
Index

Symboles

- [^ .] 47
- { } (accolades)
 - blocs d'instructions en JavaScript 147
 - délimiter les blocs de code en Ruby 52
 - enclore le corps de la fonction dans JavaScript 149
- [], opérateur en JavaScript 142
- () (parenthèses)
 - invoquer des fonctions 150
 - type en langage Haskell 405
- @ (arobase) dans les noms de variables Perl 205
- * (astérisque)
 - reconnaître une ou plusieurs instances dans les expressions régulières 47
 - reconnaître zéro ou plusieurs occurrences dans les expressions régulières 4
- | (barre verticale) 142
- ^ (circonflexe) 3
- \$ (dollar)
 - reconnaître la fin de ligne dans les expressions régulières 3, 5
- & (esperluète) 143
- > (flèche) 206
- . (point), opérateur en JavaScript 142

- % (pour cent) 205
- * sous-programme SGBSV (bibliothèque LAPACK) 269

Nombres

- 32 bits, mot de 153
 - adapter le poids de Hamming HAKMEM aux mots de 36 bits 157
- 8086, langage d'assemblage (Windows 1.0) 117

A

- abstraction
 - des entrées/sorties, indépendance, sur FreeBSD 304
 - syntaxique 427–449
- Acceptor, façade enveloppante 459
- accès aux données 240
- ACE, façades enveloppantes 458
 - façade enveloppante socket 464
- ACE_Handle_Set, variables de membre 464
- ACSS (Aural CSS) 540–545
 - ajouter des icônes auditory 541
 - associer (mapping) à des moteurs TTS 536
 - HTML élémentaire employant Emacs W3 et ACSS 545

- ACSS (*Aural CSS*) (*suite*)
 - produire des icônes auditory tout en énonçant les contenus 542
 - sortie vocale améliorée à l'aide d'une sémantique dépendant du contexte 542–545
 - structure de données encapsulant les réglages ACSS 540
- ActionFixture, classe (Java) 83
- actions (Haskell) 406
 - comme des valeurs de première classe 408
 - IO 415
 - description en compréhension de liste 419
 - et actions STM actions pour donner un type aux fonctions 418
 - situations pour lesquelles l'emploi est essentiel 418
 - type d'action STM (IO()) 421
 - main 407
 - système de type empêchant les actions IO de s'exécuter dans des actions STM 412
- actions STM 409
 - création d'une liste et combinaison à l'aide de `orElse` 421
 - donner aux fonctions le type STM à chaque fois que possible 418
- adaptabilité à la charge de travail
 - CIP 339
- Adapter ou Facade, motif 9
- addition
 - avec retenue, circuits d' 161–166
 - des facteurs, technique 40
- ADN
 - régulateur 198
 - séquences 198
 - site d'association de la protéine 217
- adressage
 - ouvert 313
 - par contenus, mémoire 51–54
- advFilter, méthode (champs Gene Sorter) 234
- advice (Emacs Lisp) 533
 - définitions attachées aux fonctions de navigation dans le calendrier 544
 - didacticiel 535
 - icônes auditory confirmation les actions des utilisateurs 541
- advised, fonction (Lisp) 535
- affectation, opérateurs (JavaScript) 143
- AFL (Audio Formatting Language) 539
- aire d'un triangle 576
- Ajax Generic Genome Browser 223
- alarm, fonction (Perl) 179
- aléatoires, utilitaire de générateur de nombres dans NumPy 331
- algèbre linéaire
 - algorithmes reformulés sous la forme d'opérations entre deux matrices 241
 - calcul scientifique sur ordinateur 239
 - logiciels pour les architectures avancées des ordinateurs 239
 - motivation pour des développements 240
- ALGOL, syntaxe de type, pour Lisp 136
- algorithmes
 - de recherche 57
 - description détaillée du sous-programme 271
 - LU récursifs 253
 - matriciels 240
 - architecture des ordinateurs, impacts sur 241
 - développement avec MATLAB 243
 - expression sous la forme d'opérations entre deux vecteurs 241
 - orientés par colonnes (LINPACK) 245
 - partitionnés par blocs 240
 - pour les matrices creuses ou denses 239
- algorithmique 568–578, 582
 - colinéarité 572–573
 - problème 571
 - tester par l'inégalité triangulaire 573
- Lisp 568
- analyse
 - de frameworks orientés objet, variabilités 456
 - d'erreur 262
 - syntaxique, techniques en LISP 136
- analyseurs syntaxiques (voir parseurs)
- annotation 198
 - grand nombre des types d'annotation du génome 200
- annulations (dans Subversion) 26
- anonymes, sous-programmes (Perl) 218
- applications basées sur des standards 338
- approche basée sur la décomposition pur résoudre des systèmes linéaires denses 242
- après advice (Emacs Lisp) 535
- arborescences de répertoires
 - contrôle de version et transformation d'arbre 14–15
 - différences 16
- arbre
 - binaire de recherche (Quicksort) 40
 - différences (éditeur de Subversion) 24

- repeuplement dynamique dans
 - eLocutor 519
- structure présentant des options dans
 - eLocutor 515
 - repeuplement dynamique 519
- transformation 12, 15
- arbre, transformation 12
- architecture
 - de services web 476
 - des ordinateurs 240
 - d'un système d'information
 - d'entreprise 337
 - multi-couches orientée-services 334
- ARN
 - messenger (ARNm) 228
 - séquences 198
- Art of Computer Programming, The 40
- assertions vérifiant l'état des verrous 302
- associatifs
 - à droite, opérateurs (JavaScript) 142
 - magasins 51–54
 - utilisation dans les langages typés dynamiquement 51
- associations 435
 - champs Gene Sorter 234
 - de la protéine, site, et ADN 217
 - des interactions 217
 - listes d' 435
 - procédure assq pour les consultations 439
 - (voir aussi bindings)
- AsTeR (Audio System For Technical Readings) 531
- asynchrone, démultiplexeur d'événements dans un serveur de journalisation réactif 464
- Atom, fils ou flux 551
- atomicité (transactions Haskell) 409
- audio
 - bureau d'Emacspeak 531–554
 - (voir aussi Emacspeak, bureau audio)
 - codes formatant l' 538
 - formatage à l'aide de voice-lock (Emacspeak) 536
- Audio Formatting Language (AFL) 539
- auditives, listes d'affichage
 - compléter Emacs pour créer 536–538
 - sortie formatée audio 538
- auditory, icônes 541
 - interactions de l'utilisateur produisant 541
 - produire tout en énonçant les contenus 543
- automates 50
 - finis 50

- autorités de certification (CA) 172
- autour d'advice (Emacs Lisp) 535
- autres niveaux d'indirection 294–304
 - multiplexage et démultiplexage 304
 - système d'exploitation prenant en charge différents
 - implémentation FreeBSD de l'appel système read 294
 - utilisation par FreeBSD de l'indirection pour abstraire les arguments de la fonction read 295–300
- avant advice (Emacs Lisp) 535
- awk, langage de programmation 54
 - créations à partir de la spécification de l'appel système read de FreeBSD 302

B

- backtick, fonction (Cryptonite) 191
- Backus-Naur Form (BNF), grammaire 65–67
- barre de progression pour les programmes volumineux 56
- base, catégories
 - ERP5
 - prédécesseur 365
 - source_project 364
 - Portal Categories, objets 362
- base de données
 - code 181
 - schémas 202
- bases de données
 - dans le noyau d'un système d'exploitation pour dénommer les périphériques 282
 - relationnelles
 - ERP5 356
 - (voir aussi relationnelles, bases de données)
- bases de l'ADN (A, T, C, G) 227
- baton (Subversion) 17
- beans de sessions
 - avec état 338
 - sans état 338
- beau code 239–263
 - architecture du système CIP 336–340
 - carence en vérification de type dans le noyau Linux 287
 - code qui fonctionne efficacement et avec précision 265
 - décomposition LU récursive 251–254
 - impacts de l'architecture des ordinateurs sur les algorithmes matriciels 240
 - itérateurs NumPy 320
 - modularité 272

- beau code (suite)*
 - multithreading pour les systèmes multicœurs 259–262
 - orientations futures pour la recherche 263
 - programmes
 - d'exemple et de test 273
 - parallèles 401
 - sous-programmes
 - DGEFA de LINPACK 245–248
 - DGETRF de LAPACK 248–250, 251
 - PDGETRF de ScaLAPACK 254–259
- Bell 33
- bénie (blessed), référence 182
- Bentley, John 93
- Bentley, Jon 31–43
- Berkeley Fast Filesystem (ffs) 296
- bibliothèque de fonctions Java IBM pour intégrer les systèmes RPG dans les logiciels client/serveur 485
- bibliothèque du CERN 266
 - beauté extérieure d'un code 267–273
 - beauté intérieure d'un code 274–280
 - la beauté est dans le flot 279
- bien formé (XML) 64
- binaire
 - arbre de recherche (Quicksort) 40
 - composition de message MIME 188
 - recherche 57, 93
 - (voir aussi dichotomique, recherche)
- binaire, recherche (voir dichotomique, recherche)
- binarité 108
- bindings 435
 - associer (mapping) à des étiquettes à l'aide d'environnements 438
 - des interactions 217
 - déterminer le binding associé grâce à un identifiant 438
 - variable 428–430
- Bio::Graphics, module (Perl) 198–225
 - étendre 221–226
 - prendre en charge des images de qualité professionnelle 223
 - exemples de sorties 199
 - image représentative 199
 - pré-requis de la conception 200
 - processus de conception 202–220, 221
 - classes d'objets 208–210
 - développer les interactions avec le module 202, 203
 - options dynamiques 217, 220
 - réglage des options 206, 207
 - script utilisant (exemple) 216, 217
 - traitement des options 211–215
 - soutenir les développeurs du Web 221
- Bio::Graphics::Glyph, classe 208–210, 211
 - méthodes factory() et option() 215
 - option(), méthode 220
 - sous-classes 215
- Bio::Graphics::Glyph::Factory, classe 212–216
 - méthodes 212
 - traitement dynamique des options 218
- Bio::Graphics::Panel, classe 202–208
 - image_and_map(), méthode 222
 - images SVG 223
 - options de configuration 207
 - png(), méthode 223
- Bio::Graphics::Track, classe 208
- bioinformatique 197
 - architectures de service web de type SOAP 477
- BioPerl, bases de données d'annotations 203
- Bio::SeqFeature::CollectionI, interface 203
- Bio::SeqFeature::Generic, classe 216
- Bio::SeqFeatureI, interface 226
- bison 303
- BitBlt (bit block transfer), fonction (Windows 1.0) 112
 - 8086, instructions de code machine vues comme des sous-programmes 117
 - implémentation du code C 116
 - raster, opérations 114
- bitmaps
 - code s'occupant de 116
 - filtre numérique
 - appliqué à 118
 - transformant la source en le bitmap de destination 119
 - ImageFilterTest, programme 120
- bits
 - décalage pour les collisions dans les dictionnaires Python 313
 - drapeaux (table d'examen binaire) 75
 - opérateurs de décalage (Java) 94
- BLACS (Basic Linear Algebra Communication Subprograms) 254
- Blandy, Jim 16, 17
- BLAS (Basic Linear Algebra Subprograms) 240, 247
 - bibliothèque ScaLAPACK 254
 - niveau 1 245
 - utilisation dans la factorisation LINPACK 247
 - niveau 2 240, 254
 - niveau 3 240, 254
- BLAS Parallèle (PBLAS) 257, 259
- block, instruction (JavaScript) 148
- blocs de code (JavaScript) 148
- bloquer
 - chaîne de blocage 372
 - itérer par dessus de manière cohérente 375

