ex:

DEL 5,10 (cancela linhas de 5 até 10)

INSTRUÇÃO END

A instrução END põe fim ao programa

INSTRUÇÃO STOP

A instrução STOP para um programa em um determinado número de linha.

Os comandos somente são executados quando é pressionado RETURN (CR).

Instruções são executadas apenas quando o programa é executado, e twm núme

ros de linha.

Comandos como NEW e LIST, são raramente usados como instrução.

PROBLEMAS

PROBLEMAS PRÁTICOS 2,1 E 2,2

Digite NEW e entre com esses programas práticos:

```
10 PRINT "INÍCIO"
20 PRINT "MEIO"
30 PRINT "FIM"
40 END
```

Digite NEW e faça a listagem (LIST) do programa:

```
10 PRINT " 8 * 5 "
20 PRINT 8 * 5
30 STOP
40 END
```

Use o comando LIST

Use o comando RUN

Digite NEW

```
5 PRINT      " Brincando com o número 2 "
10 PRINT     " 2 + 2 = "
20 PRINT      2 + 2
30 PRINT     " 2 - 2 "
40 PRINT      2 - 2
50 PRINT     " 2 * 2 ="
60 PRINT      2 * 2
70 PRINT     " 2 / 2 ="
80 PRINT      2 / 2
90 PRINT     "FEITO ?!"
100 STOP
```

Use o comando LIST

Use o comando RUN (Não digite NEW ainda)

Adicione as seguintes instruções no último programa sem re-digitar as linhas de instruções acima:

```
8 PRINT "Adicionando"
25 PRINT "Subtraindo"
45 PRINT " Multiplicando"
65 PRINT " Dividindo"
```

Então use o comando LIST e RUN para executar o programa.

RUN TESTE 2

PROBLEMAS


```
70 PRINT P
```

```
80 END
```

Inclua instruções a seguir marcadas com as flechas para expandir o programa com os títulos discriminados abaixo:

```
10 LET L = 5
```

```
20 LET P = L+L+L+L
```

```
→ 30 PRINT " PERÍMETRO DE UM QUADRADO "
```

```
→ 40 PRINT " LADO ="
```

```
→ 50 PRINT L
```

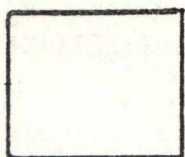
```
→ 60 PRINT " PERÍMETRO"
```

```
70 PRINT P
```

```
80 END
```

No programa acima mude a linha 10 e ache o perímetro de um quadrado cujo lado seja 12; depois lado = 15; e vá mudando o valor do lado.

Programe agora para achar a ÁREA de um quadrado:



$$\text{ÁREA} = \text{Lado} \times \text{Lado}$$

```
10 LET L = 14
```

```
20 LET A = LxL
```

```
30 PRINT "ÁREA DO QUADRADO"
```

```
40 PRINT "LADO ="
```

```
50 PRINT L
```

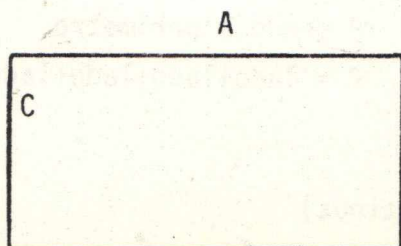
```
60 PRINT "ÁREA ="
```

```
70 PRINT A
```

```
80 END
```

Mude a linha 10 para achar áreas de outros quadrados.

Programe para achar a área de um retângulo:



```
10 LET C = 3
```

```
20 LET H = 2
```


Modifique as linhas 10 e 20 para encontrar outras áreas.

1 kilômetro = 0,621 milhas

Modifique a linha 10 para converter 60 Km em Milhas.

.....

1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431	2432	2433	2434	2435	2436	2437	2438	2439
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------

Digite os seguintes programas, para ver como as instruções de entrada fun-

```
10 PRINT "ESTE PROGRAMA FAZ A SOMA DE DOIS NÚMEROS"
```

(continua)

60 END

```
10 PRINT
20 PRINT "PROGRAMA PARA CONVERTER PÉS EM METROS"
30 PRINT "QUANTOS PÉS?"
40 INPUT F
50 LET N = 0.3048 * F
60 PRINT "CONVERTENDO PARA METROS:"
70 PRINT M
80 END
```

- A. Subtrair dois números
- B. Multiplicar dois números
- C. Dividir dois números
- D. Adicionar três números
- E. Multiplicar três números

RUN TESTE 3

[illegible]

RESUMO

A instrução GOTO seguida de um número de linha, transfere o controle para o número da linha que foi escrito após o GOTO.

GOTO pode criar um LOOP (Ciclo) infinito em um programa. Para quebrar esse LOOP, pressione a tecla CTRL segurando-a enquanto pressiona a letra C, aí então ocorre uma parada.

Exemplo:

```
10 PRINT "PROGRAMA AMOSTRA"
```

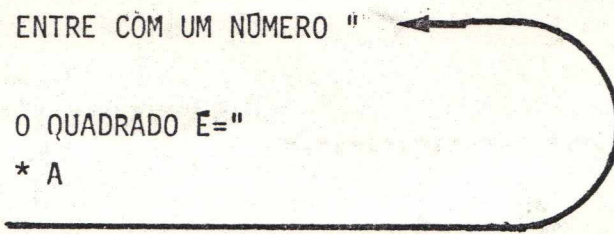
```
20 PRINT " ENTRE COM UM NÚMERO "
```

```
30 INPUT A
```

```
40 PRINT " O QUADRADO É="
```

```
50 PRINT A * A
```

```
60 GOTO 20
```



IF...THEN

IF dá uma condição. Se a condição for satisfeita tudo o que estiver na linha após THEN é executado. Se a condição não foi satisfeita, então tudo após o THEN será ignorado.

Ilustração:

```
IF A=30 THEN STOP
```

condição isto acontece somente se a condição foi satisfeita

IF...THEN pode ser usado para imprimir palavras diferentes, dependendo de uma condição. Por exemplo:

```
10 INPUT A
```

```
20 IF A=1 THEN PRINT "SIM"
```

```
30 IF A=2 THEN PRINT "NÃO"
```

```
40 IF A=3 THEN PRINT "TALVEZ"
```

```
50 END
```

A instrução IF...THEN sempre vai parar um programa repetitivo. Este programa vai parar quando nós damos entrada a INPUT -1.

```
10 PRINT "PROGRAMA PARA SOMAR DOIS NÚMEROS"
```

```
20 PRINT "ENTRE COM DOIS NÚMEROS"
```

```
30 INPUT A,B
```

```
40 PRINT A+B
```

```
50 IF A=-1 THEN END
```

```
60 GOTO 20
```


Note os vários modos que uma instrução pode entrar:

```
10 IF A=1 THEN GOTO 50
20 IF A=1 THEN 50
30 IF A=1 THEN GOTO 50
```

Você dá entrada: 10 GO TO 50

O computador dá a listagem: 10 GOTO 50 -

Você dá entrada: 10 GOTO 50

O computador dá a listagem: 10 GOTO 50

.....

