

Indice analitico del volume

! 243 245 1001 1005 != 243 245 1001 1004 * 243 244 281 1001
 1003 1032 ** 299 300 1048 1050 *** 299 1048 *...const 289
 *= 243 244 1001 1003 *& 293 + 243 244 1001 1003 ++ 243 244
 1001 1003 += 243 244 1001 1003 . 314 315 .ascii 99 .bss
 100 .byte 99 .data 100 .equ 99 .int 99 .lcomm 99
 .odbc.ini 883 .text 100 / 243 244 1001 1003 /*...*/ 234
 990 // 234 990 /= 243 244 1001 1003 0... 238 996 01 642 0x...
 238 996 2421 58 5211 58 631-1 58 66 659 732-1 58 8421 58 88
 658 ; 234 990 = 243 244 1001 1003 == 243 245 1001 1004 ? :
 243 245 1001 1005 a.out 57 235 992 abort 573 abort() 474
 abs() 476 573 ACCEPT 675 access() 553 554 actual
 argument 277 ADC 81 105 ADD 81 97 102 677 addizione binaria
 964 AH 79 AL 79 alarm() 553 allineamento della memoria 183
 ALTER TABLE 794 ALTER USER 804 AND 73 977 and 485
 AND 82 and_eq 485 argc 298 1054 argomento attuale 277 argv
 298 1054 array 48 155 284 979 1037 array di puntatori 300 1050
 asctime() 518 579 assemblatore 54 88 assembler 88 assembler
 54 assembly 54 assembly 88 assert() 443 593 assert.h 443
 593 atexit() 474 573 atof() 469 573 atoi() 469 573
 atol() 469 573 atoll() 469 573 auto 272 AX 79 basi di dati
 771 BCD 58 BH 79 big endian 52 982 big endian 63 bit 236 993
 bitand 485 BL 79 BLANK WHEN ZERO 651 blkcnt_t 545
 blksize_t 545 BLOCK CONTAINS 640 bool 322 485 borrow
 66 69 971 BP 79 break 248 249 251 1011 1013 1015 BRKINT
 560 bsearch() 475 573 BSWAP 80 Bubblesort 18 372 BUFSIZ
 522 582 BX 79 byte 236 278 993 byte order 63 byte order 52 982 C
 205 233 989 1031 CALL 83 133 144 calloc() 304 473 573
 campo 319 carattere 236 278 993 carattere esteso 328 caratteristica
 62 caricamento di un programma 179 carry 66 69 71 71 71 73 79
 105 971 975 975 case 248 1011 cast 246 1007 CBW 80 cc 211
 cc_t 559 CDQ 80 CGI 847 CH 79 char 237 994 CHAR_BIT 443
 565 CHAR_MAX 443 565 CHAR_MIN 443 565 chdir() 553 556
 chmod() 548 595 chown() 553 CL 79 CLC 84 clearerr()
 540 582 clock() 515 579 CLOCKS_PER_SEC 515 clock_t
 515 545 CLOSE 678 804 close() 407 553 closedir() 434
 558 CMC 84 CMP 84 120 124 126 COBOL 718 718 CODE-SET
 641 codice di interruzione di riga 343 codice pesato 58
 collegamento 269 COMMIT 806 comparazione binaria 75
 compilazione di un programma 179 compl 485 complemento alla
 base 962 complemento a due 60 963 965 complemento a uno 59
 963 COMPUTE 678 condotto 428 CONFIGURATION SECTION
 624 confstr() 553 const 242 242 1000 const...* 289
 const volatile 242 continue 249 251 1013 1015
 convenzione di chiamata 144 conversione di tipo 246 1007
 conversion specifier 235 992 costante letterale composta 320 cpp
 212 257 creat() 407 551 CREATE DATABASE 805 CREATE
 TABLE 791 CREATE USER 804 CREATE VIEW 802