- blur (flou), filtre 119
- BNF (Backus-Naur Form), grammaire 65, 67
- bogue d'héritage de priorité au niveau utilisateur 370–384
- Bonwick, Jeff 371
- bord, tests de valeur au 97
 - dichotomique, recherche 98–102
 - tableau 98–101
- bound-identifier=?, prédicat 445
- bouton droit de la souris, vu comme une unique saisie binaire pour eLocutor 515
- branchement, instruction (Intermediate Language) 127
 - Blt (branch if less than) et S (short branch) 129
 - étiquettes (labels) 127
- Bray, Tim 45–60
- break, instruction (JavaScript) 148
- brièveté dans le code 274, 506
 - élimination de la redondance 506
 - oublier les informations non nécessaires 506
- broadcasting (en NumPy) 329, 331
- brosse ou motif (objet graphique) 114
- brute
 - opérations sur image 113–118
 - (voir aussi raster, opérations)
 - ternaire, opération 114
- BT (business template) 362
- bufferisation (buffering) 305
- __builtin__, module (Python) 308
- bureau audio d'Emacspeak 531–554
 - (voir aussi Emacspeak, bureau audio)
- business
 - processus 357
 - template (BT) 362
- C**
- C# 118
 - algorithme de filtre numérique 123
 - code
 - écrit pour optimiser les performances 120
 - implémentant les filtres numériques d'images 118
 - indirection dans 305
 - Intermediate Language 125
- C++
 - cfront, pré-processeur 303
 - héritage et surcharge 274
 - implémentation MapReduce du programme de comptage de mots 399
 - types paramétrés 454
- c (caractère) 3
- C elegans, génome 199
- C, langage
 - code
 - d'application d'un filtre numérique 119
 - implémentant les couches du système de fichiers 303
 - rassemblant les arguments d'une fonction en une seule structure 302
 - compilation depuis un langage spécialisé, de haut niveau, et propre au domaine 303
 - constructeurs syntaxiques visant un but particulier 427
 - interfaces
 - de programmation du système d'exploitation (OS) 451
 - Perl vers les bibliothèques 176
 - mécanisme d'abstraction syntaxique 428
 - N dimensions, tableaux à 318
 - objet polymorphe 231
 - pointeurs 285
 - premières applications Windows 117
 - programmation orientée objet et 8
 - structures
 - comptage de références 288–289
 - héritage et manipulation sous le noyau Linux 287
- CA (Certification Authorities) 172
- cache
 - allocateur d'objet 373
 - placer en mémoire dans eLocutor 513–523
 - SaveReverse, sous-programme 523
- caché, répertoire pour les messages de boîte de réception 186
- cachées, variables CGI 230
 - clé dans la table de session pour Gene Sorter 230
- cadre opérationnel 239
- Caenorhabditis elegans, génome 199
- calculs
 - anticipés 259
 - du milieu 99
 - (voir aussi milieu, calcul)
 - matriciels par décomposition 242
- Calendar (Emacs) 543
- calendar-forward-week, fonction 544
- callbacks dans Bio::Graphics
 - utilisation pour chaque option passée à add_track() 218
 - utilité 220
- Cantrill, Bryan 369–386

- capture
 - d'une référence de variable par un binding de variable 428–431
 - problèmes variables 428–431 (voir aussi variables, problèmes de capture)
- caractères
 - classes de (dans les expressions régulières) 8
 - de ponctuation 538
 - reconnaître un seul dans les expressions régulières 3
- carry-save adder (CSA), circuits 161–166
- cart
 - objet (Gene Sorter) 230
 - variables (Gene Sorter)
 - communication entre les méthodes de filtre des champs 234
 - éviter les conflits de noms 230
- cartes d'images, pour la sortie de Bio::Graphics 221
- casser le code (tests esthétiques) 101
- catégories de base
 - ERP5 (voir base, catégories (ERP5))
 - objets (Portal Categories) 362
- causalités 363
- causes les plus probables d'abord 493
- CD-ROM 295
- centré sur le document, paradigme (ERP5) 355
- certification, autorités (CA) 172
- CGI, scripts
 - à vie 229
 - avantages et inconvénients 229
 - courte durée de vie 231
 - Gene Sorter 229
 - stockage de données à long terme 229
- CGI, variables cachées 230 (voir aussi cachées, variables CGI)
- chaînage 313
 - machines vectorielles 241
- chaînes
 - de threads bloqués 372
 - type de dictionnaire spécialisé dans les 311
- chemins
 - critiques 259
 - ERP5 356
- chiffrement
 - client de courrier électronique 185
 - du courrier électronique 173
 - ou signature des messages 184
- cible, emplacement de l'objet 100
- CIP (Collaborative Information Portal) 60–61, 253, 271–279, 282, 333–353
- CISC, ordinateurs 153
- classer les résultats de recherche 61
- classes
 - bibliothèques de 455
 - d'objets (Bio::Graphics) 202 (voir aussi objet, classes (Bio::Graphics))
 - HotkeyAdaptorFactory (exemple) 480
 - Service (exemple) 478
- clés
 - authentification par
 - publique, cryptographie à clé 172
 - collisions dans les dictionnaires Python
 - consultation 311
 - non ordonnées 308
 - de session 230
 - gestion des 186
 - privées 175
 - publiques, cryptographie 172 (voir aussi publique, cryptographie à clé)
 - publiques liées aux contacts dans le carnet d'adresses de l'utilisateur (Cryptonite) 175
 - USB 293
 - utilisateur pour Gene Sorter 230
- clé/valeur, paires
 - dictionnaires Python 307
- clients externes 476
- client/serveur, logiciel
 - serveurs de synthèse vocale et client Emacspeak 532
- clipboard (presse-papiers) 527
- CMF (Content Management Framework) 355
 - implémentation de la portion structurelle des applications dotées de types CMF 358
 - types 358
- codage en dur de paramètres 353
- codant les protéines, gènes 198
- code
 - complexité 552
 - compréhensible 235–238
 - brèveté et simplicité d'un code 274 (voir aussi compréhensible, code)
 - correcteur d'erreurs 165
 - et données 111
 - génération 111–132
 - FilterMethodCS 122, 125
 - FilterMethodIL 125, 133
 - fonctions graphiques dans Microsoft Windows 1.0 112–118
 - géré 118
 - points Unicode en XML 64
 - programmes générant lors de leur exécution 112
- colinéarité naïve, fonction 570

- collisions
 - contrôle des
 - pour chevaucher les fonctionnalités du génome 200
 - pour les pistes dans le module Bio::Graphics 205
 - dans les dictionnaires Python 313
- columnDb.ra, fichiers 232
- columns (Gene Sorter) 233
 - filtrage 234
- commentaires
 - fourniture de l'intégralité de la documentation dans le code 271
 - Haskell 405
- Common LISP, macros 427
- communes ou variables, analyse des parties 456
- communications sécurisées 169–196
 - (voir aussi sécurisées, communications)
- comparaisons 108
 - fonction comptant les (Quicksort) 35
 - opérateurs de 312
- compilation, phase 7
- complet, mot 158
- complétion des mots dans eLocutor 513, 521
- complexité du code 552
- complications accidentelles rencontrées par les développeurs 451
- complications inattendues rencontrées par les développeurs 451
- compréhensible, code 235–238
 - documentation 237
 - limites de la mémoire humaine 235
 - minimiser les effets de bord 238
 - noms bien choisis 236
 - utiliser une portée aussi locale que possible 238
- comptage de références d'objets
 - dans la couche virtuelle du filesystem du noyau Linux 290, 291
 - dans la couche virtuelle du système de fichiers 287
 - dans le système de fichiers virtuel du noyau Linux 288
- comptage du nombre d'opérations 262
- comptes bancaires (exemple de programmation concurrentielle) 403–405
 - emploi de verrous 402
- conception, défauts se manifestant sous forme de bogues 370
- concurrency 401–424
 - comptes bancaires (exemple) 402–405
 - différents modèles 453
 - séquentielle, modèles 462–466
 - verrous 403
- conditions de terminaison (moteur d'expressions régulières) 6
- confiance, réseau 172
 - (voir aussi réseau de confiance)
- connexion/données, gestionnaire d'événements 464
- considérations ergonomiques dans les cryptosystèmes à clé publique 174
- constantes hexadécimales en C 156
- constructeur (méthodes) 71
- consultation, table de 73
 - appels de méthodes d'instance Java distribués à travers 305
 - format binaire 75
 - chargement 76
- consultation, type (champs Gene Sorter) 233
- contexte
 - choix de menu dépendant du, dans eLocutor 517–519
 - sémantique dépendant du 542–544
- contigus, tableaux non 319
- continus, tests 493
- contrôler 54
- conversions entre s-expressions et objets de syntaxe 445
- cookies
 - clé dans la table utilisateur pour Gene Sorter 230
 - script CGI pour stocker les données 229
- copie de travail (Subversion) 15
- core dump 377–383
- correcteurs d'erreurs, codes 165
- couches
 - des systèmes de fichiers 300–302
 - sur FreeBSD 295
 - et indirection 305
- couplage faible (SOA) 339
- courrier électronique
 - chiffré, client 185
 - client de 176
 - déchiffré 185
- courrier électronique, système (Cryptonite) 170–194
 - magasin de courriers électroniques (mail store) 181
 - réplication 184, 193
- court-circuiter les opérateurs logiques (JavaScript) 142
- CPython 308
 - choisir la fonction de stockage de manière dynamique 312
- Crockford, Douglas 135–152
- Crypt::GPG, module (Perl) 182
 - audit de sécurité 189–192
- cryptographie à clé 172
 - (voir aussi publique, cryptographie à clé)