PROBLEMAS PRÁTICOS 4,1

Corra o seguinte programa no computador e tente pará-lo. Você pode sempre desligar o computador caso esqueça como parar o LOOP (ciclo infinito).

```
10 LET A= 10
20 PRINT A
30 GOTO 10
40 END
```

Veja o que ocorre quando este programa é executado (RUN):

```
10 GOTO 20
20 GOTO 10
30 END
```

Tente este programa:

```
5 PRINT "DIGITE UM NÚMERO"
10 INPUT A
20 IF A=1 THEN PRINT "SIM"
30 IF A=2 THEN PRINT "NÃO"
40 IF A=3 THEN PRINT "TALVEZ"
50 IF A=4 THEN STOP
60 GOTO 5
```

.....

o controle a várias -

同 步 電 壓 變 化 率

00A 00B 00C 00D 00E 00F 00G 00H 00I 00J 00K 00L 00M 00N 00O 00P 00Q 00R 00S 00T 00U 00V 00W 00X 00Y 00Z

RESUMO

CONTADORES

Quando o programa a seguir começa, A é igualado a zero. Então o programa adiciona 1 ao A. Na linha 25 é verificado se A é igual a 5. Desde que não o seja, o programa traz o ciclo de volta (LOOP) para a linha 10 e adiciona 1 ao A novamente. Isto continua até que A seja igual a 5.

```

5 LET A = 0
10 LET A = A+1      ← contador (COUNTER)
20 PRINT A
25 IF a = 5 THEN END
30 GOTO 10
    
```

REM

Diz ao interpretador BASIC para ignorar a linha na qual está inserido.

```

10 REM ESTE É UM PROGRAMA SIMPLES
20 PRINT 2 + 2
30 END
RUN
O computador imprime: 4
    
```

.....

PROBLEMAS PRÁTICOS 5,1

Observe o que ocorre quando este programa é colocado em execução:

```

10 LET A = A+1
20 PRINT A
30 GOTO 10
    
```

Veja como este programa pára por si mesmo:

```

10 LET A = A+1
20 PRINT A
30 IF A = 20 THEN PRINT "ESTAMOS NO MEIO DO CAMINHO"
40 IF A = 40 THEN PRINT "ACABAMOS DE CHEGAR"
50 IF A = 40 THEN END
60 GOTO 10
    
```

.....

Pratique o seguinte programa:

```
10 PRINT "ISTO É UM JOGO DE ADVINHAR NÚMEROS"
20 PRINT " PARA DUAS PESSOAS. UMA DELAS INTRODUZ "
30 PRINT "UM NÚMERO SECRETO ENTRE 1 E 100"
40 PRINT "E LIMPA A TELA. A OUTRA PESSOA"
50 PRINT "TENTA DESCOBRIR O NÚMERO SECRETO"
60 PRINT "O PROGRAMA IRÁ DANDO DICAS PARA"
70 PRINT "FACILITAR A DESCOBERTA DO NÚMERO."
80 PRINT "INTRODUZA O NÚMERO 70"
90 INPUT N
91 PRINT
92 PRINT
93 PRINT
94 PRINT
95 PRINT
96 PRINT
97 PRINT
98 PRINT
99 PRINT
100 PRINT
101 PRINT
102 PRINT
103 PRINT
104 PRINT
105 PRINT
106 PRINT
107 PRINT
108 PRINT
109 PRINT
110 PRINT
111 PRINT
112 PRINT
113 PRINT
114 PRINT "COMECE A ADVINHAR....."
115 INPUT G
116 REM ADVINHAÇÃO CORRETA
120 IF G = N THEN 200
125 REM ADVINHAÇÃO INCORRETA
130 IF G > N THEN PRINT "MUITO ALTO, TENDE UM MENOR"
140 IF G < N THEN PRINT "MUITO BAIXO, TENDE UM MAIOR"
150 IF G < N THEN LET E = E + 1
180 GOTO 115
```


—, —, —, —, —, —, —, —, —, —, —, —

1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------

RESUMO

FUNÇÕES BIBLIOTECÁRIAS

$X = \text{INT}(Y)$ FUNÇÃO INTEIRO: Admite X igual a Y, eliminando seus decimais
 $X = \text{ABS}(Y)$ FUNÇÃO ABSOLUTA: Admite X igual a Y eliminando os sinais(+,-)
 $X = \text{RND}(1)$ FUNÇÃO RANDÔMICA: Admite X igual a um número randômico entre 0 e 1.
 $X = \text{SQR}(Y)$ FUNÇÃO RAIZ QUADRADA: Admite X igual a um número, o qual multiplicado por si mesmo resulta Y.

Para obter números inteiros entre 1 e 50 digite o programa abaixo:

10 X= RND (1)

20 X= X*50

OU

10 X= INT(50* RND(1))+ 1

30 X= INT(X)

40 X= X+1

Aninhando parêntesis:

$X = \{ 5 + \{ 2 * \{ 3 + 4 \} \} \}$

$X = \{ 5 + \{ 2 * \{ 7 \} \} \}$

$X = \{ 5 + \{ 2 * 7 \} \}$

$X = \{ 5 + \{ 14 \} \}$

$X = 5 + 14$

$X = 19$

Outras FUNÇÕES BIBLIOTECÁRIAS:

$X = \text{SIN}(Y)$ x é igual ao seno de y em radianos,

$X = \text{COS}(Y)$ x é igual ao cosseno de y em radianos,

$X = \text{TAN}(Y)$ x é igual à tangente de y em radianos,

$X = \text{ATN}(Y)$ x é igual à arctangente de y em radianos,

$X = \text{LOG}(Y)$ x é igual ao logarítmo natural de y,

.....

Digite os seguintes programas e depois RUN :

```

10 REM CONTANDO DE 5 EM 5 PELO MEIO DIFICIL
20 X = 0
30 X = X+1
40 IF X = 100 THEN END
50 IF X/5 = INT {X/5} THEN PRINT "'O CONTADOR VALE?"
60 IF X/5 = INT {X/5} THEN PRINT X
70 PRINT
80 GOTO 20

```

.....

```

10 REM PROGRAMA DE CARA E COROA
20 LET H = 0
30 LET T = -0
40 PRINT "'O COMPUTADOR LANCARA A MOEDA 10 VEZES"
50 LET N = RND {1}
55 LET N = INT {2*N}
60 IF N = 0 THEN PRINT "'CARA"
70 IF N = 0 THEN H = H + 1
80 IF N = 1 THEN PRINT "'COROA"
90 IF N = 1 THEN T = T + 1
100 IF H + T = 10 THEN 120
110 GOTO 50
120 PRINT "'TOTAL DE CARAS"
130 PRINT H
140 PRINT "'TOTAL DE COROAS"
150 PRINT T
160 END

```

.....

```

10 REM JOGOS DE ESCOLHA DE NUMEROS COM NUMERO RANDOMICO
20 PRINT "'O COMPUTADOR ESCOLHE UM NUMERO ENTRE"
30 PRINT "'O E 99.VOCE TENTERA ADVINHAR ESTE"
40 PRINT "'NUMERO."
45 REM O COMPUTADOR ESCOLHE O NUMERO
50 LET N = RND{1}
60 LET N = INT {100*N}
70 PRINT "'PRONTO.. ESCOLHA O NUMERO"
80 INPUT G

```

CONTINUA

RUN TESTE 5

- 21 -

RESUMO

Há um certo número de comandos e instruções neste computador, que afetam o modo de certas palavras e símbolos que são impressos na tela.