 ctermid() 540 ctime() 519 579 ctype.h 456 571 CWDE 80
 CX 79 data 787 DATA DIVISION 638 DATA RECORD 640 db 99
 DBA 774 DBMS 771 DCL 804 dd 99 DDD 96 207 DDL 774 785
 DEC 81 97 126 DECLARATIVES 669 DECLARE 803 default
 248 1011 DELETE 678 DELETE FROM 796 DEPENDING ON 653
 descrittore 406 Dev86 338 dev_t 545 DH 79 DI 79 difftime()
 517 579 digraph 240 280 999 DIR 434 434 558 directory 434
 directory 415 dirent.h 434 558 dislocamento 123 displacement
 123 DISPLAY 679 DIV 81 112 div() 476 573 DIVIDE 680
 divisione binaria 965 div_t 468 468 DL 79 DML 774 794 do 250
 1015 double 237 994 DROP TABLE 794 DROP USER 804
 DROP VIEW 802 DSN 883 dup() 553 dup2() 553 durata di
 memorizzazione 278 DX 79 EAX 79 EBX 79 138 EBX 79 eccesso 3
 58 ECHO 560 ECHOE 560 ECHOK 560 ECHONL 560 ECX 79 EDI
 79 EDOM 450 EDX 79 EFLAGS 79 EILSEQ 450 EIP 79 ELF 188
 else 248 1010 endianess 52 982 endianess 63 ENTER 141 144
 enum 312 enumerazione 312 environ 394 ENVIRONMENT
 DIVISION 623 EOF 343 522 582 equ 99 ERANGE 450 errno

66 971 PRId16 481 568 PRId32 481 568 PRId64 481 568
PRId8 481 568 PRIDFAST16 481 568 PRIDFAST32 481 568
PRIDFAST64 481 568 PRIDFAST8 481 568 PRIDLEAST16
481 568 PRIDLEAST32 481 568 PRIDLEAST64 481 568
PRIDLEAST8 481 568 PRIDMAX 481 568 PRIDPTR 481 568
PRIi16 481 568 PRIi32 481 568 PRIi64 481 568 PRIi8 481
568 PRIIFAST16 481 568 PRIIFAST32 481 568
PRIIFAST64 481 568 PRIIFAST8 481 568 PRIILEAST16
481 568 PRIILEAST32 481 568 PRIILEAST64 481 568
PRIILEAST8 481 568 PRIIMAX 481 568 PRIIPTR 481 568
printf() 235 311 357 531 587 992 PRIO16 481 568 PRIO32
481 568 PRIO64 481 568 PRIO8 481 568 PRIOFAST16 481 568
PRIOFAST32 481 568 PRIOFAST64 481 568 PRIOFAST8 481
568 PRIOLEAST16 481 568 PRIOLEAST32 481 568
PRIOLEAST64 481 568 PRIOLEAST8 481 568 PRIOMAX 481
568 PRIOPTR 481 568 PRIu16 481 568 PRIu32 481 568
PRIu64 481 568 PRIu8 481 568 PRIUFAST16 481 568
PRIUFAST32 481 568 PRIUFAST64 481 568 PRIUFAST8 481
568 PRIULEAST16 481 568 PRIULEAST32 481 568
PRIULEAST64 481 568 PRIULEAST8 481 568 PRIUMAX 481
568 PRIUPTR 481 568 PRIx16 481 568 PRIX16 481 568
PRIX32 481 568 PRIX32 481 568 PRIX64 481 568 PRIX64
481 568 PRIX8 481 568 PRIX8 481 568 PRIXFAST16 481 568
PRIXFAST16 481 568 PRIXFAST32 481 568 PRIXFAST32
481 568 PRIXFAST64 481 568 PRIXFAST64 481 568
PRIXFAST8 481 568 PRIXFAST8 481 568 PRIXLEAST16 481
568 PRIXLEAST16 481 568 PRIXLEAST32 481 568
PRIXLEAST32 481 568 PRIXLEAST64 481 568
PRIXLEAST64 481 568 PRIXLEAST8 481 568 PRIXLEAST8
481 568 PRIXMAX 481 568 PRIXMAX 481 568 PRIXPTR 481 568
PRIXPTR 481 568 PROCEDURE DIVISION 669 675 709
processo elaborativo in memoria 182 programma autonomo 195
programma *stand alone* 195 *promotion* 309 promozione 309
prototipo di funzione 252 1017 pseudocodifica 11 psql 819
pthread_t 396 545 PTRDIFF_MAX 450 566 PTRDIFF_MIN
