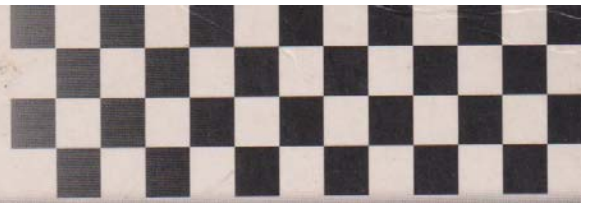
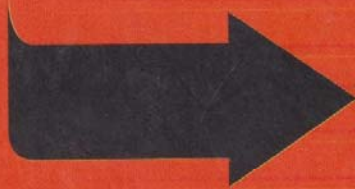


Le Starter



Programmation

en



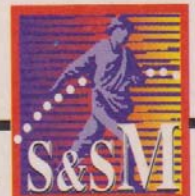
C++

Un kit avec toutes les informations et tous les outils nécessaires pour une prise en main immédiate

- ➔ Découvrir le langage C++**
- ➔ Installer et configurer Borland C++ 4.02**
- ➔ Déboguer et compiler vos programmes C++ dans l'EDI**



Gratuit ! 1 CD-ROM contenant Borland C++ 4.02, un environnement de développement C++ professionnel



www.ssm.fr

2-005-431-1

Programmation en C++



Simon & Schuster Macmillan

Retrouvez
tous nos livres sur
www.ssm.fr

Table des matières

Préface	XIX
---------------	-----

I. LE LANGAGE C++

PARTIE 1. PRÉSENTATION DU C++

Chapitre 1. Bien débiter en C++	5
Les atouts du C++	5
<i>La programmation : procédurale, structurée ou orientée objet</i>	6
<i>C++ et programmation orientée objet</i>	8
<i>Evolution du C++</i>	9
<i>C et C++ : différences</i>	10
<i>Dois-je apprendre d'abord le langage C ?</i>	10
Préparation d'un programme	10
C++, ANSI C++, Windows et autres zones d'ombre	11
<i>Environnement de développement</i>	12
Compilation et liaison du code source	13
<i>Compilation dans un environnement de développement intégré</i>	13
<i>Liaison d'un programme</i>	14
Le cycle de développement	14
Votre premier programme C++ : Bonjour.cpp	15
Erreurs de compilation	17
Récapitulatif	17
Questions-réponses	18
Chapitre 2. Structure d'un programme C++	19
Structure d'un programme simple	19
<i>Inclusion</i>	20
<i>Analyse séquentielle du programme</i>	20

Commentaires	21
<i>Différents types de commentaires</i>	21
<i>Règles d'utilisation</i>	23
Fonctions	23
<i>Utilisation de fonctions</i>	25
Récapitulatif	27
Questions-réponses	27
Chapitre 3. Variables et constantes	29
Variables	29
<i>Généralités sur la mémoire</i>	30
<i>Taille des entiers</i>	30
<i>Signée ou non signée ?</i>	32
<i>Principaux types de variables</i>	32
Déclaration de variables	34
<i>Différenciation des majuscules et des minuscules</i>	35
<i>Mots clés</i>	35
Création de plusieurs variables	36
Affectation de valeurs	36
<i>typedef</i>	37
<i>short</i> ou <i>long</i> : que choisir ?	38
<i>Propriétés des entiers non signés</i>	39
<i>Propriétés d'un entier signé</i>	40
Constantes	41
<i>Constantes littérales</i>	41
<i>Constantes symboliques</i>	41
<i>Constantes énumérées</i>	43
Récapitulatif	43
Questions-réponses	44
Chapitre 4. Expressions et instructions	45
Instructions	45
<i>Espaces blancs</i>	46
<i>Blocs d'instructions</i>	46
Expressions	47
Opérateurs	48
<i>Opérateur d'affectation</i>	48
<i>Opérateurs mathématiques</i>	49