- cryptographique, logiciel 195
 - Cryptonite (système de courrier électronique) 170–194
 - conception élémentaire 177
 - le processus de développement vu de l'intérieur 170
 - magasin de courriers électroniques (mail store) 192
 - messages déchiffrés 185
 - nettoyage du code 181
 - objectifs et décisions de la conception 177
 - prototype fonctionnel 181
 - rénover le magasin de courrier 183
 - sécuriser le code 187
 - suite de tests 180
 - Cryptonite::Mail::Config, module (Perl) 188
 - Cryptonite::Mail::HTML, paquet 182
 - Cryptonite::Mail::Server 178
 - Cryptonite::Mail::Service, classe 178
 - Crypt::PGP5, module (Perl) 182
 - CSA (carry-save adder) circuits 161–166
 - CSS2 540
 - première implémentée dans le navigateur Emacs W3 545
 - cuillerée d'eaux usées dans la barrique de vin 369
 - Cunningham, Ward 81
 - Cypherpunks 195
- D**
- \d 47
 - data display debugger (ddd) 490
 - datum->syntax, procédure (syntax-case) 432–433
 - conversions d'une s-expression en objet de syntaxe 445
 - DBD::Recall, module (Perl) 184
 - DBD::SQLite, module (Perl) 181
 - DCWorkflow 356
 - ddchange, greffon (plug-in) pour Eclipse 502
 - ddd (data display debugger) 490
 - de Carvalho, Rogerio Atem 355–366
 - deadlock (voir interblocage)
 - Dean 398
 - Dean, Jeffrey 387–400
 - débogage
 - posthume 377
 - post-mortem 377
 - sur site 377
 - systématique 492
 - déboguer 489–503
 - chasser le défaut 499–502
 - remonter des variables provoquant les erreurs aux instructions qui les ont mises en place 501
 - trouver les causes d'échec dans l'état du programme 499
 - delta, débogage 496–499
 - trouver automatiquement la cause de l'échec 494
 - début, position (fonctionnalités du génome) 200
 - décalage à droite, instruction 157
 - déchiffrés, messages de courrier électronique 185
 - déclarations 506
 - décomposition
 - fonctionnelle du logiciel non élémentaire 452
 - LU 240
 - sous-programme SGBTRF de LAPACK 275–279
 - décompte des liens (PageRank) 61
 - décompte des mots, programme (exemple)
 - programme de décompte des mots parallélisé utilisant des processeurs partitionnés
 - groupements de mots dans eLocutor 514
 - découper en tranches 318
 - DECTalk Express 532
 - défauts de conception se manifestant sous forme de bogues 370
 - define-syntax
 - forme 431
 - template spécifiant la sortie dans une forme 432
 - définir
 - nom, trouver (analyseur syntaxique JavaScript) 145
 - nouvelles variables dans un périmètre (analyseur syntaxique JavaScript) 144
 - Delivery, classe 363
 - DeliveryLine, classe 360
 - delta, débogage 496–498, 499
 - rechercher les causes d'échec dans le programme internes 498
 - sur les états du programme 500
 - delta, éditeur (Subversion) 11–29
 - avantages de 28
 - interface 17–21
 - démultiplexer les événements synchrones, interfaces de programmation (systèmes d'exploitation) 463
 - démultiplexeur asynchrone d'événements dans un serveur de journalisation réactif 464
 - dépendant du contexte, sémantique 542–544

- déroulante, somme de contrôle, algorithme
 - rsync 11
- déroulement dans eLocutor 526
- descripteurs de tableau 257
- description en compréhension de liste (Haskell) 419
- destination_project, catégorie de base 364
- deux dimensions, tableaux 317
- développement
 - hygiénique de macros (voir hygiénique, développement de macros)
 - non hygiénique de macros 430
- développement, algorithmes
 - algorithme de développement de macro hygiénique (KFFD) 430
 - syntax-case 433–446
 - commencer le développement 445
 - comparaison des identifiants 444
 - créer des enveloppes 437
 - expand 440–441
 - manipuler les environnements 438
 - prédicats structurels 437
 - produire une sortie développée 435
 - représentations 434
 - résolution des identificateurs 438
 - syntaxe, erreurs de 437
 - transformateurs de code 442, 443
- devfs, système de fichiers 282
- dichotomique, recherche 57, 93
 - avantages et inconvénients 59
 - messages 60
 - On the Goodness of Binary Search 57
 - sortir de la boucle 60
 - tester 97–110
 - performances 108
 - test unitaire 97
- dictionnaires 307–314
 - cas particulier d'optimisation pour les petits hachages 311
 - changer la taille 313
 - déterminer la taille de la nouvelle table 314
 - clés
 - et valeurs de divers types de données dans un seul dictionnaire 308
 - non ordonnées 308
 - collisions 313
 - implémentation C de Python 308
 - itérations et modifications dynamiques 315
 - opérations élémentaires sur 307
 - particulariser un cas 311
 - passer des mots clés comme arguments d'une fonction 308
 - représenter les contenus du module 308
- diff, outil 491
- différences
 - d'arbres (éditeur de Subversion) 24
 - de texte (Subversion) 17
- diffusion (en NumPy) 329, 331
- dimensions
 - N 317
 - tableaux à deux 317
 - tableaux à N (voir N dimensions, tableaux à)
 - tableaux à trois 317
 - tableaux à une 317
 - et algorithme de filtre numérique en C# 122
- direction (fonctionnalités du génome) 200
- dispatcher, état des threads dans Solaris 374
- distance
 - de Hamming entre deux vecteurs de bits 165
 - procédure (Lisp) 573
- diviser le texte en clauses appropriées (Emacspeak) 538
- diviser pour régner, stratégie
 - algorithme récursif de décomposition LU 251
 - Hamming, poids 155, 158
 - subdivision logique des tâches en sous-programmes 279
- document, centré sur, paradigme (ERP5) 355
- documentation 237
- Dongarra, Jack 239–248
- données
 - et code 111
 - représentation 305
 - structures vues comme des algorithmes gelés 40
 - types
 - paramétrés en C++ 454
 - sauter les déclarations non nécessaires 506
- données/connexion, gestionnaire d'événements 464
- Don't Repeat Yourself (principe DRY) 507
- doter Emacs de la parole 533–545 (voir aussi parole, doter Emacs de la)
- double précision, arithmétique IEEE à virgule flottante 574
- do-while, boucle 5
- droit, bouton de souris, vu comme une unique saisie binaire pour eLocutor 515
- droite, opérateurs associatifs à (JavaScript) 142
- DRY, principe (Don't Repeat Yourself) 507
- Duff, le périphérique de (Duff's Device) 11
- duplication d'informations dans le code 506
- Dybvig, R. Kent 427–449

- DynamicMethod, classe 125
 - Invoke, méthode 133
- dynamique
 - dispatch 304
 - programmation 36
 - repeuplement d'arbre dans eLocutor 519
- dynamiquement typés, langages 51
- E**
- eaux usées, dans la barrique de vin 369
- échanges HTTP 482
- éditer le texte dans eLocutor 525
- éditeur contrôlé avec un seul bouton 512, 525–529
- effets de bord en Haskell 406–408
 - caractère explicite 408
- egrep 2
- éléments, noms des (XML 1.0) 65
- élimination
 - des enregistrements pathogènes (MapReduce) 397
 - Gaussienne 239–247
 - outil MATLAB 243
- eLocutor 511–529
 - efficacité de l'interface utilisateur 528
 - fonction search 527
 - interface d'entrée
 - chercher 527
 - clic long 517–519
 - clipboard (presse-papiers) 527
 - macros 527
 - marché de niche pour une communauté plus large 512
 - modèle de conception élémentaire 512–515
 - navigation 525
 - sites web pour téléchargement et liste de discussion 529
 - typer 521
- Emacs
 - doter de la parole 533–545
 - (voir aussi parole, doter Emacs de la)
 - Emacspeak 529
 - environnement de développement intégré pour un système de courrier électronique sécurisé 176
 - W3, navigateur web 545
- Emacspeak, bureau audio 531–554
 - accès aux informations en ligne 544–552
 - Aural CSS (ACSS) 540
 - gérer la complexité du code au fil du temps 552
 - ligne de commande Web et templates d'URL 549
 - outils orientés web 545–552
 - (voir aussi web, outils orientés, dans Emacspeak)
 - parole, doter Emacs de la
 - premier jet d'implémentation 534
 - produire une sortie parlée 532
 - témoignages issus de l'implémentation et de l'utilisation 553
- emacspeak-auditory-icon, propriété 542
- emacspeak-calendar, module 543
- emacspeak-calendar-entry-marked-p, fonction 543
- emacspeak-calendar-speak-date, fonction 543
- emacspeak-minibuffer-setup-hook, fonction 541
- emacspeak-personality-voiceify-faces, variable 537
- emacspeak-speak-line, fonction 533
- emacspeak-url-template, module 549
- emacspeak-url-template-fetch, commande 550
- emacspeak-w3, module 545
- emacspeak-w3-extract-table-by-match, fonction 548
- emacspeak-websearch, outil pour accéder aux indications de Yahoo! Maps 547
- emacspeak-websearch-yahoo-map-directionsget-locations, fonction 547
- e-mail, client de 176
- emplacement de l'objet cible 100
- enregistrements (pathogènes) 397
- ensembles 165
- Enterprise JavaBeans (EJB) 338
- entrées, validation des (système de courrier électronique Cryptonite) 188
- entrées/sorties
 - indépendance de l'abstraction, sur FreeBSD 304
 - vecteur de requêtes 304
- entreprise, systèmes de planification des ressources 355
 - objectifs 356
- enveloppantes, façades 466
- enveloppes, créer 437
- environnements (algorithme de développement syntax-case) 438
 - environnement meta et environnement lors de l'exécution 440
- Equalizer 511
- équations mathématiques et solutions calculées 265
- équilibre de la charge 395
- ergonomie, considérations dans les cryptosystèmes à clé publique 174
- ERP (enterprise resource planning) 356

- ERP5 355–368
 - composants Zope utilisés par 355
 - concepts jetant les bases de la
 - représentation des processus industriels 357
 - fonctionnalités améliorant la productivité du développement 361
 - Project 362–368
 - coder 363
 - schéma de mapping object-to-relational 356
 - service de portail Portal Categories 362
 - Zope, plate-forme
 - structure à quatre niveaux représentant les classes système 360
 - erreurs
 - Euclide
 - algorithme calculant le PGCD de deux nombres 568
 - formule donnant l'aire d'un triangle 577
 - mécanisme de communication dans le sous-programme SGBSV de LAPACK 272
 - problèmes de récupération avec les verrous 404
 - tentative 505
 - espace de nom des URI 78
 - esthétique, code (voir beau, code)
 - étiquettes (labels) 435
 - en langage intermédiaire (Intermediate Language) 127
 - évaluation virtuelle, pile (en IL) 126
 - événements
 - file 533
 - gestionnaire de données ou connexion 464
 - synchrones, interfaces de programmation démultiplexant (systèmes d'exploitation) 463
 - exécution
 - des transactions 411
 - optimiste 411
 - phase 7
 - exp, procédure 440
 - exp-core, procédure 441
 - Expect, module (Perl) 189
 - exp-exprs, procédure 441
 - exp-if, procédure 442
 - exp-lambda, procédure 442
 - exp-let, procédure 443
 - exp-letrec-syntax, procédure 443
 - explicite, formulation des hypothèses (lors du débogage) 493
 - exploitation, systèmes
 - Unix 295
 - (voir aussi Unix, système d'exploitation)
 - (voir aussi systèmes d'exploitation)
 - exp-macro, procédure 440
 - exp-quote, procédure (syntax-case) 442
 - expression, fonction 140
 - expressions
 - comparaison avec les instructions 147
 - macros Lisp reposant sur des 428
 - régulières 1–8, 47–50
 - moteur (voir aussi régulières, moteur d'expressions)
 - (voir aussi régulières, expressions)
 - extend-wrap, assistant 444
 - extensibilité du matériel 89
 - extension, points d' 542
- F**
- façades enveloppantes 466
 - facteurs, technique d'addition 40
 - factorisation LU
 - implémentation simple 244
 - multithread 259–262
 - récursive 251–254
 - solutions
 - LAPACK 248
 - ScaLAPACK 254–257
 - Factory, classe (Bio::Graphics::Glyph) 212–215
 - make_glyph(), méthode 213
 - option(), méthode 213
 - familiarité (du beau code) 507
 - FAT-32, système de fichiers pour clé USB 293
 - favoris dans eLocutor (mots fréquemment employés) 524
 - Feathers, Michael 81–90
 - fenêtres, gestionnaire (Subversion) 17
 - feuilles de propriétés 359
 - ffs (Berkeley Fast Filesystem) 296
 - fgrep 2
 - fiabilité du CIP 339
 - service de flux 350
 - fichier
 - middleware.properties 351
 - web.xml du servlet fournisseur de l'interface utilisateur 478
 - fichiers, systèmes de (voir systèmes de fichiers)
 - fil RSS et Atom 551
 - Filter, classe 122
 - filterControls, méthode (champs Gene Sorter) 234
 - FilterMethodCS 122
 - FilterMethodIL 125, 133
 - DynamicMethod, instance 133