450 566 ptrdiff_t 324 450 485 566 puntatore 281 291 1032
1044 puntatore a funzione 302 puntatore a puntatori 299 300 1048
1050 puntatore nullo 304 PUSH 83 135 PUSHA 83 141 144 PUSHF
83 putc() 535 putchar() 535 582 putchar_unlocked()
539 putchar_unlocked() 539 puts() 350 537 582 P_tmpdir
522 qsort() 475 573 QuickSort 21 374 raise() 512 rand()
472 573 RAND_MAX 469 rango 236 993 993 rank 236 993 993 RCL
82 118 RCR 82 118 RDBMS 774 READ 694 read() 409 553
readdir() 434 558 readlink() 553 realloc() 304 473
573 RECORD CONTAINS 641 REDEFINES 642 647
regcomp() 380 381 593 regerror() 380 386 593 regex.h
380 593 regexec() 380 383 384 593 regexp 380 regex_t 380
381 593 regfree() 380 383 593 register 272 registro 54 79
regmatch_t 380 383 384 593 593 regoff_t 593 relazione 774
RELEASE 714 remove() 428 524 582 rename() 524 582
RENAMES 659 reopen() 351 resb 99 resd 99 restrict
306 resw 99 RET 83 133 144 return 252 1018 RETURN 713
REVOKE 805 rewind() 538 582 rewinddir() 434 558
REWRITE 697 re_sub 593 ricerca binaria 17 riordino 24 26
riporto 65 66 71 71 71 73 105 969 971 975 975 rmdir() 428 553
rm_se 593 rm_so 593 ROL 82 117 ROLLEBACK 806 ROR 82 117
rotazione 73 976 rvalue 279 R_OK 553 SAL 82 116 SAR 82 116
SBB 81 108 scanf() 361 535 590 SCHAR_MAX 443 565
SCHAR_MIN 443 565 SCNd16 481 568 SCNd32 481 568
SCNd64 481 568 SCNd8 481 568 SCNdFAST16 481 568
SCNdFAST32 481 568 SCNdFAST64 481 568 SCNdFAST8 481
568 SCNdLEAST16 481 568 SCNdLEAST32 481 568
SCNdLEAST64 481 568 SCNdLEAST8 481 568 SCNdMAX 481
568 SCNdPTR 481 568 SCNi16 481 568 SCNi32 481 568
SCNi64 481 568 SCNi8 481 568 SCNiFAST16 481 568
SCNiFAST32 481 568 SCNiFAST64 481 568 SCNiFAST8 481
568 SCNiLEAST16 481 568 SCNiLEAST32 481 568

SCNiLEAST64 481 568 SCNiLEAST8 481 568 SCNiMAX 481
568 SCNiPTR 481 568 SCNo16 481 568 SCNo32 481 568
SCNo64 481 568 SCNo8 481 568 SCNoFAST16 481 568
SCNoFAST32 481 568 SCNoFAST64 481 568 SCNoFAST8 481
568 SCNoLEAST16 481 568 SCNoLEAST32 481 568
SCNoLEAST64 481 568 SCNoLEAST8 481 568 SCNoMAX 481
568 SCNoPTR 481 568 SCNu16 481 568 SCNu32 481 568
SCNu64 481 568 SCNu8 481 568 SCNuFAST16 481 568
SCNuFAST32 481 568 SCNuFAST64 481 568 SCNuFAST8 481
568 SCNuLEAST16 481 568 SCNuLEAST32 481 568
SCNuLEAST64 481 568 SCNuLEAST8 481 568 SCNuMAX 481
568 SCNuPTR 481 568 SCNx16 481 568 SCNx32 481 568
SCNx64 481 568 SCNx8 481 568 SCNxFAST16 481 568
SCNxFAST32 481 568 SCNxFAST64 481 568 SCNxFAST8 481
568 SCNxLEAST16 481 568 SCNxLEAST32 481 568
SCNxLEAST64 481 568 SCNxLEAST8 481 568 SCNxMAX 481
568 SCNxPTR 481 568 scorrimento 71 71 975 975 SD 639
SEARCH 698 SEEK_CUR 522 582 SEEK_END 522 582
SEEK_SET 522 582 *segmentation fault* 182 segno 64 968 SELECT
627 629 632 796 sequenza multibyte 477 SET 702 SETA 86
