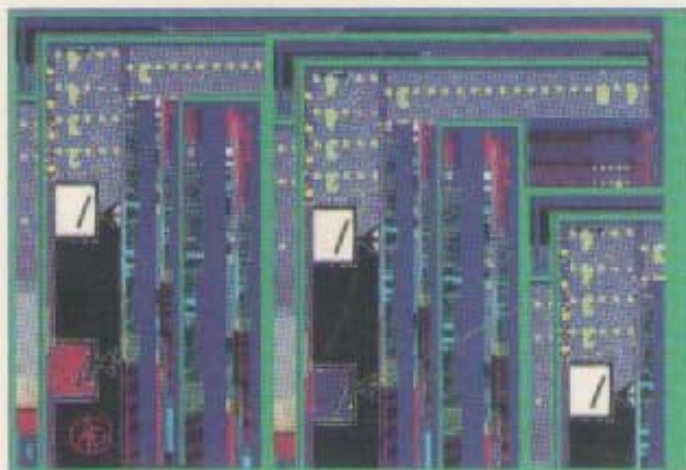


Michel Divay

2^e CYCLE • IUT • ÉCOLES D'INGÉNIEURS

Unix et les systèmes d'exploitation

Cours et exercices corrigés



DUNOD

2-005-139-1

2-005-139-1

المكتبة ٢٠٥٧

Unix et les systèmes d'exploitation

Cours et exercices corrigés

Michel Divay

Professeur à l'université de Rennes 1

DUNOD



Table des matières

AVANT-PROPOS	VII
CHAPITRE 1 • UNIX	
1.1 Présentation générale d'un SE	1
1.2 Principe de la construction d'un SE	2
1.3 Le cas de MS-DOS	3
1.4 Historique, présentation générale d'Unix	5
1.5 Les commandes générales d'Unix	8
1.5.1 Généralités sur les commandes Unix	8
1.5.2 Systèmes de fichiers arborescents	8
1.5.3 Caractères génériques (métacaractères)	10
1.5.4 Compléments de noms	11
1.5.5 Droits d'accès	11
1.5.6 Commandes de base Unix	13
1.5.7 Redirections, tubes	34
1.6 Le Shell	38
1.6.1 Les variables	39
1.6.2 Résultats d'une commande	41
1.6.3 Les expressions arithmétiques	42
1.6.4 Les scripts shell	43
1.6.5 Tableau, alias	56
1.7 Résumé	62
CHAPITRE 2 • LES FILTRES UNIX	
2.1 Paramètres d'une commande	63
2.1.1 Options de la ligne de commande	63
2.1.2 Variables d'environnement en langage C	65
2.1.3 Exemple : la nouvelle commande eko	66

2.2	Les filtres en C	69
2.2.1	Définition	69
2.2.2	Exemples de filtres écrits en C	70
2.3	Quelques filtres standard Unix	78
2.3.1	grep et les expressions régulières	78
2.3.2	Filtres tr, cut, colrm, sort, paste, uniq	80
2.3.3	Filtres nl, pr	83
2.3.4	Filtre sed (stream editor)	84
2.4	Résumé	84
CHAPITRE 3 • LE SGF UNIX		
3.1	Les différents types de fichiers	85
3.2	Les répertoires du système	87
3.3	Les différents types de systèmes de fichiers	89
3.4	Structure générale d'un sgf	89
3.4.1	Inode	89
3.4.2	Implantation des données d'un fichier	90
3.5	Répertoire	93
3.6	Les tables d'occupation	95
3.7	Le découpage en groupes	96
3.8	Répertoires et sous-répertoires	98
3.9	Système de fichiers de type ext2	100
3.9.1	Système de fichiers (vide) après mkfs	101
3.9.2	Création de fichiers et répertoires	104
3.10	Variantes pour la gestion des inodes et des blocs libres	105
3.10.1	Allocation, désallocation d'inodes (variante)	106
3.10.2	Allocation, désallocation de blocs de données (variante)	107
3.11	Définition des appels système SGF	108
3.12	Primitives d'entrées-sorties sous Unix	109
3.12.1	Description des appels système concernant les entrées-sorties	112
3.12.2	Exemple de programme avec appels système du SGF	124
3.12.3	fcntl : contrôle des caractéristiques d'un fichier ouvert	131
3.12.4	Verrouillage de fichiers	132
3.13	Bibliothèque d'entrées-sorties	143
3.13.1	Principe	143
3.13.2	La bibliothèque stdio.h	143
3.13.3	Exemple de création d'une bibliothèque d'entrées-sorties	146
3.14	Le buffer cache (entrées-sorties en mode bloc)	152
3.15	Buffer cache (Linux)	159
3.16	Date et heure	162
3.17	Résumé	163