Para limpar a tela do vídeo, você pode usar a instrução HOME, Exemplo:

```
10 PRINT "PÁGINA ANTIGA"
20 HOME
30 PRINT "PÁGINA ATUAL"
40 END
```



O comando PRINT pode ser usado, para guardar espaços ou separar linhas na tela, Exemplo:

```
10 PRINT "A"
20 PRINT
30 PRINT "B"
40 END
```



■ RUN

A
B

Quando o interpretador BASIC assume uma vírgula agregada à instrução PRINT ele coloca qualquer caracter depois da vírgula na próxima "coluna" da tela. (Naturalmente a vírgula não pode ser inserida entre aspas senão o computador irá imprimir a mesma). Exemplo:

```
10 A = 1
20 B = 2
30 C = 3
40 PRINT A,B,C
50 END
```

RUN

1 2 3

Quando ponto e vírgula são inseridos em uma instrução PRINT, nenhum espaço será dado entre os caracteres impressos. Exemplo:

```
10 A = 1
20 B = 2
30 PRINT " A " ; " B "
40 PRINT A,B
50 END
```

RUN

AB

12

Ponto e vírgula e vírgula; dizem ao computador para continuar imprimindo na mesma linha até que haja mudança de linha (naturalmente).

```
10 PRINT " ATRÁS"
20 PRINT "EM CIMA"
30 END
```

RUN

ATRÁS
EM CIMA

```
10 PRINT "ATRÁS";
20 PRINT "EM CIMA"
30 END
```

RUN (Veja como é impresso):

ATRÁS EM CIMA

O comando TAB é um Tabulador, isto é, tabula espaços para se deixar brancos em uma linha.

```
PRINT TAB (15); "COMEÇAR"
```

```
5 REM USANDO VTAB E HTAB
```

```
10 HOME
```

```
20 PRINT "EM QUAL LINHA VOCE QUER ESCREVER ?"
```

```
30 INPUT A: IF A > 22 THEN PRINT "DEMAIS, TENTE NOVAMENTE"
```

```
35 IF A > 22 THEN GOTO 30
```

```
40 PRINT "EM QUAL COLUNA VOCE QUER ESCREVER ?"
```

```
50 INPUT B
```

```
60 IF B > 40 THEN PRINT "DEMAIS, TENTE NOVAMENTE"
```

```
65 IF B > 40 THEN GOTO 50
```

```
70 PRINT "O QUE VOCE QUER ESCREVER ?"
```

```
80 INPUT A $
```

```
90 HOME
```

```
100 VTAB A
```

```
110 HTAB B
```

```
120 PRINT A $
```

```
130 PRINT
```

```
140 PRINT
```

```
150 PRINT "VOCE QUER CONTINUAR ?(1 =SIM, 2 = NAO)"
```

```
160 INPUT A $
```

```
170 IF A$ = 1 "O" THEN END
```

```
180 GOTO 10
```

```
5 REM EXEMPLO DE USO DE VIRGULAS
```

```
10 REM HOME
```

```
20 PRINT
```

```
30 PRINT
```

```
40 PRINT "QUADRADOS E CUBOS DE NUMEROS"
```


[The following text is extremely faint and largely illegible due to low contrast and blurring. It appears to be a long sequence of words or phrases arranged horizontally.]

No seguinte programa digite palavras que possam ser introduzidas em uma variável STRING.Exemplo:

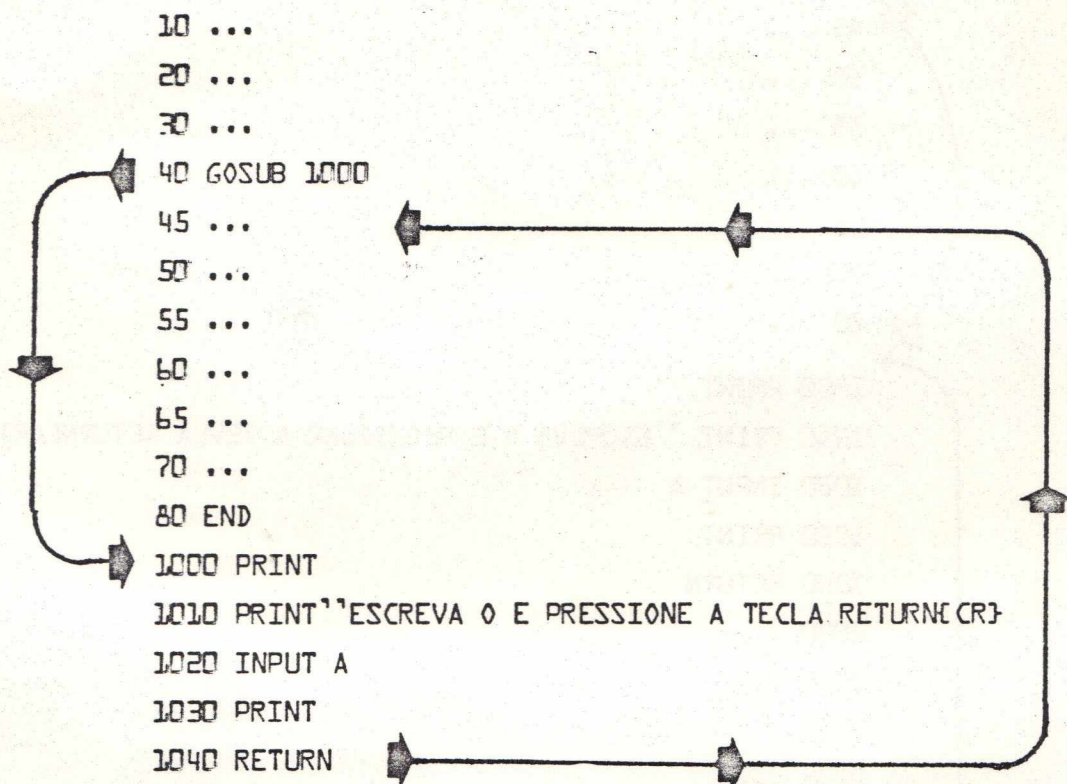
```
10 INPUT A$
20 PRINT
30 PRINT "ISTO QUE VOCE ESCREVEU E UM TIPO DE VARIÁVEL STRING."
40 PRINT A$
50 END
```

```
10 REM RESPONDA AS PERGUNTAS DESTES PROGRAMAS
15 HOME
20 PRINT "QUAL E SEU NOME ?"
30 INPUT A$
40 PRINT
50 PRINT A$ ; " | QUE NOME MAIS ESTRANHO !"
60 PRINT "ONDE VOCE VIVE ?"
70 INPUT B$
80 PRINT B$ ; " ? EU TENHO AMIGOS QUE TAMBEM VIVEM EM" ; B$ ; " ."
85 PRINT
90 PRINT "OK , " ; A$ ; " ,PODE VOLTAR AGORA PARA" ;
95 PRINT " " ; B$ ; " ."
100 END
```

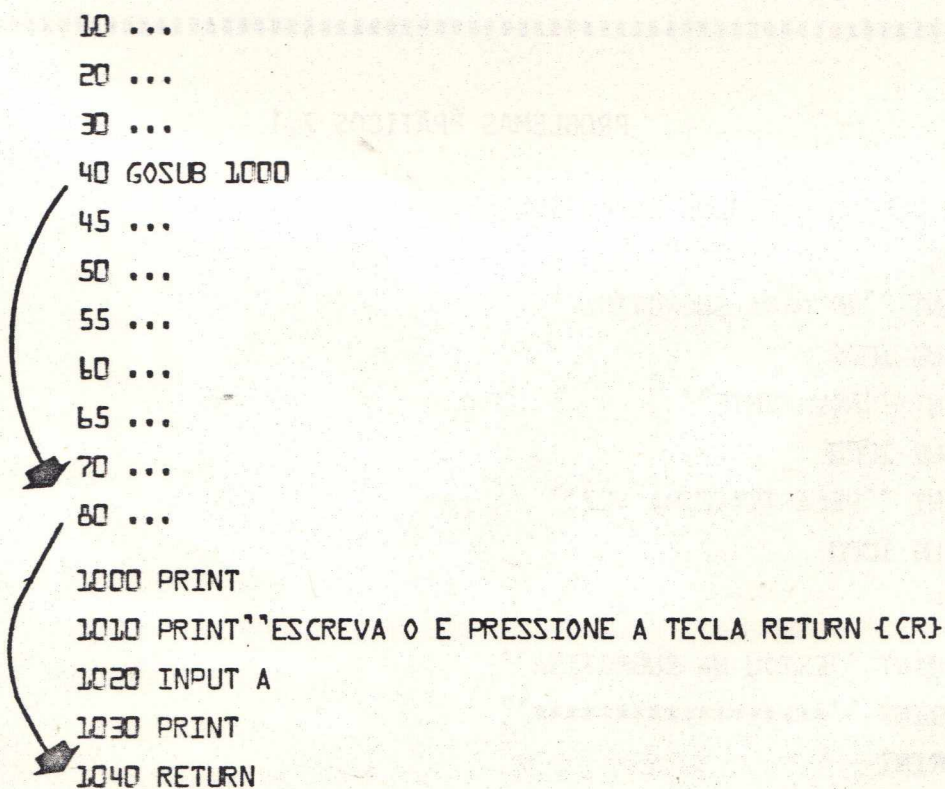
```
5 HOME
10 REM PROGRAMA CONTADOR DE ESTORIA
20 PRINT "ESCREVA O NOME DE UMA COISA: "
30 INPUT A$
40 PRINT "ESCREVA O NOME DE UMA PARTE DO CORPO: "
50 INPUT B$
60 HOME
70 PRINT "VOU CONTAR UMA ESTORIA:"
80 PRINT
90 PRINT "JOAO E JOSE"
100 PRINT "FORAM PASSEAR."
105 PRINT "MAS JOSE TRPECOU NO {A}" ; A$
110 PRINT "E CAIU NO CHAO."
120 PRINT "FOI NESTA OCASIAO QUE JOSE"
130 PRINT "QUEBROU O {A}" ; B$
150 END
```