SETAE 86 SETB 86 SETBE 86 setbuf() 527 582 SETC 86
SETE 86 setegid() 553 seteuid() 553 SETG 86 SETGE 86
setgid() 553 SETL 86 SETLE 86 setlocale() 593 SETNA
86 SETNAE 86 SETNB 86 SETNBE 86 SETNC 86 SETNE 86
SETNG 86 SETNGE 86 SETNL 86 SETNLE 86 SETNO 86 SETNS
86 SETNZ 86 SETO 86 setpgid() 553 SETS 86 setsid()
553 setuid() 553 setvbuf() 527 582 SETZ 86 *shared object*
174 *shift* 71 71 113 975 975 SHL 82 114 130 short 237 994 SHR
82 114 130 SHRT_MAX 443 565 SHRT_MIN 443 565 SI 79
SIGABRT 509 510 SIGALRM 510 SIGBUS 510 SIGCHLD 510
SIGCONT 510 SIGFPE 509 510 SIGHUP 510 SIGILL 509 510
SIGINT 509 510 SIGKILL 510 *sign* 69 signal() 512
signal.h 508 signed 237 994 *significante* 62 SIGN IS 649
SIGPIPE 510 SIGPOLL 510 SIGPROF 510 SIGQUIT 510
SIGSEGV 509 510 SIGSTOP 510 SIGSYS 510 SIGTERM 509
510 SIGTRAP 510 SIGTTIN 510 SIGTTOU 510 SIGURG 510
SIGUSR1 510 SIGUSR2 510 SIGVTALRM 510 SIGXCPU 510
SIGXFSZ 510 SIG_ATOMIC_MAX 450 566 SIG_ATOMIC_MIN
450 566 sig_atomic_t 450 509 566 SIG_DFL 512 SIG_ERR
512 SIG_IGN 512 sistema binario 952 sistema decimale 952
sistema esadecimale 953 sistema ottale 953 sizeof 284
SIZE_MAX 450 566 size_t 323 450 485 545 566 sleep() 553
snprintf() 531 587 somma binaria 964 SORT 633 709
sottrazione binaria 964 SOURCE-COMPUTER 624 SP 79
SPECIAL-NAMES 624 *specificatore di conversione* 235 992
specifiche multiboot 196 speed_t 559 sprintf() 531 587
SQL 784 830 847 SQLite 875 srand() 472 573 sscanf() 553
590 SSIZE_MAX 445 ssize_t 545 *stack* 44 978 *stack frame* 141
stand alone 195 START 703 stat() 548 595 stat.h 406 546
595 static 270 272 stdarg.h 309 466 565 stdbool.h 485
stddef.h 485 593 stderr 351 523 STDERR_FILENO 553
stdint.h 446 566 STDIN_FILENO 553 stdio 351 523
stdio.h 341 521 582 587 590 stdlib.h 304 468 573 stdout
351 523 STDOUT_FILENO 553 STOP RUN 705 *storage duration*
278 strcasecmp() 549 strcat() 295 490 576 strchr()
295 496 576 strcmp() 295 493 576 strcoll() 295 493 576
strcpy() 295 488 576 strcspn() 295 498 576 strdup()
489 stream 341 strerror() 506 576 *strerror_r*() 507
strftime() 519 579 STRING 706 string.h 295 486 576
stringa 52 293 785 983 1045 stringa estesa 328 strings.h 548
strlen() 295 507 576 strncasecmp() 549 strncat()
295 491 576 strncmp() 295 493 576 strncpy() 295 488 576
strpbrk() 295 499 576 strrchr() 295 496 576 strspn()
295 497 576 strstr() 499 576 strtod() 469 573 strtok()
469 573 strtoumax() 484 strtok() 500 576 strtok_r()
503 strtol() 469 573 strtold() 469 573 strtoll() 469
573 strtouimax() 484 strtoul() 469 573 strtoull()

469 573 struct 314 structure stat 547 struct
 dirent 434 434 558 struct termios 559 struct tm 326
 516 struttura 314 strxfrm() 494 576 st_atime 547 595
 st_blksize 547 595 st_blocks 547 595 st_ctime 547 595
 st_dev 547 595 st_gid 547 595 st_ino 547 595 st_mode