- filtres 234
 - d'images 118
 - filtres numériques 118
 - algorithme rapide 123
 - générer le code pour les éléments 128
 - pour images 118
 - netteté, filtre 119
 - fin, position (fonctionnalités du génome) 200
 - finis, automates 50
 - FIT (Framework for Integrated Test) 81–90
 - analyse syntaxique du HTML 87–90
 - classes 82
 - relations parmi 83
 - défi de la conception d'un framework 84
 - documents servant de tests 82
 - réseau ouvert 85
 - style de développement ouvert 90
 - Fixture, classe (Java) 85
 - fleuve, méandres 575
 - flex 303
 - flot dans un beau code 279
 - flou, filtre 119
 - flux Atom 551
 - Fogel, Karl 11–29
 - fonctionnalités (du génome) 200
 - densité de 201
 - fonctionnelle, décomposition du logiciel non
 - élémentaire 452
 - fonctions
 - appels de 314
 - arguments abstraits vers des pointeurs
 - d'arguments 295–300
 - atomically (Haskell) 408
 - choose (Haskell) 420
 - de partition (MapReduce) 397
 - forkIO (Haskell) 407, 419
 - JavaScript 137
 - matricielles clés en main fournies par
 - MATLAB 243
 - pointeurs de 304
 - fondamentales, instructions sur les ordinateurs
 - RISC et CISC 153
 - for, boucles
 - boucler sur des tableaux à N
 - dimensions 318
 - itérateurs Python vus comme des
 - prédicats 319
 - forme
 - method (syntax-case) 433
 - syntaxique (voir syntaxique, forme)
 - formé, bien (XML) 64
 - Fortran 245
 - bibliothèque LINPACK 245
 - nécessité d'utiliser Fortran 90 pour la
 - décomposition LU récursive 251
 - frameworks
 - appliqués au logiciel fonctionnant en
 - réseau 452
 - orientés objet 455
 - variabilités dans l'analyse 456
 - FreeBSD, système d'exploitation
 - fonctions et structures de donnée
 - d'interface 302
 - appel système read 304
 - implémentation de l'appel système
 - read 294
 - portion ne dépendant pas du système
 - de fichiers 295
 - indépendance de l'abstraction des
 - entrées/sorties de haut niveau 304
 - free_dicts, tableau (Python) 314
 - free-identifier=?, prédicat 445
 - fréquemment utilisés, mots dans eLocutor 524
 - fréquents, mots (dans eLocutor) 524
 - frugalité 276, 277
 - dans un beau code 275, 279
 - fullword immediate 158
- ## G
- Gaussian, élimination 264
 - GBrowse 222
 - gdb (GNU debugger) 490
 - déboguer
 - diviser les modifications en quatre
 - sous-ensembles ou plus 495
 - exécuter diff sur un code source 491
 - rechercher les états du programme causant
 - des échecs 500
 - GD::SVG, module (Perl) 223
 - Gene Sorter 227–238
 - beau, code 235–238
 - fonction filtrant les associations
 - traitant les jokers 236
 - dialogue avec un utilisateur par le
 - Web 229
 - implémentation de genome.ucsc.edu 230
 - interface utilisateur 228
 - polymorphisme 231, 235
 - génération de code 111–132
 - par des programmes lors de leur
 - exécution 112
 - (voir aussi code, génération)
 - gènes 228
 - champs d'expression des (Gene Sorter)
 - filtrage 234
 - codant les protéines 198
 - génie logiciel, principes 386
 - génomique 198
 - humain, séquençage 198

- navigateur 222
 - générique écrit en Ajax 223
 - reposant sur Bio::Graphics 222
- gen-var, assistant 435
- géométrie algorithmique 568–578, 582
- géré, code 118
- gestion
 - de données (mission MER) 336
 - de projet 362
 - du personnel (mission MER) 336
 - du temps (mission MER) 336
- gestionnaire
 - de fenêtres (Subversion) 17
 - de relations (ERP5) 361
 - d'événements de données ou connexion 464
- GFS (Google File System) 393
 - article plus détaillé sur la conception et l'implémentation 398
 - gestion des données d'entrée 395
- GHC (Glasgow Haskell Compiler) 422
- Ghemawat, Sanjay 387–400
- Glasgow Haskell Compiler (GHC) 422
- Glyph, classe (Bio::Graphics)
 - traitement dynamique des options 219
- glyphes (module Bio::Graphics)
 - ajouter de nouveaux 224
 - création dynamique de 211–216
 - génération de sous-glyphes 226
- GNU/Linux
- GNU/Linux (voir Linux)
- GnuPG 182
- Google
 - MapReduce 387
 - programmes MapReduce 393
 - résultats de recherche de haute qualité 61
- Google Maps 548
- Google News, recherches employant des fils Atom 551
- goto, instruction 127
- GPG 189, 190, 191
- graphe
 - direct de liens web 392
 - inversé de liens web 392
- graphiques, formats 202
- groupements de mots (eLocutor) 514
- Gulhati, Ashish 169–196

H

- hachages
 - construits par du code lisant columnDb 233
 - gènes filtrés dans Gene Sorter 235
 - non ordonnés (Perl) 205

- tables 51
 - changer la taille pour dictionnaires Python 313
 - charger un gros hachage 54
 - comparaison avec la recherche dichotomique 59
 - hacher des clés dans le même emplacement 312
 - masque de bits représentant la taille dans la structure PyDictObject 310
 - Ruby 51
- HAKMEM, mémo 157
- Hamming, distance de, entre deux vecteurs de bits 165
- Hamming, poids 153–167
 - comparaison de deux mots 159
 - compter les bits à 1 dans un tableau 160–166
 - diviser pour régner, stratégie 156, 157
 - filtre numérique
 - accès indexé direct vers un tableau modérément creux 166
 - méthodes élémentaires 154
 - somme et différence de deux mots 159
 - utilisations 165
- handicap moteur, eLocutor 511–529
- Harold, Elliotte Rusty 63–80
- Haskell 402–424
 - effets de bord et entrées/sorties 405–408
 - nature fonctionnelle de Haskell 407
 - résumé des opérations Haskell STM de base 413
- Hawking, Stephen 511
- Hayes, Brian 567–579
- héritage 136
 - des priorités 372–386
 - bogue au niveau utilisateur 370–384 (voir aussi priorités, héritage)
- historique
 - des transactions 411
 - service de flux du CIP 350
- Hoare, C.A.R. 32, 33, 40
- hook, méthodes de 457
 - dans la méthode de template run() de Logging_Server 461, 466
 - utilisation dans une implémentation de serveur de journalisation réactif 464
- hot swapping (remplacement à chaud) 353
- hôte, middleware d'infrastructure de 452
- hourglass, glyphe (exemple) 225
- HTML, rendu par Emacs W3 doté de synthèse vocale et ACSS 546
- HttpClient d'Apache Commons 482

- humain, génome, séquençage 198
- humaine, mémoire 235
- hygiénique, développement de macros
 - algorithme KFFD 430
 - conditions 429
 - implémentation de la portée lexicale pour le code source 430
- I**
- idéale, méthode de partitionnement (Quicksort) 40
- identificateurs
 - associer (mapping) à des bindings à l'aide de substitutions 434
 - captures de variables non intentionnelles
 - bindings et séquences introduits 428
 - portée dans la sortie plutôt que dans les entrées 427
 - comparaison 444
 - de templates 432
 - Python 308
- identifier?, prédicat 437
- id-label, procédure 438
- if
 - forme 442
 - instruction (JavaScript) 148
- IL, désassembleur 121
- ILGenerator, classe 126
 - Emit, méthode 126
- ImageFilter, classe 121
- images
 - brutes, opérations 113–118
 - (voir aussi raster, opérations)
 - cartes, pour la sortie de Bio::Graphics 221
 - code de traitement 111–130
 - FilterMethodCS 122–125
 - FilterMethodIL 125, 133
 - contenues dans un tableau NumPy 318
 - filtres 118
 - filtres numériques 118
 - (voir aussi filtres numériques pour images)
 - SVG (Scalable Vector Graphics) 223
- IMAP
 - goulets d'étranglement de performances
 - dans le magasin de courriers électroniques de Cryptonite après implémentation 193
 - modifier le magasin de courrier électronique Cryptonite pour utiliser 192
 - serveur vu comme un magasin de courrier électronique (Cryptonite) 184
- in situ, débogage 377
- inattendues, complications rencontrées par les développeurs 451
- inaudible (personnalité Emacspeak) 539
- include, forme 433
- indépendance
 - de l'abstraction des entrées/sorties sur FreeBSD 304
 - par rapport aux langages dans le CIP 339
- index inversé 392
- indexation 243
 - fonctionnalité 520
- indirection et couches 305
- individuelles, libertés à l'ère du numérique 169
 - protection par le côté privé des communications 194
- industriels, processus 357
- inégalité triangulaire 573
- infix, fonction (JavaScript) 141
- infixes, opérateurs (JavaScript) 140–143
- inflexion, voix monotone et sans (Emacspeak) 540
- <INPUT>, balises HTML cachées (hidden) 230
- instance, appels de méthode d', en Java 305
- instructions fondamentales sur les ordinateurs
 - RISC et CISC 153
- instructions (JavaScript) 147, 150
- intégral, recherches, en texte 60
- intégration à l'aide de REST
 - analyser du XML avec XPath 484
 - échange de données à l'aide du e-business
 - analyse syntaxique du code XML au moyen de XPath 483
 - protocoles 482
- intégration comme clé du succès dans les grosses applications 338
- interactives, applications web 201, 221
- interblocage
 - à un seul thread 383
 - verrouillage explicite dans le programme de transaction entre comptes 403
- interception en ordonnées d'une ligne verticale 570
- interface d'entrée (eLocutor) 515–528
 - chercher 527
 - clic long 517–519
 - clipboard (presse-papiers) 527
 - implémentation en cache 523
 - macros 527
 - mots communs et favoris 524
 - retracer les chemins 525
 - tampon de typage 525
 - Templates et Replace 522
 - typage simple 521
- interface HotkeyAdaptor (exemple) 480

interfaces matérielles 305
 intergiciel (voir middleware)
 Intermediate Language (IL) 118

- affectation et logique opérationnelle
 - reposant sur une pile d'évaluation virtuelle 126
- génération dynamique de 125
- généré par le compilateur C# 121
- optimisation de fonction pour le filtre numérique rapide 123

 interruptions 166
 intersections de segments 575
 intonation dans le discours 534
 inventeur, paradoxe de l' 35
 inversion

- des priorités 371
- du contrôle dans l'architecture d'exécution d'un framework orienté objet 456

 I/O en Haskell 405–408
 IPC (communication entre processus), mécanismes 453

- ACE, façades enveloppantes 459
- associer une stratégie de concurrence à 459

 IPC::Run, module (Perl) 190
 ISO-9660, système de fichiers 294
 isolation (transactions Haskell) 409
 isXMLDigit(), méthode 68
 itérateurs

