

monographies
de l'AFCET

Richard EHRMANN

l'allocation dynamique
de la mémoire
des ordinateurs

BIBLIOTHEQUE DU CERIST

15T
214

dunod

DANS LA MÊME COLLECTION

- R. FORTET, J. ABADIE, J. BERNADAT, M. COURTILLOT, J.-M. GAUTHIER, F. GENUYS, P. HUARD et G. MATTHYS : *Mathématiques des programmes économiques*.
- X D. CARRÉ, P. DARNAUT, PH. GUITART, PH. TUAN NGHIEM, P. PACAUD, J. DE ROSINSKI, B. ROY et G. SANDIER : *Les problèmes d'ordonnement* (applications et méthodes).
- B. ROY : *Aléas numériques et distributions de probabilités usuelles* (généralités sur les aléas numériques).
- P. BROISE : *Le langage Algol* (applications à des problèmes de recherche opérationnelle).
- G. DEBREU : *Théorie de la valeur*.
- X J. BERTIN, M. RITOUT et J.-C. ROUGIER : *L'exploitation partagée des calculateurs*.
- X L. BOLLIET : *Utilisation des ordinateurs à distance en temps réel et en temps partagé*.
- X J. STENGEL : *Les systèmes informatiques de programmation économique*.
- X P. BROISE, P. HUARD et J. SENTENAC : *Décomposition des programmes mathématiques*.
- J. AGARD, J. ALTABER, R. FORTET, A. KAUFMANN, P. LE GALL, M. PRÉCIGOUT, J. STENGEL et G. THOMAS : *Les méthodes de simulation*.
- G. TH. GUILBAUD : *Statistique des chroniques*.
- G. TH. GUILBAUD : *Éléments de la théorie mathématique des jeux*.
- Y. CHERRUAULT : *Approximation d'opérateurs linéaires et applications*.
- X D. RIBBENS : *Programmation non numérique LISP 15*.
- G. MATTHYS : *Recherche opérationnelle et planification nationale*.
- X L. BOLLIET : *Les systèmes conversationnels*.
- Groupe de spécialistes dirigé par J. AGARD : *Rationalisation des choix budgétaires*.
- J. C. DERNIAME et C. PAIR : *Problèmes de cheminement dans les graphes*.
- X J. PITRAT : *Un programme de démonstration de théorèmes*.
- G. MORLAT et F. BESSIÈRE : *Vingt-cinq ans d'économie électrique*.
- X G. LOUÏT : *Algorithmes de tri*.
- X R. BENAYOUN et C. BOULIER : *Approches rationnelles dans la gestion du personnel*.
- R. EHRMANN : *L'allocation dynamique de la mémoire des ordinateurs*.
- Textes rassemblés par Ph. BILLOT et J. MATHA : *Rationalisation des choix et urbanisme*.

055 111

871

MONOGRAPHIES DE L'AF CET

ASSOCIATION FRANÇAISE
POUR LA CYBERNÉTIQUE ÉCONOMIQUE
ET TECHNIQUE

551

BIBLIOTHEQUE DU CERIST

**L'ALLOCATION
DYNAMIQUE
DE LA MÉMOIRE
DES ORDINATEURS**

Richard EHRMANN

Ancien élève de l'École polytechnique

DUNOD
PARIS
1972

151 214

TABLE DES MATIÈRES

Objet de l'ouvrage	1
<i>Avant-propos</i>	1
<i>Allocation statique</i>	1
<i>Allocation dynamique</i>	2
Généralités sur les chaînes	5
<i>Représentation des chaînes en machine</i>	5
<i>Modification des chaînes</i>	6
Retirer	6
Ajouter	7
<i>Utilisation des chaînes</i>	7
Quelques exemples de chaînes	9
<i>La station d'autobus</i>	9
Le laisser-faire	9
La chicane	9
Le ticket d'appel	10
<i>La queue chez le marchand de glaces</i>	10
<i>La pile</i>	10
Cheminement sur une arborescence	11
Ordre arbitraire	13
<i>Un abus de langage</i>	13
L'allocation dynamique et le chaînage	15
<i>Le fonds commun</i>	15
<i>Les arcanes de l'allocation dynamique</i>	16
Utilisation de SIMSCRIPT dans cet ouvrage	17
<i>Déclarations</i>	17
<i>Instructions</i>	18
Do	18
Remove	18
Remove first	19
File	20
<i>Les requêtes aux procédures d'allocation dynamique</i>	23