RESUMO

Usando GOSUB :

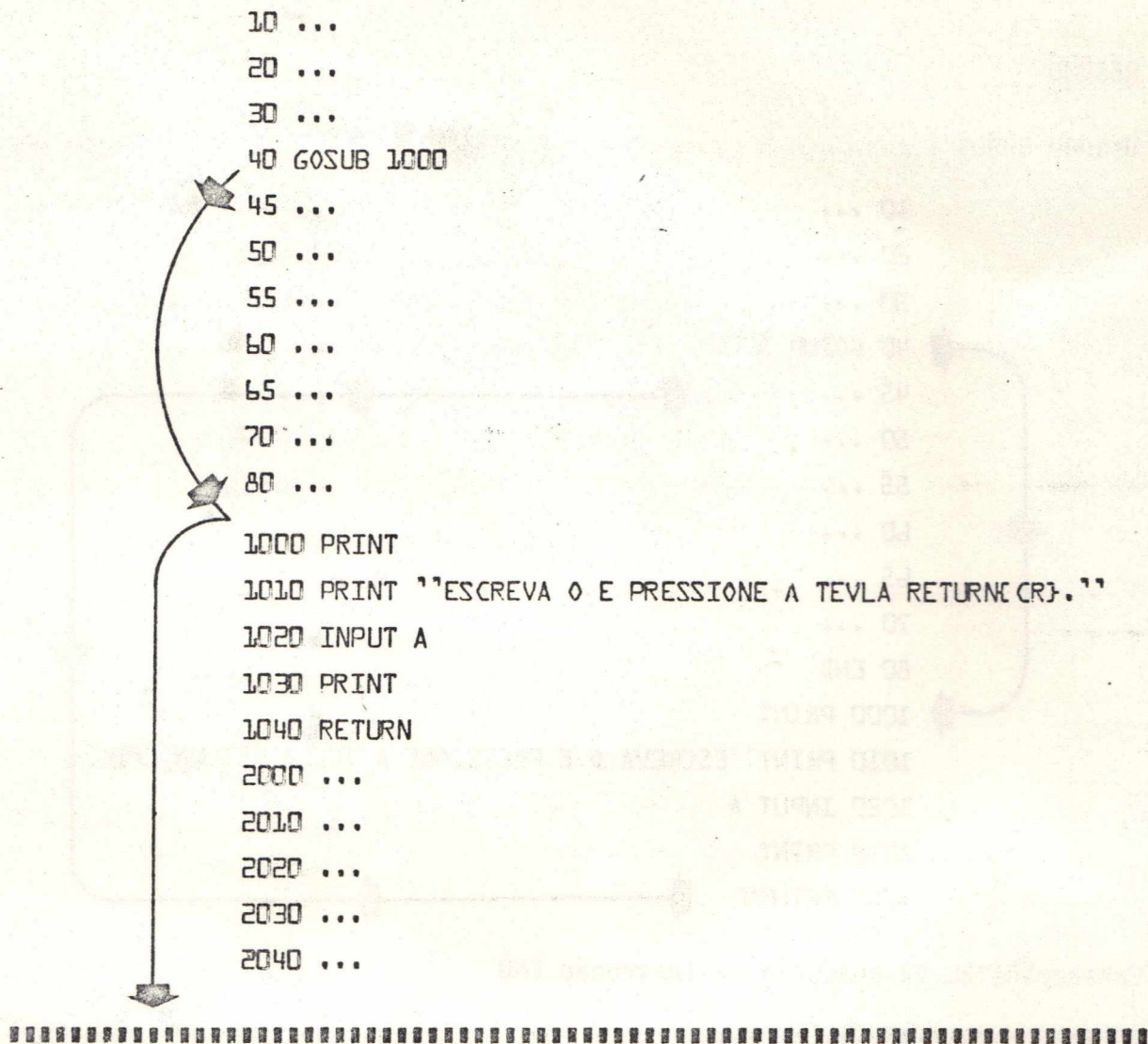


Consequências da ausência da Instrução END



? RETURN SEM GOSUB : ERRO EM 1040

Consequências da ausência da Instrução RETURN



PROBLEMAS PRÁTICOS 7,1

```

10 REM DEMONSTRACAO DE SUBROTINA
15 HOME
20 PRINT "IR PARA SUBROTINA"
30 GOSUB 1000
40 PRINT "NOVAMENTE"
50 GOSUB 1000
60 PRINT "PELA TERCEIRA VEZ"
70 GOSUB 1000
80 STOP

1000 PRINT "ESTOU NA SUBROTINA"
1010 PRINT "*****"
1020 PRINT
1030 RETURN

```

.....


```

10 REM PROGRAMA ARITMETICO COM SUBROTINA
15 HOME
20 PRINT "QUAL E SEU NOME ? "
30 INPUT A$
40 PRINT
50 PRINT A$ ; " ,FACA OS SEGUINTE PROBLEMAS"
55 E = 0
57 PRINT
60 LET A = INT {10 * RND {1}}
70 LET B = INT {10 * RND {1}}
80 PRINT A ; "X" ; B ; "="
90 INPUT C
100 IF C = A*B THEN 140
110 LET E = E+1
112 IF E = 1 THEN W = W + 1
114 IF E = 3 THEN PRINT "A RESPOSTA CORRETA E" : "A*B
115 IF E = 3 THEN PRINT "ERRADO.TENTE NOVAMENTE"
116 IF E = 3 THEN GOTO 55
117 PRINT
118 GOTO 80
140 GOSUB 1000
150 REM CONFERIR SE O ESTUDANTE FEZ 10 PROBLEMAS
160 IF {W + R} = 9 THEN 200
170 GOTO 60
200 PRINT A$ ; " .SEU PLACAR:" ; R ; "CERTA(S) DE 10 QUESTOES"
210 END

1000 REM SUBROTINA
1010 LET Q = INT{3*RND{1}}
1020 IF Q = 0 THEN PRINT "CORRETO"
1030 IF Q = 1 THEN PRINT "ACERTOU," ; A$
1040 IF Q = 2 THEN PRINT "EXCELENTE"
1050 R = R + 1
1060 PRINT
1070 RETURN

```

.....