Combiner des opérateurs d'affectation et des opérateurs mathématiques	50
Incrémentation et décrémentation	50
<i>Préfixe et suffixe</i>	51
Priorités	53
Parenthèses imbriquées	54
Vrai ou faux	54
Opérateurs relationnels	55
L'instruction <i>if</i>	56
<i>La clause else</i>	56
<i>Instructions if avancées</i>	58
Les accolades	60
Opérateurs logiques	62
<i>L'opérateur logique And</i>	62
<i>L'opérateur logique OR</i>	63
<i>L'opérateur logique NOT</i>	63
Précisions sur les évaluations logiques	63
Précisions sur TRUE et FALSE	64
Récapitulatif	65
Questions-réponses	65
Chapitre 5. Fonctions	67
Définition	67
Déclaration et définition de fonctions	68
<i>Déclaration d'une fonction</i>	68
<i>Définition de la fonction</i>	71
Variables locales	73
Variables globales	76
Instructions de fonction	76
Arguments de fonction	76
<i>Utilisation de valeurs de renvoi</i>	76
Les paramètres sont des variables locales	77
Valeurs de renvoi	79
Paramètres par défaut	81
Surcharge de fonctions	84
<i>Mode de fonctionnement des fonctions</i>	85
<i>La pile et les fonctions</i>	87
Récapitulatif	87
Questions-réponses	88

PARTIE 2. CLASSES

Chapitre 6. Classes de base	91
Création de types	91
<i>Type : définition</i>	91
<i>Pourquoi créer un type ?</i>	92
Classes et membres	92
<i>Déclarer une classe</i>	93
<i>Précisions sur les conventions d'attribution de noms</i>	94
<i>Définition d'un objet</i>	95
<i>Classes et objets</i>	95
Accès aux membres d'une classe	95
<i>Affectation à des objets</i>	95
Private ou Public ?	96
<i>Données membres privées</i>	97
Mise en œuvre des méthodes de classe	99
Constructeurs et destructeurs	101
<i>Constructeurs par défaut</i>	102
<i>Constructeurs fournis par le compilateur</i>	103
Récapitulatif	106
Questions-réponses	106
Chapitre 7. Concepts avancés sur les classes	107
Fonctions membres const	107
Y a-t-il incompatibilité entre interface et mise en œuvre du programme ? ..	108
Où figurer les déclarations de classes et les définitions de méthodes ?	109
<i>Emplacement des déclarations de classes</i>	110
Implémentation inline	111
Classes imbriquées	113
Récapitulatif	118
Questions-réponses	118
Chapitre 8. Déroulement d'un programme	121
Les itérations	121
<i>Origines de l'instruction goto</i>	121
<i>Pourquoi jeter l'instruction goto aux oubliettes ?</i>	122
Boucles while	123
<i>Instructions while complexes</i>	124
<i>Instructions continues et break</i>	126
<i>La boucle while(1)</i>	128

Les boucles <i>do...while</i>	130
Les boucles <i>for</i>	132
<i>Boucles for évoluées...</i>	134
<i>Boucles for vides</i>	138
<i>Boucles imbriquées</i>	139
Instructions <i>switch</i>	140
Récapitulatif	143
Questions-réponses	143

PARTIE 3. GESTION DE LA MÉMOIRE

Chapitre 9. Pointeurs	147
Définition	147
<i>Stockage d'une adresse dans un pointeur</i>	150
<i>Nom de pointeur</i>	151
<i>L'opérateur d'adressage indirect</i>	151
<i>Pointeurs, adresses et variables</i>	153
<i>Pointeurs et gestion de données</i>	153
<i>Lecture d'une adresse</i>	155
Utilité des pointeurs	157
La pile et l'espace mémoire disponible	158
<i>Le mot clé new</i>	159
<i>Le mot clé delete</i>	160
<i>Défaillances de la mémoire</i>	162
Récapitulatif	163
Questions-réponses	164
Chapitre 10. Concepts avancés sur les pointeurs	165
Création d'objets dans le tas	165
Suppression d'objets	165
Lecture de données membres	167
Construction et destruction de données membres dans le tas	168
Le pointeur <i>this</i>	170
Utilité du pointeur <i>this</i>	172
Pointeurs incontrôlables	172
Pointeurs constants	173
<i>Pointeurs constants et fonctions membres constantes</i>	174
<i>Pointeurs const this</i>	176
Récapitulatif	176
Questions-réponses	176