 547 595 st_mtime 547 595 st_nlink 547 595 st_rdev 547
 595 st_size 547 595 st_uid 547 595 SUB 81 97 107
 SUBTRACT 707 super blocco 413 switch 248 1011 symlink()
 553 SYNCHRONIZED 650 sysconf() 553 system() 474
 S_IFBLK 546 595 S_IFCHR 546 595 S_IFDIR 546 595
 S_IFIFO 546 595 S_IFLNK 546 595 S_IFMT 546 595
 S_IFREG 546 595 S_IFSOCK 546 595 S_IRGRP 407 546 595
 S_IROTH 407 546 595 S_IRUSR 407 546 595 S_IRWXG 407 546
 595 S_IRWXO 407 546 595 S_IRWXU 407 546 595 S_ISBLK()
 546 595 S_ISCHR() 546 595 S_ISDIR() 546 595
 S_ISFIFO() 546 595 S_ISGID 407 546 595 S_ISLNK() 546
 595 S_ISREG() 546 595 S_ISSOCK() 546 595 S_ISUID 407
 546 595 S_ISVTX 407 546 595 S_IWGRP 407 546 595 S_IWOTH
 407 546 595 S_IWUSR 407 546 595 S_IXGRP 407 546 595
 S_IXOTH 407 546 595 S_IXUSR 407 546 595 tcflag_t 559
 tcgetattr() 561 tcgetpgrp() 553 TCSADRAIN 561
 TCSAFLUSH 561 TCSANOW 561 tcsetattr() 561
 tcsetpgrp() 553 tmpnam() 524 termios.h 559 TEST 84
 thread 395 time() 517 579 time.h 515 579 time_t 325 516
 516 545 TinyCOBOL 721 tmpfile() 524 582 tmpnam() 524
 582 TMP_MAX 522 582 toascii() 465 tolower() 464 571
 TOSTOP 560 toupper() 464 571 traboccamento 65 71 104 969
 translation unit 278 trigraph 240 280 999 true 485 ttyname()
 553 ttyname_r() 553 tupla 774 typedef 319 types.h 545
 U 238 996 UCHAR_MAX 443 565 uid_t 545 UINT16_C() 447
 566 UINT16_MAX 446 566 uint16_t 446 566 UINT32_C()
 447 566 UINT32_MAX 446 566 uint32_t 446 566
 UINT64_C() 447 566 UINT64_MAX 446 566 uint64_t 446
 566 UINT8_C() 447 566 UINT8_MAX 446 566 uint8_t 446
 566 UINTMAX_C() 449 566 UINTMAX_MAX 449 566
 uintmax_t 449 566 UINTPTR_MAX 449 566 uintptr_t 449
 566 UINT_FAST16_MAX 448 566 uint_fast16_t 448 566
 UINT_FAST32_MAX 448 566 uint_fast32_t 448 566
 UINT_FAST64_MAX 448 566 uint_fast64_t 448 566
 UINT_FAST8_MAX 448 566 uint_fast8_t 448 566
 UINT_LEAST16_MAX 447 566 uint_least16_t 447 566
 UINT_LEAST32_MAX 447 566 uint_least32_t 447 566
 UINT_LEAST64_MAX 447 566 uint_least64_t 447 566
 UINT_LEAST8_MAX 447 566 uint_least8_t 447 566
 UINT_MAX 443 565 UL 238 996 ULL 238 996 ULLONG_MAX 443
 565 ULONG_MAX 443 565 umask() 548 595 ungetc() 535 582
 Unicode 668 union 318 unione 318 unistd.h 406 552 unità di
 traduzione 257 278 unixODBC 883 unlink() 428 553 557
 Unproto 338 unsigned 237 994 UPDATE 795 USAGE 648
 USHRT_MAX 443 565 VALUE 651 VALUE OF 641 va_arg 309
 va_arg() 466 565 va_copy() 466 565 va_end 309
 va_end() 466 565 va_list 309 324 466 565 va_start 309
 va_start() 466 565 VEOF 559 VEOL 559 VERASE 559 vettore
 48 979 vfprintf() 531 587 vfscanf() 535 590 VINTR 559
 virgola mobile 61 967 VKILL 559 VMIN 559 void 243 252 322
 1001 1017 volatile 242 vprintf() 357 531 587 VQUIT 559
 vscanf() 361 535 590 vsnprintf() 531 587 vsprintf()