RESUMO

READ...DATA (LÊ...DADOS)

```
10 READ X
20 PRINT X
30 GOTO 10
40 DATA 1,2,3,4,5,6
```

RUN

```
1 |
2 |
3 |
4 |
5 | — o computador imprime X
6 |
```

? FALTA DADOS - ERRO (FIM DE DADOS)

RESTORE

RESTORE faz com que o computador use DATA novamente:

```
10 READ Y
20 PRINT Y
30 IF Y = -1 THEN RESTORE
40 GOTO 10
50 DATA 1,2,3,-1
```

RUN

impressão

```
1 |
2 |
3 |
-1 | — a partir daqui RESTORE volta
1 | ao início da leitura de
2 | DATA.
3 |
-1 |
```

Aqui o computador lê um dado para X e o próximo para Y de cada vez.

```
10 READ X,Y
20 PRINT X,Y
30 GOTO 10
40 DATA 1,2,3,4,5,6
```

RUN

```
1 |
3 |
5 |
2 |
4 | — o computador imprime X,Y a cada vez.
6 |
```

? FALTA DADOS - ERRO (FIM DE DADOS) .O computador dará mensagens de erro por falta de dados.

READ ... DATA e variáveis STRING

```
10 READ X$
20 PRINT X$
30 GOTO 10
40 DATA UM,DOIS,TRÊS,VAI
```

RUN

? FALTA DE DADOS (FIM DE DADOS)

impressão

```
UM
DOIS
TRÊS
VAI
```


RUN

UM, DOIS, TRÊS, VAI

[illegible]

```
10 REM PRATICANDO COM READ...DATA
20 READ A
30 PRINT A
40 GOTO 10
50 DATA 5,6,7,8
```

35 IF A = 8 THEN RESTORE

15 LET B = 0

```
36 LET B = B+1
```

```
37 IF B = 8 THEN STOP
```

6 DATA 1,2,3,4

```
22 READ X,Y
```

```
23 PRINT X,Y
```

E elimine as linhas 35,36,37

RUN TESTE 7

[The following page contains faint, illegible markings or bleed-through from another document.]

RESUMO

Este é um exemplo de LOOP (ciclo) com contador. Para cada valor de A = 1 até 3 o computador imprime "****",

```
10 A = 0
20 A = A + 1
30 PRINT "****"
40 IF A = 3 THEN END
50 GOTO 20
```

RUN

```
***
***
***
```

Uma instrução FOR...NEXT, tem a mesma função de um contador para um LOOP. O seguinte programa tem como resultado a mesma impressão do programa anterior.

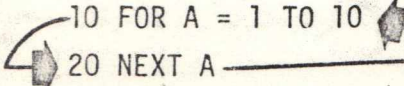
```
10 IF A = 1 to 3      _____ Dã o limite superior e inferior de A
20 PRINT "****"
30 NEXT A             _____ Completa o LOOP
```

RUN

```
***
***
***
```

Veja como FOR... NEXT trabalha em um LOOP. É dado um valor a A, e prossegue.

```
10 FOR A = 1 TO 10
20 NEXT A
```



Todas as instruções incluídas entre FOR...NEXT, são executadas toda vez que o LOOP é feito.

```
10 FOR A = 1 TO 2
```

```
{ 20 Z = A*A
  30 PRINT "A * A =" ; Z
  40 Q = A/O
  50 PRINT "A / Z =" ; Q
  60 PRINT
  70 NEXT A }
```

LINHAS de 20 a 60 são executadas cada vez que o valor de A aumenta.

RUN

```
A * A = 1
A / Z = 1
A * A = 4
A / Z = 0,5
```


36 NEXT Y

.....

A(0)	A(1)	A(2)	A(3)	A(4)	A(5)	A(6)
50	49	48	47	46	45	44

A(7)	A(8)	A(9)	A(10)	A(11)	A(12)
43	42	41	40	39	38

.....

PROBLEMAS PRÁTICOS 8,2

Digite esse programa:

```

10 PRINT "ESTE E O SISTEMA DE RESERVAS PARA"
20 PRINT "OS VOOS DE UMA COMPANHIA AEREA"
30 PRINT "EXISTEM 7 VOOS CADA UM COM 4 PASSAGENS"
40 PRINT "ATENCAO : TODOS OS VOOS SAO DE SEGUNDA CLASSE"
50 REM DEFINIR O NUMERO DE PASSAGENS
60 FOR X = 1 TO 7
70 LET A {X} = 4
80 NEXT X
90 PRINT "EM QUAL VOO VOCE DESEJA VIAJAR ?"
100 INPUT X
110 IF A{X} = 0 THEN PRINT "PERDAO.NAO TEMOS MAIS PASSAGENS.TENTE OUTRO
    VOO"
120 IF A{X} = 0 THEN 140
130 GOTO 90
140 REM REDUZIR O NUMERO DE PASSAGENS
150 LET A{X} = A{X}-1
160 PRINT "TUDO CERTO.A PASSAGEM PODE SER EMITIDA."
170 GOTO 90

```

.....

```

10 REM GUARDANDO UM NUMERO ALEATORIO EM UM VETOR
20 FOR C = 1 TO 10
30 LET {C} = INT(RND(1))*1000
40 NEXT
50 REM MOSTRANDO O VETOR
70 PRINT
80 FOR C = 1 TO 10
90 PRINT "N {"; C ";}="; N{C}
100 NEXT C

```

RUN TESTE 8

.....

RESUMO

Os programas podem ser simplificados usando LOOPS aninhados (compostos), ou seja um LOOP dentro de outro. O LOOP Z no programa B, substitue as 5 instruções PRINT do programa A.

PROGRAMA A

```
10 FOR A= 1 TO 3
20 PRINT "1*****"
30 PRINT "2*****"
40 PRINT "3*****"
50 PRINT "4*****"
60 PRINT "5*****"
70 NEXT A
```

PROGRAMA B

```
10 FOR A= 1 TO 3
20 FOR Z= 1 TO 5
30 PRINT Z;"*****"
40 NEXT Z
50 NEXT A
```

LOOPS compostos trabalham como engrenagens. Quando a roda dentada maior, gira uma vez, a roda dentada menor gira várias vezes.

A cada vez que o LOOP externo completa um ciclo, o LOOP interno completa cinco, como é disposto abaixo:

```
10 FOR A= 1 TO 100
20 FOR B= 1 TO 5          LOOP externo
30 NEXT B                LOOP interno
40 NEXT A
```

A colocação correta da instrução NEXT, é muito importante nos LOOPS FOR... NEXT. No programa abaixo A, a instrução NEXT do LOOP Z está fora de posição. Ao invés de aumentar o valor de Z, o computador retorna para a linha 10 e aumenta o valor de A cada vez que executa a linha 40. Quando a linha 50 é atingida, aparece então uma mensagem de erro.

PROGRAMA A

```
10 FOR A = 1 TO 4
20 FOR Z = 1 TO 2
30 PRINT Z
40 NEXT A
50 NEXT Z
```

RUN

```
1
1
1
1
1
? NEXT SEM FOR: ERRO EM 50
```

PROGRAMA B

```
10 FOR A = 1 TO 4
20 FOR Z = 1 TO 3
30 PRINT Z
40 NEXT Z
50 NEXT A
```

RUN

```
1
2
3
1
2
3
1
2
3
1
2
3
1
2
3
```

COLOCACAO CORRETA DA INSTRUCAO

NEXT

Digite esses programas:

```

10 FOR P = 1 TO 10
20 PRINT "PROBLEMA"
30 FOR N = 1 TO 2
40 PRINT "ESCOLHA UM NUMERO"
50 INPUT A {N}
60 NEXT N
70 PRINT A {1} ; " *" ; A {2} ; " = " ; A {1} * A {2}
75 PRINT
80 NEXT P
90 END

```

- - - - -

```

10 HOME
20 PRINT "QUANTAS PESSOAS TEM NA SUA FAMILIA ?"
30 INPUT N
40 FOR C = 1 TO N
50 PRINT "QUAL A IDADE DA PESSOA" ; C;"?"
60 INPUT A(C)
70 NEXT C
80 PRINT "AQUI ESTA O GRAFICO DAS IDADES."
90 FOR C = 1 TO N
100 PRINT "PESSOA" ; C
110 FOR X = 1 TO A(C)
120 PRINT " * ";
130 NEXT X
140 PRINT
150 NEXT C

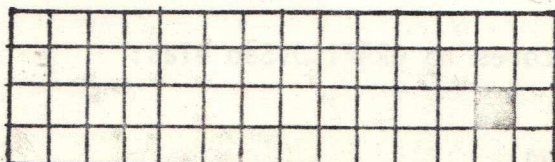
```

—

LIÇÃO 9 PARTE 2

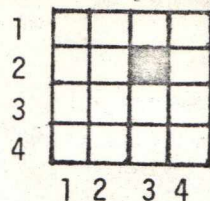
RESUMO

A tela do computador é dividida em quadrados, algo como papel gráfico.