Chapitre 11. Références	177
Définition	177
Création d'une référence	177
Extraction de l'adresse d'une référence	179
Quels objets peut-on référencer ?	182
Pointeurs nuls et références nulles	183
Transmettre des arguments de fonction par référence	183
<i>Passer des pointeurs à la fonction swap()</i>	185
<i>Implémentation de références dans la fonction swap()</i>	187
Atouts des en-têtes et des prototypes de fonctions	189
Renvoi de valeurs multiples	189
<i>Renvoi de valeurs par référence</i>	191
Récapitulatif	193
Questions-réponses	193
Chapitre 12. Concepts avancés sur les références et les pointeurs	195
Performances du passage de paramètres par référence	195
<i>Passer un pointeur constant</i>	199
Une autre solution : le passage de références à des objets	202
Choisir entre références et pointeurs	204
Une cause fréquente d'erreur	205
Renvoi d'une référence à un objet du tas	207
Où est passé le pointeur ?	209
Récapitulatif	210
Questions-réponses	210

PARTIE 4. OUTILS ÉVOLUÉS

Chapitre 13. Fonctions avancées	215
Fonctions membres surchargées	215
Utilisation de valeurs par défaut	218
Choisir entre des valeurs par défaut et des fonctions surchargées	221
Le constructeur par défaut	221
Surcharge de constructeurs	221
Initialisation d'objets	222
Le constructeur de copie	223
Récapitulatif	228
Questions-réponses	229

Chapitre 14. Surcharge d'opérateurs	231
Surcharge d'opérateurs	231
<i>Ecrire une fonction d'incrémentat</i> ion	232
<i>Surcharge de l'opérateur suffixe</i>	234
<i>Différence entre préfixe et suffixe</i>	234
<i>Opérateur plus (+)</i>	236
<i>Surcharge de l'opérateur (+)</i>	238
<i>Restrictions</i>	240
<i>Savoir utiliser la surcharge</i>	240
<i>Opérateur (=)</i>	241
Opérateurs de conversion	243
<i>L'opérateur unsigned short()</i>	246
Récapitulatif	248
Questions-réponses	248
 Chapitre 15. Tableaux	 249
Définition	249
Eléments de tableau	250
Ecriture d'éléments hors limites	251
Erreurs de copie hors de la limite supérieure d'un tableau	252
Initialisation de tableaux	253
Tableaux d'objets	254
Tableaux à plusieurs dimensions	255
<i>Initialisation de tableaux à plusieurs dimensions</i>	257
Les tableaux et la mémoire	259
Tableaux de pointeurs	259
Déclaration de tableaux dans le tas	261
Pointeur de tableau et tableau de pointeurs	262
Pointeurs et noms de tableaux	262
Suppression de tableaux dans le tas	264
Tableaux char	265
strcpy() et strncpy()	268
Classes String	269
Récapitulatif	270
Questions-réponses	271

PARTIE 5. HÉRITAGE ET POLYMORPHISME

Chapitre 16. Héritage	275
Définition	275
<i>Héritage et dérivation</i>	276
<i>Le règne animal</i>	277
<i>Syntaxe de la dérivation</i>	277
Membres de classe : privés ou protégés ?	280
Constructeurs et destructeurs	282
<i>Passer des arguments aux constructeurs de base</i>	285
Substitution de fonctions	290
<i>Surcharge et substitution</i>	292
<i>Masquage de la méthode de la classe de base</i>	293
<i>Appeler la méthode de base</i>	294
Récapitulatif	296
Questions-réponses	297
Chapitre 17. Polymorphisme et classes dérivées	299
Méthodes virtuelles	299
<i>Mode opératoire des fonctions membres virtuelles</i>	304
<i>Tous les chemins ne mènent pas aux objets !</i>	306
<i>Slicing</i>	306
<i>Destructeurs virtuels</i>	309
<i>Constructeurs de copie virtuels</i>	309
<i>Avantages et inconvénients des méthodes virtuelles</i>	313
Récapitulatif	313
Questions-réponses	314
Chapitre 18. Concepts avancés sur le polymorphisme	315
Problèmes de l'héritage simple	315
Types de données abstraits (ADT)	320
<i>Fonctions virtuelles pures</i>	324
<i>Hiérarchies complexes d'abstraction</i>	331
<i>Utilité des ADT</i>	336
Récapitulatif	336
Questions-réponses	337
Chapitre 19. Listes chaînées	339
Listes chaînées et autres structures	339
Etude de cas	341
<i>Transfert de responsabilité</i>	341