 531 587 vsscanf() 535 590 VSTART 559 VSTOP 559 VSUSP
 559 VTIME 559 wait() 391 WCHAR_MAX 450 566 WCHAR_MIN
 450 566 wchar_t 325 328 450 485 566 wcstoimax() 484
 wcstombs() 479 573 wcstouimax() 484 wctomb() 478
 573 WEOF 343 while 249 1013 WINT_MAX 450 566 WINT_MIN
 450 566 wint_t 325 450 566 word 78 WORD_BIT 445
 WORKING-STORAGE SECTION 644 WRITE 708 write() 409
 553 WWW-SQL 847 w_OK 553 x86 101 x86-32 77 XCHG 80 xor
 485 XOR 82 XOR 73 977 xor_eq 485 X_OK 553 zero 69 79 zero
 terminated string 52 983 zombie 393 # 234 990 #define 258

#define() 260 #define()...# 260 #define()...## 260
 #define()...__VA_ARGS__ 260 #define...## 258 #elif
 263 #else 263 #endif 263 #error 267 #if 263 #ifdef 264
 #ifndef 264 #if !defined 264 #if defined 264
 #include 258 #line 265 #pragma 269 #undef 265 & 243
 245 281 1001 1006 1032 &* 293 &= 243 245 1001 1006 && 243
 245 1001 1005 ^ 243 245 1001 1006 ^= 243 245 1001 1006 ~ 243
 245 1001 1006 ~= 243 245 1001 1006 \... 238 998 \0 238 998 \?
 238 998 \a 238 998 \b 238 998 \f 238 998 \n 238 998 \r 238
 998 \t 238 998 \v 238 998 \x... 238 998 \" 238 998 \\ 238 998
 \' 238 998 | 243 245 1001 1006 |= 243 245 1001 1006 || 243
 245 1001 1005 {...} 234 990 \$PGDATA 809 811 \$PGHOST 819
 \$PGPORT 819 \$PGTZ 841 _Bool 322 _Exit() 474 573
 _exit() 553 _IOFBF 522 582 _IOLBF 522 582 _IONBF 522
 582 _POSIX2... 445 _POSIX... 445 _Pragma 269
 _PROTOTYPE 335 _XOPEN... 445
 __bool_true_false_are_defined 485 __DATE__ 268
 __FILE__ 268 __func__ 312 __LINE__ 268
 __STDC_HOSTED__ 268 __STDC_IEC_559__ 268
 __STDC_IEC_COMPLEX__ 268 __STDC_ISO_10646__ 268
 __STDC_VERSION__ 268 __STDC__ 268 __TIME__ 268
 __udivdi3() 442 __umoddi3() 442 __VA_ARGS__ 260
 '...' 238 996 , 247 1008 - 243 244 1001 1003 -- 243 244 1001
 1003 -- 243 244 1001 1003 -> 315 < 243 245 1001 1004 <= 243
 245 1001 1004 << 243 245 1001 1006 <<= 243 245 1001 1006 >
 243 245 1001 1004 >= 243 245 1001 1004 >> 243 245 1001 1006
 >= 243 245 1001 1006 % 243 244 1001 1003 %+... 354 528 %...c
 354 359 528 %...d 354 359 528 %...e 354 359 528 %...f 354 359
 528 %...g 359 528 %...hd 354 359 528 %...hhd 359 528 %...hhi
 359 528 %...hhn 528 %...hho 359 528 %...hhu 359 528 %...hxx
 359 528 %...hi 354 359 528 %...hn 528 %...ho 354 359 528 %...hu
 354 359 528 %...hx 354 359 528 %...i 354 359 528 %...lc 354 359
 528 %...ld 354 359 528 %...Le 354 359 528 %...Lf 354 359 528
 %...Lg 359 528 %...li 359 528 %...lld 354 359 528 %...lli 359
 528 %...lln 528 %...llo 354 359 528 %...llu 354 359 528
 %...llx 354 359 528 %...ln 528 %...lo 354 359 528 %...ls 354
 359 528 %...lu 354 359 528 %...lx 354 359 528 %...n 528 %...o 354
 359 528 %...s 354 359 528 %...u 354 359 528 %...x 354 359 528
 %0... 354 528 %= 243 244 1001 1003 %-... 354 528