- conception en NumPy 321–327
 - mise en place de l'itérateur 323
 - progression de l'itérateur 322
 - structure de l'itérateur 325
 - suivre le compteur de l'itérateur 324
 - terminaison de l'itérateur 322
- interface de l'itérateur en NumPy 327
- NumPy 320
- utilisation en NumPy 328–332
 - anecdotes 331
 - itération sur toutes les dimensions sauf une 328
 - itérations multiples 329

 itération 318

- à travers des dictionnaires Python 315
- sur l'ensemble d'une chaîne bloquante de manière cohérente (en Solaris) 373

 Iterative_Logging_Server 462

- évaluer 466

 Iterator, motif 464

J

J2EE (Java 2 Enterprise Edition) 337

- adaptabilité à la charge de travail 339
- basée sur SOA 338
- fiabilité 340

Java

- API pour accéder à des fonctions s'exécutant sur un serveur AS/400 477
- bibliothèque pour intégrer les vieux systèmes RPG dans les logiciels client/serveur 485
- bibliothèques de classe pour la programmation réseau 452
- bogue de l'algorithme de recherche dichotomique 94
- conversion d'un moteur d'expressions régulières en 8
- couches client et données du CIP 338
- générateur de nombres aléatoires et utilitaires Arrays 102
- implémentation de la recherche dichotomique 57
- implémentation en Python (Jython) 311
- programme de recherche dichotomique (exemple) 57
- programme Hello World 506
- table, consultation 73
- tableaux 57

 JavaScript 136–152

- constantes 144
- expressions 140
- fonctions 150
- instructions 146–150
- jetons 138
- littéraux tableau et objet 151
- opérateurs
 - d'affectation 143
 - infixes 140, 143
 - préfixes 143
- portée 144, 145, 146, 147
- priorité 139
- Simplified 136
- sites web créant une interaction intelligente côté client à travers 549
- table des symboles 137

 JDOM 64

- vérification du caractère du nom 67

 jetons

- JavaScript
 - arité 138
 - méthode std 147
- macros de pré-processeur C reposant sur des jetons 428

 jokers

- dans les expressions régulières 1
- shell 8

- journalisation
 - formateurs d'enregistrement des journaux 454
 - serveurs de journalisation
 - concurrents 466–472
 - séquentiels 462–466
 - service de connexion par réseau (application donnée en exemple) 453
- journalisation, framework serveur 454–462
 - serveurs de journalisation séquentiels
 - concepts principaux des frameworks orientés objet 455
 - parties communes 456
 - variation 458
- JUnit 95
- Jython 311
- K**
- Kent, Jim 227–238
- Kernighan, Brian 1–9
- KFFD (algorithme de développement de macro hygiénique) 430
- Kleene, Stephen 1
- Knuth, Donald 40
- Kolawa, Adam 265–280
- Kroah-Hartman, Greg 281–292
- Kuchling, Andrew 307–315
- L**
- lambda expressions
 - dans la forme define-syntax 431
- Lampson, Butler 293
- langage intermédiaire 118
 - (voir aussi Intermediate Language (IL))
- langage Sawzall pour l'analyse du journal d'une application 398
- langages de programmation 427
 - MATLAB 243
 - orientés objet (OO)
 - (voir aussi orientés objet (OO), langages de programmation)
 - TCL 532
- LAPACK 248–251
 - factorisation LU 248
 - SGBTRF routine 276
 - sous-programme SGBTRF 275–279
- led (left denotation), méthode 140
- let, forme 442
- letrec-syntax 432
 - formes, exp-letrec-syntax, transformateur 443
- let-syntax, forme 443
- lex (générateur d'analyseur lexical) 303
- lexicale, portée 430
- lexicaux, bindings de variables 435
- libgd 223
- liens, décompte (PageRank) 61
- ligne verticale 570
 - interception en ordonnées 570
- ligne-segment, algorithmes d'intersection 575
- limites d'erreur 262
- linéaire, sonde 313
- LINPACK 245–248
 - implémentation de la factorisation 246
- Linux
 - environnement de bureau pour un système de courrier électronique sécurisé 176
 - système de fichiers natif 293
- lisibilité du code 505–509
- Lisp
 - advice 533
 - extension, points d' 542
 - définition de procédure 568
 - macros 427
- liste, compréhension (Haskell) littéraires (en JavaScript) 137, 151
- listes d'associations 435
 - (voir aussi associations, listes d')
- literal, symbole dans l'analyseur syntaxique de JavaScript 138
- littéraires
 - de tableaux (JavaScript) 137, 151
 - d'objets (JavaScript) 137
- localité 396
- Logging_Server, classe de base abstraite 454, 459
 - méthodes open() et request() 460
 - run(), méthode 461
- Log_Handler, classe 461
- logiciel
 - bogues vus comme une « cuillerée d'eaux usées dans la barrique de vin » 369
 - commercial vendu sur étagère 337
 - correction et pureté de 369
 - génie, principes 386
 - lecteur de fils 551
 - manœuvré à l'aide d'un seul bouton 511–529
- logiques, opérateurs (court-circuiter) 142
- loi d'Amdahl 241
- long, motif le plus 5
 - reconnaître le plus long le plus à gauche 6
- lookdict() et lookdict_string, fonctions de recherche 315
- Luszczek, Piotr 239–264
- M**
- m4, développeur de macros 428
- Mac OS X 176

- machines
 - à un seul processeur multi-noyaux 241
 - vectérielles 241
 - vectorisation des algorithmes de l'algèbre linéaire 247
 - virtuelles 305
- macros
 - Common LISP 427
 - container_of 285
 - conversion de kobject en struct device 289
 - définition de macro syntax-case 431
 - développement hygiénique (voir hygiénique, développement de macros)
 - eLocutor 527
 - non développement hygiénique 430
 - offsetof 285
 - pour l'abstraction syntaxique 427
- magasins associatifs 51–54
- magazines 373
- Mail Daemon (cmaild) 180
 - suite de tests 180
- Mail::Client, module (Perl) 194
- Mail::Folder, module (Perl) 184
- Mail::Folder::Shadow, module (Perl) 185
- Mail::IMAPClient, module (Perl) 193
- main, action (Haskell) 407, 419
- Mak, Ronald 333–353
- mapping (Python) 310
- MapReduce 387–398, 399
 - implémentation en C++ du programme de comptage de mots (exemple) 399
 - ressources une lecture plus approfondie 397
- mark-object, icône auditory 542
- marques 434
 - apparaissant dans une enveloppe d'identifiant 438
 - de sommet 436
- Mars Exploration Rover (MER), mission 334
 - besoins 335
- match, fonction 4
- Matcher et Pattern, classes (Java) 8
- matchhere, fonction 4
- matchstar, fonction 5
- matériel 239–263
 - décomposition LU récursive 251–254
 - évolution du code élégant avec, impact de l'architecture des ordinateurs sur les algorithmes matriciels sous-programme DGEFR de LAPACK 248–251
 - sous-programmes DGEFR de LAPACK 248
- interfaces 305
- MATLAB 243
- matrices creuses 239
- Matsumoto, Yukihiro 505–509
- mbox, fichiers 181
- méandres de fleuve 575
- Mehta, Arun 511–521
- mémoire
 - adressable par contenus 50–54
 - conservation de la 270
 - humaine 235
 - modèles pour un tableau à N dimensions 319
 - transactionnelle 402, 424
 - utilisation de (sous-programme SGBSV de LAPACK) 272
 - vérification de la taille du problème et de la mémoire de l'ordinateur 278
- messenger, ARN (ARNm) 228
- messages 60
 - signés ou chiffrés 184
- Meta Classes (ERP5) 360
- meta, environnement (procédure exp) 440
- méta-caractères (dans les expressions régulières) 1, 7
- méta-données dans les couches du système de fichiers 301
- method, forme (syntax-case) 433
- méthode
 - d'instance, appels, en Java 305
 - HTTP POST 477
 - scientifique 492
 - synchronisée 402
- Microsoft Windows (version 1.0) 112–118
- middleware
 - couplage faible 338
 - dans une architecture multi-couches orientée-services 334
 - infrastructure d'hôte 452
- Middleware Monitor Utility (CIP) 350
- milieu, calcul 99
 - méthode calculateMidpoint() 100
- MIME, structure des messages de courrier électronique Cryptonite 184, 192
- minimiser les données entrées 498
 - automatisation des tests et ventilation des données en morceaux plus petits 499
 - problème de prototype avec une approche fragile 502
 - problème de recherche 493
 - déterminer si un changement dépend d'un autre 494
 - diviser pour régner, stratégie 494
 - processus systématique de 492
 - cohérent 492

- modèles
 - de calcul fork-join 259
 - de concurrence séquentielle 462–466
 - de constructeur 480–481
 - de mémoire pour un tableau à N dimensions 319
 - mathématiques 239
- modularité
 - dans le CIP 339
 - d'un beau code 271
- modulo
 - fonction 158
 - non signé, instruction 158
- Monnerat, Rafael 355–368
- monotone, voix sans inflexion (Emacspeak) 540
- moteur
 - d'expressions régulières (voir régulières, moteur d'expressions)
 - handicaps 511–528
- motif
 - distribution d'événements 456
 - fréquemment rencontré dans la programmation d'ordinateurs 427
 - le plus long 5
 - (voir aussi long, motif le plus)
 - ou brosse (objet graphique) 114
 - utilisé dans un framework serveur de journalisation orientée objet 454
 - vide, reconnaissance 5
- mots 524
 - complétion dans eLocutor 513, 521
 - complets 158
 - prochain (dans eLocutor) 521
 - programme de décompte (exemple) (voir décompte des mots, programme (exemple))
 - réservés 145
- mots clés
 - arguments dans les appels de fonctions (dictionnaires Python) 314
 - bindings de 431, 435
 - réserver les noms de 429
- mouvements
 - de données dans la hiérarchie mémoire 250
 - ERP5 357
- Movement, classe (ERP5) 358
 - représenter le retrait d'argent et le transfert de matériaux 360
- multithreadé 302
- Mutex, façade de l'enveloppant 459
- N**
- N dimensions, tableaux à, défis principaux dans les opérations 318
- naïve, fonction de colinéarité 570
- nature prototypale 137
- navigateurs de génome 222
 - reposant sur Bio::Graphics 222
- navigation dans eLocutor 525
- NB (taille du bloc) 250
- N-dimensionnel 317
- ne pas se répéter (principe DRY) 507
- Neomailbox, service de courrier électronique sécurisé 186
- .NET
 - Common Language Runtime 118
 - Intermediate Language 123
- netteté, filtre 119
- NFS (Network File System) 300
- nœuds (ERP5) 357
- nombres aléatoires, utilitaire générateur, dans NumPy 331
- noms
 - bien choisis 236
 - des éléments (XML 1.0) 65
 - espace, URI 78
 - une seule convention de style 238
 - validation en XML 64
- non contigus, tableaux 319
- non hygiénique, développement de macros 430
- non ordonné, hachage (Perl) 205
- notation 2
 - do (Haskell) 406
 - composition d'actions STM 410
- noyau Linux 281–292
 - devfs 282
 - gestion
 - de l'alimentation de périphériques 282
 - propre et persistante de périphérique 282
 - modèle unifié de pilote et de dispositif 285
 - comptage du référencement d'objets ans la couche virtuelle du système de fichiers 289
 - modèle unifié de pilote et de périphérique 282
 - adaptable à des quantités variables de périphériques (jusqu'à plusieurs milliers) 291
 - périphériques 281
 - noyau, primitives de synchronisation 372
- NPR, programmes 550
- nucléotides (A, T, C, G) 198
- nud (null denotation), méthode 140
- nul, reconnaissance de motif 5
- Null Mutex 466
- Null Object, motif 459
- nullfs, système de fichiers 298