Listes, nœuds et arborescence...	341
Bilan de l'exercice	351
Récapitulatif	351
Questions-réponses	352
PARTIE 6. FONCTIONNALITÉS SPÉCIALES	
Chapitre 20. Classes et fonctions spéciales	355
Données membres statiques	355
Fonctions membres statiques	358
Imbrication d'objets	360
<i>Accéder aux membres d'une classe imbriquée</i>	367
<i>Limiter les accès aux membres imbriqués</i>	367
<i>Avantages et inconvénients de l'imbrication</i>	368
<i>Copier des objets : par valeur ou par référence ?</i>	368
Classes amies	368
Fonctions amies	369
Pointeurs de fonctions	369
<i>Notation</i>	373
<i>Tableaux de pointeurs de fonctions</i>	373
<i>Passer des pointeurs de fonctions à d'autres fonctions</i>	375
<i>Utilisation de typedef avec des pointeurs de fonctions</i>	378
Pointeurs de fonctions membres	381
<i>Tableaux de pointeurs de fonctions membres</i>	384
Récapitulatif	386
Questions-réponses	387
Chapitre 21. Le préprocesseur	389
Le préprocesseur et le compilateur	389
Consultation du fichier intermédiaire	389
La commande #define	390
<i>Utilisation de #define pour les constantes</i>	390
<i>Utilisation de #define lors du débogage</i>	390
<i>La commande #else</i>	391
L'inclusion : mode d'emploi	392
<i>Définition à partir de la ligne de commande</i>	394
<i>Annulation d'une définition</i>	394
<i>Compilation conditionnelle</i>	394

Fonctions macros	394
<i>Pourquoi toutes ces parenthèses ?</i>	396
<i>Les macros face aux fonctions et modèles</i>	397
Gestion de chaînes	398
<i>L'opérateur #</i>	398
<i>Concaténation</i>	398
Macros prédéfinies	399
<i>assert()</i>	400
Récapitulatif	414
Questions-réponses	415

PARTIE 7. PROGRAMMATION AVANCÉE

Chapitre 22. Analyse et conception orientées objet	419
Le cycle de développement	419
Simulation d'un système d'alarme	420
<i>Conceptualisation</i>	421
<i>Analyse des besoins</i>	421
<i>Analyse conceptuelle et analyse organique</i>	421
<i>Autres objets</i>	422
<i>Classes</i>	423
<i>Gestion des alarmes</i>	424
<i>Boucles de gestion d'événements</i>	424
Etude de cas : le projet PostMaster	427
<i>Contraintes et réalité</i>	428
<i>Diviser pour mieux régner</i>	428
<i>Format des messages</i>	429
<i>Conception initiale des classes</i>	430
<i>La hiérarchie d'héritage : avec ou sans racine ?</i>	431
<i>Conception des interfaces</i>	433
<i>Construire un prototype</i>	435
<i>La règle des 80/80</i>	435
<i>Concevoir la classe PostMasterMessage</i>	436
<i>Interface de programmation (API)</i>	436
<i>Cohésion des groupes</i>	438
<i>Attribuer les fonctionnalités</i>	438
<i>Développer un gestionnaire</i>	439
Récapitulatif	447
Questions-réponses	447

Chapitre 23. Modèles ou templates	449
Définition	449
Types paramétrés	450
Définition d'un modèle	450
Utilisation de modèles	459
La bibliothèques de modèles standards	468
Récapitulatif	468
Questions-réponses	469