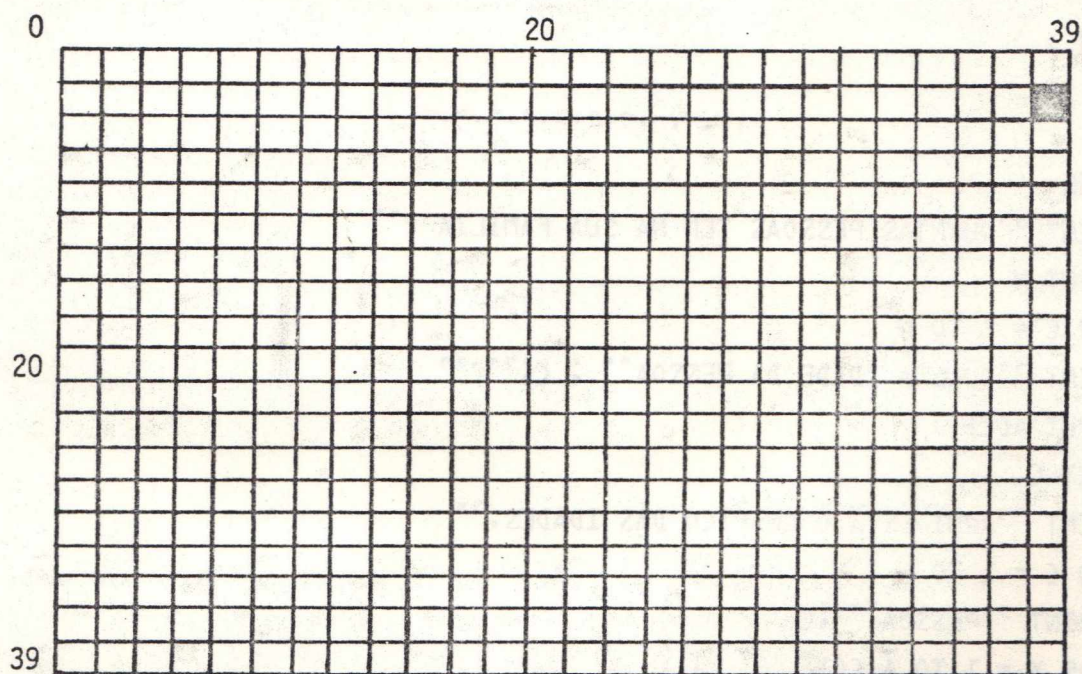


O quadrado está ligado

Cada quadradinho pode ser ligado ou desligado, Ligando ou desligando, criamos os chamados pontos gráficos. Cada quadradinho tem uma localização específica na tela. Aqui está a localização 3 vertical e 2 horizontal.



Quando formos criar gráficos, a tela do computador é dividida em 40 colunas verticais e 40 colunas horizontais. As colunas horizontais são numeradas de 0 a 39 da esquerda para a direita. As colunas verticais também são numeradas de 0 a 39 só que de cima para baixo na tela.



Cada quadradinho representa um local distinto na tela. Este que é mostrado acima está localizado na posição 39 da coluna horizontal e na posição 2 na coluna vertical. Quando é definida a posição de um ponto na tela, é sempre colocado o número da coluna horizontal na frente do número da coluna vertical. Uma vírgula separa os números. Exemplo:

39,2
horizontal vertical

Para usar nossos GRÁFICOS você deve entrar com o comando de GRÁFICOS, que é GR.

10 GR ← Prepara a tela para GRÁFICOS
20 COLOR = 6
30 PLOT 10,13
40 END

O segundo passo é especificar uma das 16 cores no GRÁFICO. São elas:

(continua)

0 preto	8 marrom
1 magenta	9 laranja
2 azul escuro	10 cinza
3 púrpura	11 rosa forte
4 verde escuro	12 verde
5 cinza	13 amarelo
6 azul mediano	14 aqua
7 azul claro	15 branco

OBS: Dependendo do computador as cores podem tornar-se alteradas.

Para uma cor específica use o comando COLOR, Exemplo:

```
10 GR
```

```
20 COLOR = 6 ← especifica a cor azul médio
```

```
30 PLOT 10,13
```

```
40 END
```

O terceiro passo em Gráficos, é ligar sua localização. Esta é a função da + instrução PLOT.

```
10 GR
```

```
20 FOR C = 1 TO 15
```

```
30 COLOR = C
```

```
40 PLOT 20,10 ← Liga a localização 20 horizontal e 10 vertical
```

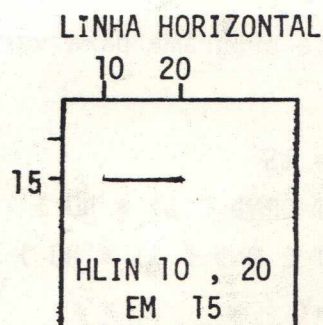
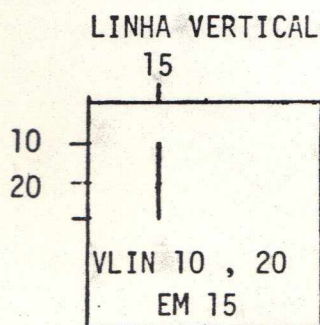
```
50 FOR X = 1 TO 500
```

```
60 NEXT X
```

```
70 NEXT C
```

O programa muda a cor do quadradinho na posição 20,10 cada vez que o valor de C for alterado.

Em adição, ligando pontos individuais uma linha pode ser desenhada na tela. Os comandos usados são VLIN para vertical e HLIN para horizontal.



Os dois primeiros números em cada comando mostram o começo e o fim da linha. O terceiro número mostra aonde a linha deve ser posicionada na horizontal-vertical. Os comandos Gráficos dividem a tela em duas partes. As 4 últimas linhas da tela são usadas para o texto. A porção acima é usada para os gráficos. Textos e gráficos não podem ser misturados na mesma linha. Para voltar ao modelo normal, você precisa de um comando TEXT seguido de um comando HOME .

← LOOP de tempo

PROBLEMAS PRÁTICOS 9,2

em que lugar estarão posicionados:

```
17 PRINT "VALORES COM UMA VIRGULA"
```

Digite este programa para ver o que ocorre:

```
20 X = INT {RND { 1} * 40 }
```

Mude linha 10 e tente prever o que ocorre:

```
10 FOR C = 1 TO 15
```

```
30 FOR X = 0 TO 39 STEP 3
```

(continua)

RESUMO

Para programar uma linha com várias Instruções, separe uma da outra com dois pontod, Exemplo:

```
10 PRINT A:GOTO 10
```

GOSUB

Quando houver um GOSUB numa linha de multi-instruções; o seu RETURN se dará na instrução seguinte aos dois pontos após o GOSUB.