- Null_Mutex, façade 462
 - numériques, filtres 118
 - pour images 118
 - (voir aussi filtres numériques pour images)
 - (voir aussi filtres numériques)
 - numéro de révision 12
 - NumPy (Python) 317–332
 - accéder à n'importe quelle zone d'un tableau par tranches 317
 - conception de l'itérateur 321–326
 - mise en place 323
 - progression 322
 - structure 325
 - terminaison 322
 - interface de l'itérateur 327
 - opérations dans des tableaux à N dimensions 318
 - origines de l'itérateur 320
 - utilisation de l'itérateur 328–332
 - anecdotes 331
 - pour les algorithmes à N dimensions 318
- O**
- object-caching, allocateur 373
 - objets
 - cible, emplacement 100
 - classes (Bio::Graphics) 202
 - choix 208, 211
 - de syntaxe 434
 - (voir aussi syntaxe, objets de)
 - de type « reader » et « rwriter » de fichiers (service de flux du CIP) 340
 - dynamiques 136
 - en cache, allocateur 373
 - littéraux (JavaScript) 137
 - polymorphes 231, 232
 - Observer, motif 456
 - Oliphant, Travis E. 317–332
 - OpCodes, classe 126
 - OpenBSD 176
 - OpenPGP 172
 - communiquer avec dans le système de courrier électronique
 - Cryptonite 190
 - magasin de courrier électronique connaissant MIME 184
 - opérations clés rendues accessibles aux utilisateurs 174
 - sécurité enchâssée dans Cryptonite 173
 - opérateurs
 - associatifs à droite (JavaScript) 142
 - de transposition (') 243
 - logiques (court-circuiter) 142
 - ternaires (?:) en JavaScript 142
 - opérateurs, priorités 135–152
 - descendante 135
 - JavaScript 136, 152
 - constantes 144
 - fonctions 150
 - instructions 147, 148
 - jetons 138
 - opérateurs préfixes 143
 - portée 144, 145, 146
 - priorité 139
 - table des symboles 137
 - opérations
 - brute ternaire 114
 - entre une matrice et un vecteur
 - reformulation de l'algèbre linéaire en termes de 241, 248
 - impliquant deux matrices
 - algorithmes de l'algèbre linéaire reformulés sous la forme de 241
 - modularité pour la performance et la transportabilité 250
 - impliquant deux vecteurs
 - désignées sous le nom de niveau 1 245
 - optimisation
 - cas de base pour le code récursif 319
 - vérificateur XML
 - conserver les espaces de noms en mémoire temporaire après vérification 78
 - consultation, table de 73–78
 - vérification d'un chiffre 70
 - Order, classe (ERP5) 363
 - catégorie élémentaire de source 362
 - order_validateData, script 365
 - ordinateurs RISC
 - à caches hiérarchisés 241
 - ordonné, hachage non (Perl) 205
 - ordonnées, interception d'une ligne verticale 570
 - orienté objet (OO)
 - framework pour le logiciel réseau 451–473
 - parties communes d'un framework serveur de journalisation 456
 - langages de programmation
 - expédition dynamique vers diverses méthodes de sous-classe 304
 - frameworks pour le logiciel fonctionnant en réseau 452
 - programmation, appel de méthode en Perl 206
 - OS X, Mac 176
 - Otte, William R. 451–452
 - outil udev (Linux) 287
 - ouvert, adressage 313

P

- PageRank 61
- paires clé/valeur (voir clé/valeur, paires)
- paires de base (base pairs, bp) 198
 - gestion de l'échelle dans la représentation des fonctionnalités du génome 201
- paquetage Java open source Log4j d'Apache 344
- paquets, designs 71
- parallélisme, application dans les recherches web 61
- paramètres à stocker dans des fichiers de configuration 352
- Params::Validate, module (Perl) 188
- parlée, sortie 532
- parole, doter Emacs de la 533–545
 - didacticiel advice 535
 - générer une sortie audible riche 535–540
 - implémentation de la file d'événements dans un serveur de synthèse vocale 533
 - premier jet d'implémentation 533
- Parse, classe (Java) 83–89
 - analyse syntaxique du code 87
 - doCells(), méthode 86
 - doRows(), méthode 86
 - doTables(), méthode 85
 - ignore, méthode 86
 - méthodes last() et more() 89
 - représentation d'un document HTML complet 89
- parseurs
 - JavaScript 136–152
 - jetons 138
 - opérateurs préfixes 143
 - priorité 139
 - top-down, opérateur 135
 - XML 71
- parsing, techniques en LISP 136
- partenaires métier 475–485
 - routage de services au moyen d'un modèle de constructeur 480
- partitionnement
 - en Quicksort 40
 - index en fonction d'une recherche dichotomique dans les tableaux de messages 62
 - méthode idéale (Quicksort) 40
- Pattern et Matcher, classes (Java) 8
- patterns (voir motifs)
- Patzer, Andrew 475–487
- pb (paires de bases) 198
- PBLAS (BLAS Parallèle) 257–259
- pentes
 - comparaison pour tester la colinéarité 572
 - mesures selon l'axe des y 571
- performances
 - conception médiocre, cause fondamentale des problèmes 272
 - implémentation du dictionnaire Python et 308
 - récurtivité et 318
 - tester pour la recherche dichotomique 108
- périphériques
 - USB et PCI dans un sysfs Linux 283
- périphériques (sous Linux) 281
 - pas de gestion propre persistante 282
 - struct device en guise de classe de base 282
- Perl 54
 - bibliothèque GD 223
 - BioPerl 198–202
 - chemin menant du prototype à la production 181
 - fonctionnalité AUTOLOAD 185
 - interface Tie 182
 - mode cperl d'Emacs 176
 - résumé des portions les plus tordues de la syntaxe 205
 - système de courrier électronique
 - Cryptonite 176
- Persistence::Database::SQL, classe 182
- Persistence::Object::Postgres, classe 182
- Persistence::Object::Simple, classe 182
- personality, propriétés du texte (Emacspeak) 536
 - conversion en codes de formatage audio 538
- Petzold, Charles 111–133
- Peyton Jones, Simon 401–424
- PGP 172
 - backend de chiffrement pour le courrier électronique Cryptonite 181
- phase
 - de compilation 7
 - d'exécution 7
- pile d'évaluation virtuelle (en IL) 126
- pitch-range, réglage (Emacspeak) 540
- pivots 242
- PKI (Public Key Infrastructure)
 - imperfections de 172
- planification des ressources d'entreprise, systèmes 355
 - (voir aussi entreprise, systèmes de planification des ressources)
- plate-forme Zope 355–360
 - (voir aussi Zope, plate-forme)
- plus court motif 5

- poids de Hamming 153–167
 - (voir aussi Hamming, poids)
 - pointeurs 5
 - de fonctions 304
 - fonctions, pointeurs de 295
 - manque d'une vérification de type à l'exécution sous le noyau Linux 286
 - passés à un struct device (sous Linux) 285
 - structures PyDictEntry 310
 - vers la structure vop_vector 294
 - points de code Unicode en XML 64
 - polymorphisme dans Gene Sorter 231–235
 - punctuation
 - caractères 538
 - réglage (Emacspeak) 540
 - pop, méthode 146
 - portail (CMF) 356
 - Portal, classes 360
 - portal_types, service 358
 - portée 144–147
 - fonctions 150
 - lexicale 430
 - nouvelle portée pour une fonction ou pour un bloc 146
 - position de fin (fonctionnalités du génome) 200
 - Postgres, base de données 182
 - Practice of Programming 2
 - Pratt, Vaughan 135
 - pre-action et post-action, hooks 546
 - predecessor, catégorie élémentaire 365
 - prédiction
 - complétion des mots et prochain mot dans eLocutor 521
 - évaluer l'efficacité du typage de for dans eLocutor 528
 - préfixes, opérateurs (en JavaScript) 143
 - pré-processeur, macros (C) 427
 - pré-traitement du texte avant envoi au serveur de synthèse vocale (tts-speak d'Emacspeak) 538
 - primitives de synchronisation du noyau 372
 - principes du génie logiciel 386
 - priorités 139
 - inversion 371
 - opérateurs 135–152
 - (voir aussi opérateurs, priorités)
 - priorités, héritage 372–386
 - au niveau utilisateur 376–383
 - implémentation pour des verrous au niveau utilisateur 380
 - lever des verrous au niveau utilisateur 383
 - obtenir un résultat correct 374
 - privées, clés 175
 - probables, causes les plus, d'abord 493
 - problèmes
 - capture de variables 428–431
 - (voir aussi variables, problèmes de capture)
 - matrices denses 240
 - Santa Claus (programmation concurrentielle avec STM) 414–422
 - processeurs partitionnés pour des programmes parallélisés 390
 - prochain mot (dans eLocutor) 521
 - programmation
 - distribuée avec MapReduce 387–399
 - exécution dans son ensemble 394
 - extensions du modèle 397
 - graphe inversé de liens web 392
 - index inversé 392
 - modèle de programmation 390
 - tri distribué 393
 - dynamique 36
 - langages 427
 - modulaire
 - non supportée par les verrous et les variables conditionnelles 405
 - programmation concurrentielle avec STM 423
 - programmes parallèles moins modulaires 401
 - orientée objet (voir orientée objet (OO), programmation)
 - Windows en C# 121
 - programme de comptage de mots (exemple) 387–391
 - décomposition du problème en deux fonctions, Map et Reduce 391
 - parallélisé 388
 - avec processeurs partitionnés 389
 - avec stockage partitionné 388
- programmes
 - d'exemple 273
 - DGEMM 250
 - DGETF2 250
 - DTRSM 249
 - générant du code lors de leur exécution 112
 - multithread écrits en C 287
 - NPR 550
 - parallèles 401
 - pilote pour les fonctions Map et Reduce (exemple) 391
- Programming Pearls 32, 93
- progression, barre de, pour les programmes volumineux 56

Project (ERP5) 362–368
 coder 363–368
 attributs de domaine et comportement
 d'un projet représentatif 365
 relations avec Trade 362
 projet 362
 gestion 362
 projet BioMoby 477
 prononciation, règles (pré-traitement du texte
 dans Emacspeak) 538
 propriétés
 des verrous 380
 feuilles de 359
 protéines
 gènes codant 198
 régulatrices attachées à un site spécifique de
 l'ADN 217
 site d'association, et ADN 217
 protocole e-business
 échange de données avec 482–487
 Rosettanet 477
 échange de requêtes et de réponses
 avec 482–487
 site web 482
 prototypale, nature 137
 publique
 clé liée aux contacts dans le carnet
 d'adresses de l'utilisateur
 (Cryptonite) 175
 cryptographie à clé 172
 clé, authentification par 172
 puissances d'association (jetons JavaScript) 140
 +, opérateur 140
 put-text-property, fonction 536
 PyArrayIterObject, structure 325, 326
 PyDictEntry, structure 310
 PyDictObject, structure 309
 champ ma_lookup 312
 PyDict_SetItem(), fonction 315
 PyIntObject, type 312
 PyObject, structure 312
 PyStringMap, classe (Jython) 311
 PyStringObject, type 312
 Python 314
 algorithme de débogage delta
 implémentation 497
 dictionnaires 307–315
 particulariser un cas 311
 ERP5 355
 structure de document implémentée
 comme une classe de portail 355
 itérateurs 319
 NumPy 317–332
 conception de l'itérateur 321
 site de téléchargement 307

python, commande (Mac OS et
 GNU/Linux) 307

Q

Quicksort, programme 31–43
 aphorismes portant sur la beauté 42
 faire décroître le code tout en augmentant
 calcul en faisant sortir le code de la
 boucle 36
 calcul en ôtant la boucle interne 37
 calcul par programmation
 dynamique 36
 compter les comparaisons employées
 par un tri de tableau 33
 fonction 33–38
 incrémenter la sortie de la boucle 34
 insérer un élément dans un arbre
 binaire de recherche 40
 moyenne des comparaisons sous forme
 de pseudo-code 36
 squelette avec un seul argument de
 taille 35
 squelette implémenté comme une
 fonction 35
 squelette réduit au décompte 34
 techniques utilisées 39
 version finale du calcul 38
 idées pour raffiner le code 42
 quicksort(), fonction 32
 quote, forme (syntax-case) 435