II. BORLAND C++ 4.0

Avant-propos	473
Chapitre 1. Pour commencer	475
Installation de Borland C++	475
<i>Configuration matérielle et logicielle requise</i>	475
<i>Procédure d'installation</i>	476
<i>Lancement de Borland C++ 4.0 sous Windows 3.1</i>	482
<i>Lancement de Borland C++ 4.0 sous Windows 95/98</i>	482
<i>Comment activer l'aide</i>	484
Configuration de l'EDI	485
<i>Modification des barres d'icônes</i>	485
<i>Choix de préférences de l'EDI</i>	487
<i>Enregistrement des paramètres de l'EDI</i>	487
Utilisation de l'Editeur	488
<i>Configuration de l'éditeur de l'EDI</i>	488
<i>Mise en surbrillance de la syntaxe</i>	489
La fenêtre de messages	490
Parcourir votre code	491
<i>Visualisation des objets (vue générale sur les classes)</i>	492
<i>Déplacement dans les symboles globaux</i>	494
<i>Visualisation des symboles dans votre code</i>	496
Utilisation des outils en ligne de commande	496
<i>DPMI et les outils en ligne de commande</i>	497
<i>Mémoire vive et MAKESWAP.EXE</i>	497
<i>Gestionnaire de run-times et outils en ligne de commande</i>	498
Lancement d'autres programmes depuis l'EDI	499



Chapitre 2. Utilisation du gestionnaire de projet	501
Qu'est-ce que la gestion de projets ?	501
Construction d'un projet	503
<i>Création d'un projet multi-cible</i>	505
<i>Conversion des anciens projets</i>	506
<i>Conversion de projets en fichiers MAKE</i>	506
<i>Modification de la visualisation du projet</i>	506
Création d'un projet	508
<i>Construction d'une partie de projet</i>	508
Modification de l'arborescence du projet	509
<i>Modification des attributs de cible avec TargetExpert</i>	509
<i>Modification des attributs de nœud</i>	510
<i>Ajout et suppression de nœuds</i>	511
<i>Ajout et suppression de cibles</i>	512
<i>Déplacement des nœuds et des cibles</i>	512
<i>Copie de nœuds</i>	513
Utilisation des Source Pools	514
Définition des options d'un projet	515
<i>Dérogeration locale</i>	515
<i>Utilisation des feuilles de style</i>	516
<i>Visualisation des options d'un projet</i>	520
Traducteurs	520
<i>Installation d'un traducteur</i>	521
<i>Utilisation de Special dans le TurboMenu</i>	522
<i>Installation de visualiseurs et d'outils</i>	523
Chapitre 3. Compilation	525
Compilation dans l'EDI	525
<i>Utilisation des options de compilation dans l'EDI</i>	526
Utilisation des compilateurs en ligne de commande	527
<i>Fichiers de configuration</i>	528
<i>Fichiers de recherche</i>	529
<i>Règles de priorité des options</i>	529
Options de compilation	529
Répertoires	546
<i>Algorithmes de recherche de fichiers</i>	548
Compiler Defines	549
Compiler Code-generation	550
Compiler Floating Point	551
Compiler Compiler Output	552

Compiler Source	553
Compiler Debugging	554
Compiler Precompiled headers	555
16-bit Compiler Processor	556
16-bit Compiler Calling Convention	557
16-bit Compiler Memory Model	558
16-bit Compiler Segment Names Data	561
16-bit Compiler Segment Names Far Data	562
16-bit Compiler Segment Names Code	563
16-bit Compiler Entry/Exit Code	563
32-bit Compiler Processor	566
32-bit Compiler Calling Convention	566
C++ Options Member Pointer	567
C++ Options C++ Compatibility	568
C++ Options Virtual Tables	570
C++ Options Templates	571
C++ Options Exception handling/RTTI	571
Optimisations	572
Optimizations Specific	572
Optimizations Size	574
Optimizations Speed	576
Messages	578
Messages Portability	579
Messages ANSI Violations	580
Messages Obsolete C++	581
Messages Potential C++ Errors	581
Messages Inefficient C++ Coding	582
Messages Potential errors	583
Messages Inefficient Coding	583
Messages General	584
Make	584
Options en ligne de commande	585
Chapitre 4. Débogage dans l'EDI	591
Types de bugs	591
<i>Erreurs à la compilation</i>	591
<i>Erreurs à l'exécution</i>	592
<i>Erreurs de logique</i>	592
Création d'informations de débogage	593
Indication des arguments du programme	593