IF ...THEN

Na linha de multi-instruções, se a condição não foi satisfeita; todo o resto é ignorado.

REM

Na linha de multi-instruções, tudo após o REM não é executado.

O ponto de interrogação (?) é a abreviação de PRINT quando se dá entrada a um programa.

.....

PROBLEMAS PRÁTICOS 10,1

Digite este programa como está, e depois veja como pode encurtá-lo usando linhas de multi-instruções:

```
1 HOME
5 PRINT "NUMEROS E QUADRADOS"
10 GOSUB 1000
20 READ Y
30 PRINT Y
50 REM CALCULO DE QUADRADOS
70 PRINT "AO QUADRADO E" : " " :
80 PRINT Y*Y
85 GOSUB 1000
90 IF Y      10 THEN GOTO 20
200 PRINT "TENHO DITO "
210 END
1000 PRINT "*****"
1020 RETURN
2000 DATA 1,2,3,4,5,6,7,8,9,10
```

.....

PROBLEMAS PRÁTICOS 10.2

- 43 -


```
10 PRINT "INTRODUZA UM NUMERO." : INPUT A
20 PRINT A
30 GOTO 10
```

AMOSTRA DE ENTRADA (INPUT):

```
1000000
1000000
233000000000000000
.01
.003
.1234567
.001234567
1.00000056
.0000000333333
```

Tente com outros números.

Após mude o programa, dê entrada a dois números e os multiplique. Veja como consegue notação científica para os números. Depois divida os dois números.

RUN TESTE 10

.....

QUESTÕES ESCRITAS DAS LIÇÕES DE 1 A 5

23. Digite um programa que imprime de 1 a 10 e que termina quando alcançar 10.
24. Digite uma instrução que produz um número randômico de 1 a 100.
25. Digite uma instrução que coloque valor de -7,3 numa caixa denominada A.

QUESTÕES ESCRITAS DAS LIÇÕES DE 6 A 10

11. Digite uma instrução que coloque o seguinte STRING na variável B\$:

VOCE ESTÁ AQUI

12. Digite um programa dando como entrada seu nome e que tenha a seguinte impressão:

? PAULO

ALÔ PAULO

13. Digite um programa usando um LOOP FOR...NEXT que conte regressivamente de 10 a 1 e imprime os resultados.
14. Digite um programa que conte de 1 a 5 usando a instrução READ DATA.
15. Digite um programa que designe o valor 1 a cada variável numa MATRIZ de A(1) até A(5).

RESPOSTAS DAS LIÇÕES DE 1 A 5

23.

```
10 REM PROGRAMA PARA CONTAR ATE 10 E PARAR
20 LET X = 0
30 LET X = X + 1
40 PRINT X
50 IF X = 10 THEN STOP
60 GOTO 30
```

24.

```
10 PRINT INT (RND (0) * 100) + 1
```

25.

```
10 LET A = -7,3 OU 10 A = -7,3
```

RESPOSTAS DAS LIÇÕES DE 6 A 10

11.

```
10 LET B$ = "VOCE ESTA AQUI"
10 B$ 1 "VOCE ESTA AQUI" OU
```

12.

```
10 INPUT N$
20 PRINT "ALO" ; N$ ; " I "
```

13.

```
10 FOR X = 10 TO 1 STEP -1
20 PRINT X
30 NEXT X
40 END
```

14.

```
10 DATA 1,2,3,4,5
20 READ C
30 PRINT C
40 IF C = 5 THEN STOP
50 GOTO 20
```

15.

```
10 FOR X = 1 TO 5
20 LET A(X) = 1
30 NEXT X
```


PALAVRAS RESERVADAS DA LINGUAGEM BASIC

A linguagem BASIC, de seu Microcomputador, como qualquer outra linguagem de programação tem uma série de palavras RESERVADAS, isto é, um conjunto de palavras que representam COMANDOS e FUNÇÕES específicas dentro da execução de um determinado programa.

Este apêndice do CURSO PASSO A PASSO, que ensina a linguagem BASIC, tem por objetivo fazer um RESUMO geral dos comandos que são mais utilizados nesta linguagem.

É claro que você já percebeu que os comandos da linguagem BASIC são nada mais que simples palavras em INGLÊS, que têm funções que se aproximam bastante de seu significado semântico.

Em seguida, você terá uma lista por ordem alfabética das palavras RESERVADAS mais usadas na linguagem BASIC. A cada uma delas procuramos associar palavras da nossa LÍNGUA que mais se aproximam do significado original.

E, finalmente, ainda acompanhando a tradução do comando, citaremos a sua função específica e quando é utilizado. Evidentemente as funções executadas estarão vinculadas aos parâmetros sobre os quais são utilizadas; por esta razão, torna-se imprescindível que você já tenha noções sobre estas funções de cada comando existente na linguagem BASIC. Você naturalmente adquire noções, após estudar as aulas iniciais de nosso curso PASSO A PASSO.

Procure manter vivo em sua memória toda a correspondência existente entre os COMANDOS e suas FUNÇÕES.

A linguagem BASIC caracteriza-se por ser de fácil aprendizagem justamente pela assimilação do significado dos comandos devido a representarem simples palavras do vocabulário INGLÊS.

COMANDOS DA LINGUAGEM BASIC E SEUS SIGNIFICADOS PRATICOS

COMANDO	TRADUÇÃO	FUNÇÃO
ABS	valor absoluto ou módulo	é uma função que devolve ao operador o valor absoluto de um número real.
AND	e, união	operação lógica "e"
ASC	ASC II	é uma função que devolve ao operador o número que representa o símbolo indicado dentro da TABELA ASC II de codificação de caracteres.
AT	em	É o comando utilizado em alta resolução de gráficos; indica até que ponto desejamos desenharmos uma linha.
ATN	arco-tangente	é a função que devolve ao operador o arco em radianos cuja tangente é o parâmetro citado na função.
CALL	chamar, executar	é o comando usado para iniciar a execução de uma rotina escrita em linguagem de máquina.
CHR\$	caracter	é a função que indica ao operador o caracter da TABELA ASC II correspondente ao número usado como parâmetro. É a função oposta à ASC.
CLEAR	limpa	este comando é usado para anular todas as variáveis da memória, sem que seja apagado o programa da memória.
COLOR=	cor	usado para definir qual cor será escolhida para desenhos de baixa resolução
CONT	continue	é o comando que reinicia a execução de um programa do ponto onde havia sido interrompido através de CTRL-C (break)
COS	cosseno	é a função que fornece o valor do cosseno de ângulo em radianos.
DATA	dados	utilizado para registro de dados que serão usados durante a execução de um programa.
DEF	definir	comando necessário para iniciar a definição de uma função matemática dentro da linguagem BASIC.
DEL	eliminar, apagar	serve para apagar linhas específicas de um determinado programa.

COMANDOS	TRADUÇÃO	FUNÇÃO
DIM	dimensão	é usado na definição da dimensão de variáveis endereçadas (vetores/array)
DRAW	desenho	comando usado para consumir o desenho de uma forma definida em assembler (linguagem de máquina) em gráficos de alta resolução.
END	fim	comando para encerrar execução de programas.
EXP	exponencial	função exponencial. Devolve ao operador o exponencial do parâmetro usado.
FLASH	"pisca-pisca"	define modo de impressão caracterizado pela alternância dos modos NORMAL e INVERSE.
FN	função	define ou faz menção a uma função matemática definida pelo programador.
FOR	para	início de rotina de LOOP. Usa-se em conjunto com comandos TO STEP NEXT
FRE	livre	devolve ao operador o número de bytes livres na memória quando referenciado com o parâmetro zero(0).
GET	obter	recebe do operador um número ou caractere introduzido através do teclado.
GOSUB	vá até sub-rotina	envia a execução de um programa para determinada sub-rotina. Usa-se com o RETURN.
GOTO	vá até	envia a execução para outra linha determinada pelo programador.
GR	gráficos de baixa resolução	serve para chamar resolução de gráficos.
HCOLOR	cor de alta resolução	define a cor usada em desenhos de alta resolução.
HGR	alta resolução (um)	chama a primeira página de alta resolução de gráficos.
HGR2	alta resolução (dois)	chama a segunda página de alta resolução de gráficos.
HIMEM	mais alta memória	define a mais alta alocação de memória
HLIN	linha horizontal	função dos gráficos de baixa resolução efetua o desenho de linha horizontal.
HOME	lar, casa	comando que limpa totalmente a tela.