Les critères de qualité	25
<i>Le minimum de temps</i>	25
<i>Le maximum de choses</i>	26
<i>Le volume mémoire</i>	26
<i>Indépendance du critère mémoire</i>	27
<i>Synthèse</i>	28
Le cas des fiches de longueur fixe	29
<i>La pagination</i>	30
<i>Le fonds commun SIMSCRIPT</i>	30
Les stratégies SPEC et REG du système AED	33
<i>Terminologie</i>	33
<i>La stratégie SPEC</i>	34
<i>La stratégie REG</i>	35
Première approche	35
Réalisation complète	37
Les procédures d'assistance	41
<i>Fonds débordé</i>	41
<i>Gestion extérieure des zones</i>	42
Multiplicité des zones	42
Hypothèse de longueur fixe	43
Hypothèse de longueur identique	44
<i>Quelques orientations de l'assistance</i>	44
Etendre la zone requise	44
Créer une zone	45
Concéder la requête à une autre zone	45
Chercher une molécule trop grande et l'affecter telle quelle	45
Chercher une molécule trop grande et la scinder	46
Tenter d'utiliser des molécules trop petites	46
La collecte des résidus	47
<i>Le compactage</i>	48
<i>Le rappel des égarés</i>	49
<i>La concaténation de molécules contiguës : GARB</i>	49
Appréciation	51
Le modèle PARENTE	53
Les stratégies ouvertes	55
L'intendance	59
<i>L'initialisation</i>	59
<i>La répartition</i>	60
Le refus de fournir	61
L'excès de fourniture	61

Table des matières

IX

Motivation des stratégies	63
<i>SPEC</i>	63
La connaissance <i>a priori</i> des longueurs	63
L'initialisation	63
La répartition	63
L'assistance	63
<i>REG</i>	64
Inconvénients	64
<i>Stratégie ouverte</i>	65
 La protection mémoire	 67
<i>Accroissement quantitatif du risque</i>	67
<i>Accroissement qualitatif du risque</i>	67
Répercussion sur un programme étranger	68
Note sur la pagination	68
Répercussion sur le programme d'allocation dynamique	70
Notre suggestion	71
 La segmentation	 73
<i>Requêtes multiples</i>	73
Nier le problème	73
Allouer pas à pas	74
Simuler l'allocation	75
Remarques	76
<i>Longueur des segments</i>	76
<i>Protection</i>	77
<i>Requêtes groupées</i>	78
Groupement minimum	78
Groupement élaboré	78
 Tests par simulation	 81
<i>Modèle</i>	81
<i>Dossier de programmation</i>	84
Les sous-programmes	84
Structure des fiches	85
Chaînes	86
Symboles	86
<i>Programme</i>	87
 Quelques modèles	 91
<i>Pagination</i>	91
<i>SIMSCRIPT</i>	93
La procédure CNV	94
La procédure CRT	95

Table des matières

La procédure DSTR 96
 Les procédures essentielles 97
 Emploi de remove 98
 Trois rubriques en un mot 99
Le modèle GPSS 100
Le modèle PARENTE 103
Notre suggestion 108

Références 113

Index alphabétique 115

OBJET DE L'OUVRAGE

AVANT-PROPOS

Nous supposons le lecteur familiarisé avec les ordinateurs et surtout avec la programmation, quitte à lui en présenter quelques aspects insolites.

L'unité centrale d'un ordinateur comporte essentiellement une unité de calcul et une mémoire.

La mémoire se divise, en premier lieu, en *mémoire centrale* rapide et mémoire de réserve. La mémoire de réserve (disques, bandes, cartes, etc.) ne fait pas partie de l'unité centrale. Nous n'avons en vue, dans cet ouvrage, que la mémoire centrale. Cependant les considérations développées ici, trouveront aussi des applications dans la gestion des mémoires de réserve à accès direct (disques par exemple).