CHAPITRE 4 • LES APPELS SYSTÈME POUR LES PROCESSUS

4.1 Définitions	165
4.2 Les processus	166
4.3 Les appels système (processus)	167
4.3.1 Caractéristiques d'un processus	167
4.3.2 L'appel système pipe (tube anonyme)	169
4.3.3 L'appel système execve et les fonctions de la famille exec	170
4.3.4 L'appel système fork de création de processus	173
4.3.5 Autres appels système sur les processus	174
4.4 Utilisation de fork, exec, pipe, dup2	175
4.4.1 Exécution de deux processus séquentiels indépendants	175
4.4.2 Redirection des entrées et des sorties sur un fichier	176
4.4.3 Redirection des entrées et des sorties sur un tube	179
4.5 Interpréteur de commandes	184
4.6 Résumé	190

CHAPITRE 5 • PROCESSUS ET SÉMAPHORES

5.1 Introduction	191
5.2 Terminologie	191
5.3 Exclusion mutuelle	192
5.3.1 Problème : pourquoi ça ne marche pas ?	192
5.3.2 Exemple de principe de l'exclusion mutuelle	193
5.4 Exclusion mutuelle et sémaphores	194
5.4.1 Définitions	194
5.4.2 Réalisation d'opérations P et V avec des tubes (pipe Unix)	194
5.4.3 Principe de l'exclusion mutuelle avec sémaphore	196
5.4.4 Exemple du carrefour routier	196
5.4.5 Changement d'états d'un processus	199
5.4.6 Exemple des baigneurs	200
5.5 Synchronisation avec des sémaphores	208
5.5.1 Définition	208
5.5.2 Exemple des moyens de transport	209
5.5.3 Exemple du producteur-consommateur	212
5.6 Interblocage (deadlock)	215
5.7 Résumé	216

CHAPITRE 6 • COMMUNICATION ENTRE PROCESSUS

6.1 Communiquer : du simple au complexe	217
6.2 Les signaux	218
6.3 Principe du client-serveur	228
6.4 Les tubes nommés ou FIFO	229
6.4.1 Exemple de communication par tube nommé	230
6.4.2 Exemple des baigneurs	232
6.4.3 Client-serveur avec des tubes nommés	233

6.5	Les files de messages	234
6.5.1	Définitions	234
6.5.2	Exemple : menu de test des files de messages	235
6.6	Client serveur avec files de messages	240
6.6.1	Le principe	240
6.6.2	Les clients : étudiant, enseignant	241
6.6.3	Communication par file de messages	246
6.7	Les sémaphores Unix	250
6.7.1	Définitions	250
6.7.2	P et V avec les sémaphores Unix	251
6.8	Les segments de mémoire partagée	256
6.8.1	Définitions	256
6.8.2	Producteur-consommateur (tampon 1 place)	257
6.8.3	Producteur-consommateur (tampon N places)	259
6.9	Synchronisation et mémoire partagée : les 5 philosophes	261
6.9.1	Première solution : fausse ; il peut y avoir interblocage	261
6.9.2	La solution correcte	262
6.10	Application avec mémoire partagée et sémaphores : bataille navale	266
6.11	Résumé	271
CHAPITRE 7 • GESTION DE LA MÉMOIRE ET ORDONNANCEMENT		
7.1	Relocation des programmes	273
7.2	Gestion de la mémoire : les variantes	274
7.2.1	Un seul utilisateur en mémoire en plus du SE	274
7.2.2	Plusieurs utilisateurs plus le SE en mémoire centrale	274
7.2.3	Segmentation	276
7.2.4	Pagination et mémoire virtuelle	277
7.3	Exemple : Linux	280
7.3.1	Adresses virtuelles	280
7.3.2	Adresses réelles	283
7.4	Ordonnancement (scheduling)	298
7.4.1	Mémorisation des processus	298
7.4.2	État d'un processus, files d'attente	301
7.4.3	Scheduler et priorités	301
7.4.4	Arbre des processus	303
7.5	Résumé	304
7.6	Conclusion générale	304
CORRIGÉS DES EXERCICES		305
BIBLIOGRAPHIE		339
INDEX		341