Contrôle de l'exécution du programme	594
<i>Vérification de la sortie générée par le programme</i>	594
<i>Mode pas à pas</i>	594
<i>Tracé du code en mode approfondi</i>	596
<i>Contourner une partie du code</i>	596
<i>Arrêt du programme</i>	598
<i>Recommencer depuis le début</i>	598
Visualisation des valeurs	598
<i>Qu'est-ce qu'une expression ?</i>	599
<i>Visualisation des expressions</i>	599
<i>Evaluation et modification des expressions</i>	601
<i>Visualisation des éléments de données</i>	604
<i>Examen des valeurs de registres</i>	605
Points d'arrêt	606
<i>Pose de points d'arrêt</i>	606
<i>Utilisation des points d'arrêt</i>	607
<i>Personnalisation des points d'arrêt et du point d'exécution</i>	610
Gestion des erreurs GPF	611
Utilisation de la fenêtre Event Log	611
Débogage des DLL	612
Débogage en mode soft et en mode hard	613
Chapitre 5. Présentation de Resource Workshop	615
Ressources Windows	615
Types de fichiers ressources	617
<i>Fichiers ressources bitmap</i>	619
Chapitre 6. Projets, ressources et identificateurs	621
Création d'un projet	621
Ouverture d'un projet existant	622
Comment utiliser la fenêtre projet	623
<i>Ressources intégrées ou liées</i>	623
<i>Affichage d'informations dans la fenêtre projet</i>	624
<i>Sélection d'une ressource</i>	625
Comment travailler avec les ressources	625
<i>Chargement d'une ressource</i>	625
<i>Ajout d'une ressource</i>	627
<i>Comment déplacer une ressource</i>	629
<i>Comment copier une ressource entre projets</i>	630
<i>Comment supprimer une ressource</i>	631

<i>Comment renommer une ressource</i>	631
<i>Spécification des options de mémoire ressource</i>	631
Comment utiliser les identificateurs	633
<i>Composants des identificateurs de ressource</i>	633
<i>Fichiers identificateurs</i>	634
<i>Création des fichiers identificateurs</i>	635
<i>Fichiers en-tête C</i>	635
<i>Gestion automatique des identificateurs</i>	636
<i>Comment travailler sans fichier d'identificateurs</i>	637
<i>Ajout d'identificateurs</i>	637
<i>A partir de la boîte de dialogue Identifiers</i>	638
<i>Edition d'identificateurs</i>	639
<i>Suppression d'identificateurs</i>	639
<i>Liste des identificateurs</i>	640
<i>Lancement d'un éditeur de ressources</i>	640
Préférences de configuration	641
<i>Text Editor</i>	641
<i>Multi-Save</i>	642
<i>Target Windows Version</i>	642
Comment travailler avec des fichiers binaires	643
Comment créer des ressources 32 bits	644
Chapitre 7. Comment utiliser l'éditeur Bitmap	647
Lancement de l'éditeur Bitmap	647
Couleurs de premier plan et de fond, et pixels	648
Utilisation de la palette Tools (Outils)	649
<i>Outil Rectangle de sélection</i>	650
<i>Outil Ciseaux</i>	650
<i>Outil Zoom</i>	651
<i>Outil Gomme</i>	652
<i>Outil Stylo</i>	652
<i>Outil Pinceau</i>	653
<i>Outil Aérographe</i>	653
<i>Outil Boîte de peinture</i>	654
<i>Outil Ligne</i>	654
<i>Outil Texte</i>	655
<i>Dessin de cadres vides</i>	656
<i>Dessin de cadres pleins</i>	656
<i>Outil Main</i>	657
<i>Sélections de style</i>	657



Utilisation des deux volets de la fenêtre	657
Lecture de la ligne d'état	658
Comment travailler avec les couleurs	659
<i>Choix du nombre de couleurs d'une ressource</i>	660
<i>Utilisation des couleurs de premier plan et de fond</i>	660
<i>Zones couleur transparentes et inversées</i>	661
<i>Comment afficher et cacher la palette Colors</i>	663
Personnalisation des couleurs	663
<i>Edition de couleurs dans la palette Colors</i>	663
Ajout de texte à une ressource	665
<i>Alignement du texte</i>	666
<i>Choix des polices, de la taille et du style de texte</i>	666
Choix des formes de pinceaux	667
Choix des motifs de dessin	668
Choix d'un style de ligne	668
Alignement d'une zone sélectionnée	669
Redimensionnement d'une zone sélectionnée	670
Choix des options globales de l'éditeur Bitmap	671
Index	673