COMANDO	TRADUÇÃO	FUNÇÃO
HPLOT	desenhos de pontos para alta resolução de gráficos	comando de alta resolução de gráficos usado para desenho de pontos e linhas retas
HTAB	tabulador horizontal	comando que indica a partir de que linha horizontal o computador deve começar a escrever.
IF	se(condicional)	comando de decisão lógica. Usa-se com THEN.
INPUT	entrada	serve para introduzir dados no programa via teclado do computador.
INVERSE	Inverso	faz com que a impressão seja em letras pretas sobre fundo branco.
LEFT\$	esquerda	especifica todos os caracteres à esquerda do parâmetro usado na variável alfanumérica (STRING).
LEN	comprimento	devolve ao operador o número de caracteres existentes na variável alfanumérica usada como parâmetro.
LET	permltir	faz com que determinada variável assumira um determinado valor.
LIST	listar	faz o computador mostrar a listagem das instruções que compõe um programa.
LOAD	carregue	usa-se para armazenar um programa da memória para uma fita magnética.
LOG	logarítmo	devolve ao operador o logarítmo natural do número usado como parâmetro.
LOMEM	mais baixa memória	usado para definir a mais baixa localização de memória.
MID\$	melo	especifica um pedaço de variável alfanumérica. Usa-se com 3 parâmetros: a variável, o número do caracter inicial e o número de caracteres desejados.
NEW	novo	elimina tudo que estiver na memória. Não resguarda dados nem programas.
NEXT	seguinte	define o final de um LOOP (Iniciado por instrução FOR)
NORMAL	normal	define o modo de impressão caracteriza do por caracteres brancos sobre fundo preto.
NOT	negação	operação lógica "NÃO"

COMANDO	TRADUÇÃO	FUNÇÃO
NOTRACE	não, rastrear	desativa o rastreamento (modo) de um programa.
ON	sobre	executa uma sub-rotina GOSUB ou um - JUMP- GOTO para uma determinada parte de um programa em função de determinada variável (seu valor).
ONERR	se ocorrer	indica ao computador, que caso ocorra algum erro durante a execução de um programa, o computador deve enviar a execução para outra parte do programa sem exibir mensagem de erro.
OR	ou	operação lógica: "OU"
PEEK	retirar, mostrar	devolve ao operador o valor correspondente ao local de memória usado como parâmetro.
PLOT	colocar um ponto	usado em baixa resolução de gráficos, para colocar um ponto.
POKE	colocar, por	coloca um determinado endereço de memória em um determinado byte.
POS	posição	devolve ao operador a coluna horizontal onde está posicionado o cursor de impressão.
PRINT	imprimir	imprime textos, números e valores de funções e operações matemáticas na tela, na impressora ou em fita de dados.
READ	ler	usados na leitura de dados registrados mediante o comando DATA.
RECALL	recobrar re-obter	carrega na memória do computador valores numéricos de variáveis armazenadas em fitas magnéticas.
REM	marca, registro documento	não possui função ativa em programação serve para fazer comentários dentro da listagem de determinado programa.
RESTORE	re-armazena	posiciona o apontador de leitura de dados do comando DATA/READ numa parte dos dados que já tenha sido lida.
RESUME	continuar, retornar à execução	reinicia a execução de um programa no ponto em que foi cancelado devido à mensagem de erro.
RETURN	retornar	comando necessário para representar o fim de uma sub-rotina. Quando encontrado, a execução retorna à expressão seguinte ao GOSUB correspondente.

COMANDO	TRADUÇÃO	FUNÇÃO
RIGHT\$	direita	inverso ao comando LEFT\$- Representa todos os caracteres à direita de um determinado caracter especificado, dentro de uma variável alfanumérica citada como parâmetro.
RND	randômico aleatório	função que usada com qualquer parâmetro, devolve ao operador um número aleatório entre 0 e 1
ROT=	rotação	faz com que uma forma escrita em alta resolução seja desenhada com rotação determinada. Quando se faz ROT=16 temos rotação de meio π radianos sobre a forma original do desenho.
RUN	correr	comando usado para dar início à execução de um programa da memória.
SAVE	salvar	armazena em fita cassete ou outros tipos de fita magnética o programa existente na memória do computador.
SCALE=	escala	faz com que uma forma escrita em alta resolução possa ser desenhada com uma ampliação determinada. Quando se faz SCALE=2 temos uma ampliação que corresponde ao desenho de uma forma igual ao dobro da original.
SCRN	cor no ponto da tela	usa-se em alta resolução para saber qual das cores de baixa resolução foram usadas para desenhar determinado ponto na tela.
SGN	sinal	função sinal. Devolve zero para valores nulos, -1 para valores negativos e, +1 para valores positivos.
SHLOAD	carregar forma de alta resolução	carrega para o computador uma forma de alta resolução, escrita em assembler, existente em registro em fita magnética
SIN	seno	é a função que fornece o valor do seno de um ângulo em radianos.
SPC(espaço	imprime tantos espaços em branco quantos forem indicados no parâmetro usado
SPEED=	velocidade	especifica para o computador qual deve ser a velocidade de impressão na tela do monitor. Pode-se usar qualquer número de 0 a 255. A velocidade máxima é 255 corresponde a 1000 caracteres/segundo.
SQR	raiz quadrada	função que fornece a raiz quadrada de um número real.

COMANDO	TRADUÇÃO	FUNÇÃO
STEP	passo	comando auxiliar em controle de LOOPS que define de quanto cada variável de controle será alterada a cada interação do LOOP
STOP	pare	cancela a execução de um programa e ao mesmo tempo indica a linha em que foi interrompida a execução.
STORE	armazenar	registra em fita magnética valores numéricos de variáveis armazenadas na memória do computador = RECALL.
STR\$	"string"	transforma uma variável numérica em uma cadeia de caracteres alfanuméricos corresponde ao comando VAL.
TAB(tabulador	deixa certa quantidade de espaços (margem) na linha antes de começar a imprimir Similar ao comando HTAB.
TAN	tangente	função que devolve o valor da tangente do ângulo em radianos.
TEXT	textos	volta ao modo de apresentação de textos após o uso de qualquer página de resolução gráfica.
THEN	então	define qual instrução(ou conjunto de instruções) o computador executa após considerar satisfeita a proposição introduzida do comando IF.
TO	até	usa-se com o comando FOR, define valores iniciais e finais de uma variável de controle de LOOPS.
TRACE	rastreamento	faz dentro da execução do programa, com que o computador imprima o número de cada linha que vai sendo executada. Auxilia o programador a localizar erros
USR	uso	comando usado na chamada de uma sub-rotina escrita em linguagem de máquina quando se deseja passar para esta rotina um determinado parâmetro.
VAL	valor	devolve ao operador o valor numérico de uma cadeia de caracteres alfanuméricos (string)
VLIN	linha vertical	função usada na resolução baixa de gráficos. Similar à HLIN, só que facilita o desenho vertical.
VTAB	tabulador	comando usado para especificar ao computador em que linha vertical ele deve começar a escrever texto ou mensagem.