

Herb Sutter

Mieux
programmer
en C++

47 problèmes pratiques résolus

E Eyrolles

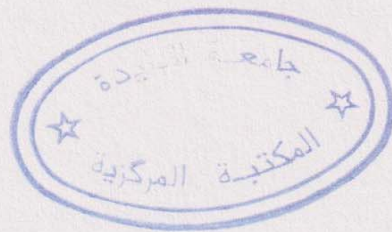
005-338-1

2-005-338-1


Mieux programmer en C++

47 problèmes pratiques résolus

Herb Sutter



Traduit de l'anglais par Thomas Pétillon

 **Eyrolles**

Sommaire

Préface

Avant-propos

Programmation générique avec la bibliothèque standard C++

Pb n° 1.	Itérateurs	1
Pb n° 2.	Chaînes insensibles à la casse (1 ^{re} partie)	4
Pb n° 3.	Chaînes insensibles à la casse (2 ^e partie)	7
Pb n° 4.	Conteneurs génériques réutilisables (1 ^{re} partie)	9
Pb n° 5.	Conteneurs génériques réutilisables (2 ^e partie)	10
Pb n° 6.	Objets temporaires	19
Pb n° 7.	Algorithmes standards	24

Gestion des exceptions

Pb n° 8.	Écrire du code robuste aux exceptions (1 ^{re} partie)	28
Pb n° 9.	Écrire du code robuste aux exceptions (2 ^e partie)	32
Pb n° 10.	Écrire du code robuste aux exceptions (3 ^e partie)	35
Pb n° 11.	Écrire du code robuste aux exceptions (4 ^e partie)	41
Pb n° 12.	Écrire du code robuste aux exceptions (5 ^e partie)	43
Pb n° 13.	Écrire du code robuste aux exceptions (6 ^e partie)	48
Pb n° 14.	Écrire du code robuste aux exceptions (7 ^e partie)	54
Pb n° 15.	Écrire du code robuste aux exceptions (8 ^e partie)	56
Pb n° 16.	Écrire du code robuste aux exceptions (9 ^e partie)	58
Pb n° 17.	Écrire du code robuste aux exceptions (10 ^e partie)	62
Pb n° 18.	Complexité du code (1 ^{re} partie)	64
Pb n° 19.	Complexité du code (2 ^e partie)	66

Conception de classes, héritage

Pb n° 20.	Conception d'une classe	71
Pb n° 21.	Redéfinition de fonctions virtuelles	78
Pb n° 22.	Relations entre classes (1 ^{re} partie)	83

Pb n° 23. Relations entre classes (2 ^e partie)	86
Pb n° 24. Utiliser l'héritage sans en abuser	93
Pb n° 25. Programmation orientée objet	102
Pare-feu logiciels	105
Pb n° 26. Éviter les compilations inutiles (1 ^{re} partie)	105
Pb n° 27. Éviter les compilations inutiles (2 ^e partie)	108
Pb n° 28. Éviter les compilations inutiles (3 ^e partie)	112
Pb n° 29. Pare-feu logiciels	115
Pb n° 29. La technique du « Pimpl Rapide »	117
Résolution de noms, espaces de nommage, principe d'interface	125
Pb n° 31. Résolution de noms, principe d'interface (1 ^{re} partie)	125
Pb n° 32. Résolution de noms, principe d'interface (2 ^e partie)	128
Pb n° 33. Résolution de noms, interface de classe (3 ^e partie)	137
Pb n° 34. Résolution de noms, interface d'une classe (4 ^e partie)	140
Gestion de la mémoire	147
Pb n° 35. Gestion de la mémoire (1 ^{re} partie)	147
Pb n° 36. Gestion de la mémoire (2 ^e partie)	149
Pb n° 37. auto_ptr	155
Quelques pièges à éviter	167
Pb n° 38. Auto-affectation	167
Pb n° 39. Conversions automatiques	170
Pb n° 40. Durée de vie des objets (1 ^{re} partie)	171
Pb n° 41. Durée de vie des objets (2 ^e partie)	173
Compléments divers	181
Pb n° 42. Initialisation de variables	181
Pb n° 43. Du bon usage de const	183
Pb n° 44. Opérateurs de casting	190
Pb n° 45. Le type bool	195
Pb n° 46. Transferts d'appel	198
Pb n° 47. Contrôle du flot d'exécution	201
Post-scriptum	209
Bibliographie	211
Index	213