R

Rails 507
 Rake (outil de construction) 508
 Rakefile (exemple) 508
 Raman, T. V. 531–554
 rapports de tâches (ERP5) 363
 raster, opérations 113–118
 fonctions BitBlt et StretchBlt 114
 Reactive_Logging_Server
 classe 463
 évaluer 466
 programme principal utilisant l'interface
 de programmation socket 465
 Reactor, motif 456
 Recall, framework de réplication 184
 recherches, captures des variables 431
 recherches 58
 adressable par contenus, stockage 50, 54
 classer les résultats 61
 dichotomique, recherche 57
 données de blog 46
 écrire un algorithme de recherche 54

- emacspeak-websearch, module de
 - recherche orientée par les tâches 546–549
- Google News employant des fils Atom 551
- messages 60
- régulières, expressions 47–51
 - utilisation dans un programme qui affiche les lignes 48–50
- temps impliqué dans l'exécution et la programmation 45
- web 60, 61
- reconnaissance de motif vide 5
- reconnaître un seul caractère dans les expressions régulières 3
- recouvrements de voix (Emacspeak) 540
- Recursive Descent 135
- récurtivité 7
 - factorisation LU 251–257
- redondance 506
- ré-entrantes, fonctions 238
- réexécution (transactions en STM) 411
- ré-expression et symétrie de programmation 36, 40
- ref(), fonction (Perl) 219
- references (variable) 429
- réglage des ponctuations (Emacspeak) 540
- règles de prononciation (pré-traitement du texte dans Emacspeak) 538
- régulateur, ADN 198
- régulatrice, protéine attachée à un site spécifique de l'ADN 217
- régulières, expressions 1–8, 47–50
 - associer (mapping) à des actions avec lex 303
 - notation 2
 - pour la validité des données dans le système de courrier électronique Cryptonite 188
 - programme affichant les lignes (exemple) 47–51
- régulières, moteur d'expressions
 - conditions de terminaison 6
 - implémentation 3
 - raisons de la compacité du code 9
- relationnelles, bases de données
 - dans eLocutor, bases de données de prédiction 521
 - ERP5 357
- relations, gestionnaires (ERP5) 361
- répertoire
 - arborescences (voir arborescences de répertoires)
 - caché pour les messages de boîte de réception 186
- repeuplement dynamique d'arbre dans eLocutor 519
- Replace, fonctionnalité (eLocutor) 522
- réplication 184, 193
- Replication::Recall, module (Perl) 184
- représentations (algorithme de développement syntax-case) 434
- requêtes
 - d'entrées/sorties, vecteur 304
 - GET et POST via HTTP 476
- réseau de confiance 172
 - visibilité de l'information dans la vue Key Ring de Cryptonite 174
- réseau, logiciel fonctionnant en 451–455, 456
 - conception d'un framework serveur de journalisation 455–462
 - assembler le tout 459
 - parties communes 456
 - implémentation des serveurs de journalisation concurrents 467–473
 - séquentiels 462–467
 - journaliser l'application du service 453
- réservés, mots 145
- ressources d'entreprise, systèmes de planification 355
 - (voir aussi entreprise, systèmes de planification des ressources)
- ressources (ERP5) 357
- REST (Representational State Transfer) 476–487
 - proposer des services à des clients externes 476–480
 - définition de l'interface des services 477–480
 - routage de service au moyen d'un modèle de constructeur 481
- restrictions introduites dans un code 271
- résultats de recherche, classer 61
- return, instruction (JavaScript) 149
- réutilisation
 - de données pour réduire le trafic mémoire 250
 - d'un code, favorisée par une bonne conception et par la clarté 274
- réveiller les threads en Solaris 370
- révision, numéro de 12
- rivière, méandres 575
- robustesse 350
 - architecture d'un système 337, 338
 - CIP (Collaborative Information Portal) hot swapping 353
 - étude de cas : service de flux 340–343
 - architecture du service 340
 - fiabilité 343
 - fonctionnalité 340
 - hot swapping (remplacement à chaud) 353
 - reconfiguration dynamique 351

- RowFixture, classe (Java) 83
- RPG 484
- RSS et Atom, fils 551
- rsync, algorithme de somme de contrôle
 - déroulante 11
- Ruby, langage de programmation 48–54
 - compter le nombre de chargements d'un article 52
 - design simplifié et dénudé 50
 - implémentation
 - de hachage 56
 - de tableau 57
 - optimiser le programme qui rend compte des articles les plus populaires 53
 - prise en charge du code esthétique
 - brièveté et DRY (ne pas se répéter) 507
 - brièveté (exemple de Hello World) 506
 - simplicité 508
- runtime, environnement (procédure exp) 440
- S**
- same-marks?, prédicat 439
- saut ou branchement, instruction (IL) 127
- Savioia, Alberto 91–110
- Scalable Vector Graphics (SVG), images 223
- ScaLAPACK, factorisation LU 255
- Scheme, langage
 - développer les expressions renfermant des macros en expressions du langage de base 445
 - primitives de conversion des chaînes vers et depuis les symboles 435
 - syntax-case 431
 - syntaxe de quasi-apostrophe pour créer la structure de liste 435
 - transformation d'une forme or en let et if 428
- Schmidt, Douglas C. 451–470
- scientifique, méthode 492
- scrolling dans eLocutor 526
- secret des communications 194
- sécurisées, communications 169–196
 - bidouille la civilisation 195
 - complexité de l'envoi sécurisé de messages 172
 - protection de la sphère privée pour les libertés individuelles 195
 - système de courrier électronique
 - Cryptonite 170–194
- Seiwald, Christopher 555–565
- self-evaluating? prédicat 437
- sémantique
 - dépendant du contexte 542–544
 - vectorielle par hôte 392
- séquences
 - ADN 198
 - ARN 198
- séquentielle, modèles de concurrence 462–466, 467
- sérialisation 179
- serveurs
 - d'applications IBM WebSphere 477
 - de synthèse vocale 532
- Service, classe (Cryptonite) 178
- service de flux (CIP) 340–350
 - architecture du service 340
 - fiabilité 343, 350
 - conservation de l'historique 343
 - historique 350
 - fonctionnalité 340
- services, fourniture à des clients externes 476–480
- services web
 - communication entre les applications clientes et les services
 - middleware 339
 - utilisation par le CIP du service de flux
 - pour gérer les requêtes des clients et fournir les réponses 340
- servlet Java s'exécutant sur un serveur
 - d'application IBM WebSphere 477
- Session, classe (Java) 84
- session, clé de 230
- s-expression 432
 - conversion vers et depuis l'objet de syntaxe à l'aide de datum->syntax 445
 - représenter une forme quote 442
- shards 62
- shift right immediates, instruction 157
- signature de type (Haskell) 405
- signés ou chiffrés, messages 184
- Simple Object Access Protocol (SOAP) 476
- SimpleItem, feuille de propriétés 359
- simplicité dans le code 274
 - Ruby, langage de programmation 507
- Simplified, JavaScript 136
- single-binding, formes
 - let 443
 - letrec-syntax 443
- site
 - débogage sur 377
 - web de cet ouvrage 399
- slicing 318
- SMP (symmetric multiprocessing)
 - machines 241
- SOA (service-oriented architecture) à trois couches 337
- SOAP 476

- SOCK_Acceptor, type 466
- Solaris
 - développement du sous-système critique du noyau 370
 - héritage
 - de priorités au niveau utilisateur 376
 - des primitives pour les sous-programmes de synchronisation du noyau 372
 - interaction entre l'allocateur de mémoire du noyau et Zettabyte Filesystem (ZFS) 372
 - réveiller les threads 370
- solutions calculées et équations mathématiques 265
- somme de contrôle déroulante, algorithme rsync 11
- sommet, marque de 436
- sonde linéaire 313
- Sorter 231
- sortie parlée 532
- source_project, catégorie élémentaire 364
 - rapatrier les tâches d'un projet 364
- souris, bouton droit, vu comme une unique saisie binaire pour eLocutor 515
- sous-fonctionnalités et sous-sous-fonctionnalités (fonctionnalités du génome) 200
- sous-programmes 279
 - DGETF2, goulet d'étranglement pour les ordinateurs aux processeurs rapides 251
 - pilotes 274
 - SGBSV (bibliothèque LAPACK) 267–273
 - détails d'implémentation 274
- sous-système 274
- soustraction, instruction 157
- spécifications pour un code 273
- spin, verrou (verrou de thread) dans Solaris 374
- Spinellis, Diomidis 293–306
- SQL
 - backend de stockage de courrier électronique (Cryptonite) 184
 - base de données 230
 - code rapatriant les données des champs d'association Gene Sorter 234
- sqrt renvoyant un résultat irrationnel 574
- std (statement denotation), méthode (jetons JavaScript) 147
- Stein, Lincoln 197–226
- STM (Software Transactional Memory) 402–423
 - implémentation de la mémoire transactionnelle 411
 - problème de Santa Claus (exemple de programme) 414–422
 - transactions en Haskell 409
- stockage
 - de données 240
 - partitionné pour un programme parallélisé 389
- Store, classe (Java) 85
- stratégie du « diviser pour régner » (voir diviser pour régner, stratégie)
- Strategy, motif 459
- streamer, service (CIP), streaminess (interface Subversion) 24
- StretchBlt, fonction (Windows 1.0) 112
 - raster, opérations 114
- strip, procédure 436
- structures
 - basées sur l'héritage 274
 - champ de la structure de base définissant le type de pointeur 286
 - contenir un bloc d'arguments de fonction 302
 - dans les programmes multithread 287–290
 - d'arbre présentant des options dans eLocutor 515–519
 - de contrôle
 - définies par l'utilisateur (en Haskell) 408
 - spécifiques d'une application 423
 - dentry (Linux) 291
 - des périphériques (Linux) 282
 - comptage de références 287
 - usage de la mémoire 291
 - héritage et manipulation sous le noyau Linux 283–287
 - inode (Linux) 291
 - kobject (Linux) 288
 - modifiée pour utiliser struct kref 290
 - kref (Linux) 289
 - les structures de dictionnaire ne sont plus utilisées 314
 - MIME des messages de courrier électronique Cryptonite 184–192
 - PyObject 312
 - usb_interface (Linux) 285
 - pointeur 286
 - structures imbriquées 290
- substitutions 434
 - associer (mapping) un nom symbolique et la liste des marques à une étiquette 435
- subtracts, instruction 157
- Subversion 11–28
- surcharge 274
- SVG (Scalable Vector Graphics), images 223