La mémoire centrale se subdivise en une mémoire centrale proprement dite et une mémoire transitoire (« accumulateurs », « registres », « anté-mémoire », ...) dont les propriétés physiques peuvent être notablement différentes. Ces possibilités dépendent du matériel et ne seront pas envisagées ici. Nous étudierons exclusivement l'aspect fonctionnel de l'allocation dynamique, valable quelle que soit la machine, concernant donc la seule mémoire centrale proprement dite. Nous présenterons les méthodes sous un aspect *entièrement programmé* qui devra nécessairement être adapté dans chaque cas particulier.

ALLOCATION STATIQUE

L'exécution d'un calcul par l'ordinateur, suppose qu'une partie de la mémoire soit affectée à chacun des éléments du calcul. Par « élément du calcul », nous entendons aussi bien les opérateurs que les variables.

Cette affectation se réalise habituellement avant le calcul proprement dit. Nous supposerons résolus tous les problèmes afférents à une affectation statique, y compris le fait que, si un calcul requiert une zone mémoire de dimension supérieure à la dimension physique existante, la « solution » est que le calcul ne peut être exécuté.

L'utilisation des ordinateurs se généralisant et se perfectionnant, on a tenté de traiter tout de même de tels problèmes, si à *chaque instant*, la zone mémoire requise est assez petite. Un cas résolu est celui où il existe une partition du programme (le *programme* représente l'algorithme de calcul), telle que chaque

partie ne requière qu'une quantité admissible de mémoire : la partie active est seule placée en mémoire centrale, le reste étant stocké en mémoire de réserve. Le coût des échanges entre mémoire centrale et réserve, restreint, même dans ce cas, l'utilisation de la méthode. Il faut, notamment, que la suite des parties actives au cours de l'exécution du programme, soit courte. En tout cas, il n'y a pas là d'allocation dynamique de la mémoire : la mémoire est affectée en totalité à des parties successives d'un programme, tout comme — dans un train de moniteur — elle est affectée à des programmes successifs.

ALLOCATION DYNAMIQUE

Les problèmes que nous envisageons, et qui donnent lieu à *allocation dynamique* sont ceux où apparaissent *des requêtes et des restitutions aléatoires* de zone mémoire ; un fonds commun de mémoire est affecté (statiquement) aux besoins de l'allocation dynamique ; chaque requête se traduit par une allocation de zone mémoire, de dimension conforme à la demande, puisée dans le fonds. Cette zone mémoire est agrégée au fonds commun, dès que l'utilisateur ne la retient plus. Par requêtes et restitutions *aléatoires*, nous voulons dire :

- d'abord que celui qui écrit le programme d'allocation, n'est pas en mesure de déterminer quand ce programme sera appelé ;
- ensuite, et accessoirement, que ce programme d'allocation pourra être testé, par rapport à diverses lois de probabilité, auxquelles pourraient obéir les requêtes et restitutions.

Actuellement, nous connaissons deux grandes catégories d'applications, exigeant l'allocation dynamique de la mémoire : la simulation et les systèmes d'ordinateurs admettant un haut degré de simultanéité (*temps réel* et *multi-programmation* notamment). Ces applications sont d'ailleurs étroitement apparentées et diffèrent surtout par leur clientèle : informaticiens ici, et là, utilisateurs quelconques. D'autres applications existent déjà, et l'outil fera peut-être, plus tard, naître le besoin.

En simulation, le modèle suit un très grand nombre d'informations, de longueur variable ; ces informations réclament ou libèrent de la mémoire, quand elles pénètrent ou quittent le système simulé.

Les systèmes d'ordinateurs commandent l'exécution de tâches qui nécessitent de la mémoire ; une tâche achevée libère donc une zone mémoire, qui peut être utilisée pour prendre en compte de nouvelles tâches.

Le temps réel participe de ces deux emplois.

Nous envisagerons les principes généraux utilisés (ou utilisables), pour réaliser l'allocation dynamique. Nous préciserons les sens possibles d'une éventuelle « optimisation ». Nous étudierons et nous donnerons le moyen

d'évaluer (par simulation), quelques réalisations. Nous proposerons aussi une méthode particulière.

Au préalable nous devons définir les outils qui nous serviront à nous expliquer :

- un concept, *les chaînes*,
- un langage, le **SIMSCRIPT**.