svn_error_t (type d'erreur de Subversion) 17
 svnsync, fonctionnalité (Subversion) 29
 switch, instructions (Java) 73
 problèmes posés par des groupes de cas plus volumineux 75
 symboles, table (analyseur syntaxique JavaScript) 137
 symétrie 36
 symmetric multiprocessing (SMP)
 machines 241
 synchrones, événements, interfaces de programmation démultiplexant (systèmes d'exploitation) 463
 synchronisation du noyau, primitives 372
 SyncML, protocole 356
 syntax-car, opérateur 444
 syntax-case
 algorithme de développement 434–446
 syntaxe, erreurs de 437
 exemple d'expander 446–449
 forme 431
 introduction à 431–434
 définition de macros 431
 forme d'entrée ou de sortie suivie de points de suspension 432
 hygiène 432
 système 431
 syntax-cdr, opérateur 444
 syntaxe, objets de 434
 analyse syntaxique et construction 444
 conversion vers et depuis les s-expressions 445
 simplifier et dénuder 436
 structure non atomique 437
 syntax-error 437
 syntaxique
 abstraction 427–449
 analyse, techniques en LISP 136
 analyseur (voir parseurs)
 forme, utilisation par un programme développeur 441
 syntax-pair?, opérateur 444
 syntax-rules, système 431
 synthèse vocale 532–544
 avec Emacs 533–545
 (voir aussi parole, doter Emacs de la)
 dans eLocutor 527
 matériel (DECTalk Express) 532
 serveurs 532
 sysfs (Linux) 283
 systématique, débogage 492
 système de courrier électronique (Cryptonite) 170–194
 (voir aussi courrier électronique, système (Cryptonite))

systèmes
 linéaires denses 242
 multi-noyaux 241
 impact sur les algorithmes matriciels 241
 parallèles à mémoire distribuée 241
 systèmes de fichiers
 code d'accès aux systèmes de fichiers 294
 couches du système de fichiers 299–302
 utilisation par FreeBSD de l'indirection pour abstraire les arguments de la fonction read 295–300
 systèmes d'exploitation
 bases de données ou devfs pour gérer le nom des périphériques 282
 prendre en charge différents systèmes de fichiers 293
 couches du système de fichiers 299, 300
 Unix 295
 (voir aussi Unix, système d'exploitation)
 verrous pour les structures critiques des systèmes d'exploitation 302

T

table
 consultation 73
 des symboles (analyseur syntaxique JavaScript) 137
 tableaux
 à deux dimensions 317
 à N dimensions (voir N dimensions, tableaux à)
 à trois dimensions 317
 à une dimension 317
 et algorithme de filtre numérique en C# 122
 filtre numérique
 stocker l'octet dans le tableau de destination 132
 zéro octet stocké dans le tableau de destination 132
 Hamming, poids
 circuits CSA 161–165
 littéraux (JavaScript) 137, 151
 non contigus 319
 pixels de bitmap stockés dans 121
 tâches copiées 396
 tâches (ERP5) 363
 associées à un projet 363
 prédécesseurs de tâche 365
 rapports 363
 workflow de Task 365
 taille du bloc (NB) 250

- tampon, utilisation de 305
 - Task Report, workflow 367
 - taskReport_notifyAssignee, script 367
 - TCL, langage 532
 - technologies XML 356
 - Template Method, motif 454
 - structure et application à un serveur de journalisation 457
 - templates
 - dans eLocutor 522
 - d'URL (Emacspeak) 549
 - (voir aussi URL, templates (Emacspeak))
 - identificateur de 432
 - spécifiant la sortie dans une forme définie-syntaxe 432
 - temporaire, placer en mémoire (voir cache, placer en mémoire)
 - termination
 - conditions (moteur d'expressions régulières) 6
 - zéros de, dans un mot 165
 - ternaire
 - opérateur (?) en JavaScript 142
 - opération brute 114
 - tections 62
 - tester 91–110
 - beauté dans les tests 92
 - dichotomique, recherche 93–108
 - bord, tests de valeur au 101
 - test unitaire 97
 - tests aléatoires 102–107
 - framework de tests JUnit 95
 - importance des programmes tests 272
 - tests unitaires 97
 - texte 534
 - différence (Subversion) 17
 - intégral, recherches en 60
 - pré-traitement avant envoi au serveur de synthèse vocale (tts-speak d'Emacspeak) 538
 - Text::Template, module (Perl) 182
 - Thompson, Ken 2
 - threads
 - bloqués, chaîne de 372
 - réveiller en Solaris 370
 - verrou 374
 - timbre, réglage pour une voix plate (Emacspeak) 540
 - tolérance aux erreurs et aux incidents 396
 - Top Down Operator Precedence 135
 - tourniquets (Solaris) 370
 - (voir aussi turnstiles (Solaris))
 - tracks (module Bio::Graphics) 203–206
 - options de configuration 206
 - positions et forces relatives de l'ADN et du site d'association des protéines 217
 - sous-classe de Glyph 210
 - Trade BT 363
 - traitement d'images, code 111–130
 - (voir aussi image, code de traitement)
 - tranches, découper en 318
 - transactions (en Haskell) 408
 - transfert de données dans la hiérarchie de la mémoire 240
 - transformateurs, core 441, 444
 - transformation d'arbre 12, 15
 - transposons 228
 - travail, copie de (Subversion) 15
 - tri
 - distribué 393
 - et relation d'ordre
 - garantis(MapReduce) 397
 - fonction 32
 - MapReduce 396
 - triangle-collinear, fonction 573
 - triangulaire, inégalité 573
 - trois dimensions, tableaux 317
 - trouver la définition d'un nom (analyseur syntaxique JavaScript) 145
 - tts-format-text-and-speak, fonction (Emacspeak) 538
 - tts-speak
 - fonction 535
 - module (Emacspeak) 538
 - tts-speak-using-voice, fonction 539
 - turnstile_block(), fonction 370
 - turnstile_pi_waive(), fonction 384
 - turnstiles (Solaris) 370
 - priorités, inversion 371
 - typage dans eLocutor 521–525
 - accélérer 513
 - type (bindings) 435
 - TypeAdapter, classe (Java) 83
 - types de donnée Haskell, distinction entre des actions et des valeurs pures 422
 - typés dynamiquement, langages 51
- ## U
- UBM (Unified Business Model) 357
 - ufs, système de fichiers (BSD 4.2) 296
 - umapfs, système de fichiers 296
 - s'interposer par-dessus l'implémentation NFS 300
 - une dimension, tableau à, et algorithme de filtre numérique en C# 122
 - une dimension, tableaux 317
 - Unicode, points de code en XML 64
 - Unified Business Model (UBM) 357

Unix, système d'exploitation 295
 interface de programmation socket 451
 sources internes et externes 304
 variantes de l'appel système read 304
 unsigned modulus, instruction 158
 URI (Uniform Resource Identifiers) 78
 espace de noms 78
 URL, modèles (Emacspeak) 549
 rapatrier la météo d'une ville ou d'un état 551
 USB, clés 293
 utilisateur
 bogue d'héritage de priorité au niveau utilisateur 370–384
 clé pour Gene Sorter 230

V

valeur/clé, paires (voir clé/valeur, paires)
 valeurs
 au bord, tests de 97
 (voir aussi bord, tests de valeur au)
 bindings 435
 de première classe 422
 rechercher 50
 validation
 saisies de l'utilisateur 188
 XML 63, 64
 var, instruction (JavaScript) 147
 variabilités dans l'analyse de frameworks orientés objet 456
 variables
 CGI 229
 CGI cachées 230
 (voir aussi cachées, variables CGI)
 conditionnelles
 Blocage dans les transactions entre comptes bancaires 403
 coordination de programmes concurrents 402
 programmation modulaire non supportée 404
 définition dans le bloc courant (JavaScript) 147
 lire ou écrire dans une variable mutable 406
 mutables 406
 portée en JavaScript 144, 145, 146, 147
 problèmes de capture 428–431
 résoudre avec un développement de macro hygiénique 429
 variables ou communes, analyse des parties 456
 VB (Visual Basic) 513
 contrôle TreeView 516
 fonctionnalité d'indexation accélérant le rapatriement 520

vecteur de requêtes d'entrées/sorties 304
 vectorisation et parallélisme 240
 vérificateurs XML 63–80
 (voir aussi XML, vérificateurs)
 vérification de type pour les pointeurs 286
 verrouillage 302
 bogue d'héritage de priorité au niveau utilisateur de Solaris 370–386
 comptes bancaires gérés par verrous 402
 gestion des assertions de verrouillage 303
 manque de modularité 404
 problèmes avec les verrous
 oublier qu'un thread est réveillé ou qu'il faut le réveiller 404
 prendre les mauvais verrous 404
 prendre trop de verrous 404
 prendre trop peu de verrous 404
 récupérer les erreurs 404
 problèmes liés au verrouillage 403
 stratégies pour 454
 verrous
 de thread 374
 propriété 380
 verticale, ligne 570
 interception en ordonnées 570
 vide, reconnaissance de motif 5
 virtuelle
 machine 305
 pile d'évaluation (en IL) 126
 vnode, interface d'appel 302
 vnodeop_desc, structure 300
 vocale, synthèse 532–544
 avec Emacs 533–545
 (voir aussi parole, doter Emacs de la)
 dans eLocutor 527
 serveurs 532
 voice-monotone 540
 voix
 monotone, sans inflexion (Emacspeak) 540
 recouvrements (Emacspeak) 540
 vop_generic_args, structure (FreeBSD) 300
 vop_vector, structure (FreeBSD) 294
 peupler les champs bypass et default 298
 pointeur
 vers une fonction de contournement 298
 vers une structure vop_vector de la couche sous-jacente du système de fichiers 298
 prise en charge des couches du système de fichiers 296

W

Warren, Henry S., Jr. 153–167

web
 navigateurs de génome reposant sur
 Bio::Graphics 222
 outils orientés, dans Emacspeak 549–552
 lecteurs de fils 551
 recherches 60–61
Web of trust 172
 (voir aussi réseau de confiance)
Wheeler 305
while, instruction (JavaScript) 148
wildcards (voir jokers)
Windows Forms, code (programme
 ImageClip) 120
Windows, systèmes d'exploitation
 interface de programmation de
 threads 451
 système de fichiers natif 293
Wingerd, Laura 555–565
with-syntax, forme 433
workflows
 de Task Report 367
 ERP5
 implémentation du comportement de
 tâche 365
 Task Report 368
Wrapper Facade, motif 454, 458
wrapper, façades 466
wraps (voir enveloppes)

X

XML
 analyse syntaxique de données à l'aide de
 XPath 483–487
 données requête/réponse via HTTP
 POST 477
 spécification de la Version 1.0 66
 technologies 356
XML, vérificateurs 63–80
 deuxième optimisation 71
 première optimisation 69
 quatrième optimisation 78
 rôle de la validation 64
 troisième optimisation 73, 78
XMLObject, classe 359

Y

yacc (générateur d'analyseurs syntaxiques) 303
Yahoo! Maps 547

Z

Zeller, Andreas 489–503
zéros de terminaison dans un mot 165
ZFS (Zettabyte Filesystem) 372
Zimmermann, Phil 172
ZODB 356
Zope, plate-forme 355–360
 CMF (Content Management
 Framework) 355
 composants clés utilisés par ERP5 356
ZPT (Zope Page Templates) 356

