Questionnaire

La disparité de l'enseignement de l'informatique en classes préparatoires nous a fait adopter depuis 6 ans des groupes de niveau.

La scolarité fait distribuer un questionnaire pour classer les élèves en groupes de niveau.

Forts, 6 groupes
Moyens, 8 groupes
Moyens Débutants, 4 groupes
Débutants, 2 groupes

Le questionnaire n'est pas long. Remplissez le sérieusement, svp. La note en informatique au concours d'entrée et le matériel interviendront aussi.

Le retourner à la scolarité à la fin de l'amphi, pour constituer les groupes le plus rapidement possible.

L'informatique à l'X

Jean-Jacques.Levy@inria.fr

INRIA – Rocquencourt

tel: 39-63-56-89

Plan

1. L'informatique scientifique
2. Une technologie exponentielle
3. Enseignement de l'informatique à l'X
4. Le Tronc Commun d'informatique
5. Itération / Récursivité

Panthéon

.

AltaVista

AltaVista est le plus grand index du Web:

30 millions de pages trouvées sur 275600 serveurs, et 3 millions d'articles des 14000 *news groups*.

16 millions de requêtes par jour (de semaine)

AltaVista Alpha 500, 256 M0 / 4 G0.
Traffic externe. Envoie des requêtes aux indexeurs.

Index du Web Alpha 8400 5/300, 10 processeurs, 6 G0 / 210 GO.
L'index fait 40 GO. Les requêtes prennent moins d'1s.

Scooter DEC 3000/900 Alpha, 1 GO / 30 GO.
Parcours les pages Web et les envoie à l'indexeur.

Index News Alpha 250 4/266, 196 MO /13 GO.
Très occupé, car beaucoup de mises à jours.

Serveur de News: Alpha 400 4/233, 160 MO / 24 GO.
Plus simple que d'aller voir sur les autres serveurs.

Comment fait-on?? Recherche en table, hachage, mémoire secondaire, etc. François Bourdoncle?

Ariane 501

Le lanceur a commencé à se désintégrer à environ H0 + 39 secondes sous l'effet de charges aérodynamiques dues à un angle d'attaque de plus de 20 degrés.

cet angle d'attaque avait pour origine le braquage en butée des tuyères des moteurs à propergols solides et du moteur principal Vulcain ;

le braquage des tuyères a été commandé par le logiciel du calculateur de bord (OBC) agissant sur la base des données transmises par le système de référence inertielle actif (SRI2). A cet instant, une partie de ces données ne contenait pas des données de vol proprement dites mais affichait un profil de bit spécifique de la panne du calculateur du SRI 2 qui a été interprété comme étant des données de vol ;

la raison pour laquelle le SRI 2 actif n'a pas transmis des données d'attitude correctes tient au fait que l'unité avait déclaré une panne due à une exception logiciel ;

l'OBC n'a pas pu basculer sur le SRI 1 de secours car cette unité avait déjà cessé de fonctionner durant le précédent cycle de données (période de 72 millisecondes) pour la même raison que le SRI 2 ;

l'exception logiciel interne du SRI s'est produite pendant une conversion de données de représentation flottante à 64 bits en valeurs entières à 16 bits. Le nombre en représentation flottante qui a

été converti avait une valeur qui était supérieure à ce que pouvait exprimer un nombre entier à 16 bits. Il en est résulté une erreur d'opérande. Les instructions de conversion de données (en code Ada) n'étaient pas protégées contre le déclenchement d'une erreur d'opérande bien que d'autres conversions de variables comparables présentes à la même place dans le code aient été protégées ;

l'erreur s'est produite dans une partie du logiciel qui n'assure que l'alignement de la plate-forme inertielle à composants liés. Ce module de logiciel calcule des résultats significatifs avant le décollage seulement. Dès que le lanceur décolle, cette fonction n'est plus d'aucune utilité ;

La fonction d'alignement est active pendant 50 secondes après le démarrage du mode vol des SRI qui se produit à H0 - 3 secondes pour Ariane 5. En conséquence, lorsque le décollage a eu lieu, cette fonction se poursuit pendant environ 40 secondes de vol. Cette séquence est une exigence Ariane 4 mais n'est pas demandé sur Ariane 5 ;

l'erreur d'opérande s'est produite sous l'effet d'une valeur élevée non prévue d'un résultat de la fonction d'alignement interne appelé BH (Biais Horizontal) et lié à la vitesse horizontale détectée par la plate-forme. Le calcul de cette valeur sert d'indicateur pour la précision de l'alignement en fonction du temps ;

la valeur BH était nettement plus élevée que la valeur escomptée car la première partie de la trajectoire d'Ariane 5 diffère de celle d'Ariane 4, ce qui se traduit par des valeurs de vitesse horizontale considérablement supérieures.

Ariane 501 – bis

Conclusion 1: (erronée)

écrire le tout en C, qui ne teste pas les débordements d'entiers. Les SRI auraient continuer à envoyer des données fausses et inutiles.

Conclusions 2: (correctes)

faire des spécifications formelles du problème (et pas seulement du programme).

faire les tests exhaustifs correspondants aux spécifications, ou faire une preuve formelle.

redondance logicielle (pour lutter contre les erreurs logiques).

Les logiciels sont des objets complexes

Quelques résultats scientifiques

1. F9 n'est pas premier
   (algorithmes et calcul distribué)

2. 20 ans de Delaunay en 1h (calcul formel)

3. Puces correctes (démonstration automatique)

4. 2500 points Elo aux échecs (hardware)

5. Scanner et imagerie médicale (vision)

6. Génôme (BD, algorithmes, calcul parallèle)

7. Cinéma (synthèse d'image)

8. Météo, physique des hautes énergies
   (super calculateurs)

F9 (*Mark S. Manasse, Arjen Lenstra*)

$2^{512} + 1 =$
13407807929942597099574024998205846127479365820592393377723561443721764030073546976801874298166903427690031858186486050853753882811946569946433649006084097 = 2424833
×
74556028256478842083373957362004549187833663426 57
×
74164006262753080152478714190193747405994078109751902390582131614441575950470500809281871169394073 7

Avril à Juin 1990 avec

Dean Alvis, Thomas G. Arnold, Bob Ayers, Pete Balkus, Jœl Bartlett, David M. Bernardo, Anita Borg, Wieb Bosma, Patrick Boyle, Richard P. Brent, John Brownie, Bruce Bullis, Jerome Chailloux, Chran-Ham Chang, Henri Cohen, Leisa Condie, Bob Devine, Jeff Dike, John Dillon, Jeremy Dion, Mike Doody, Alan Eustace, Mike Ferrara, John Forecast, Hania Gajewska, Stephen Gildea, Andy Goldstein, Tim Greenwood, Ramsey Haddad, Gary Harding, B. J. Herbison, Jim Horning, Felix S Hsu, Scott Huddleston, Peter Ilieve, Jeff Janock, Mike Johnson, Kevin Jones, Bill Kalsow, Irving Kaplansky, Harsh Kapoor, Bill Katz, Christopher A. Kent, Alan Kirby, Manfred Kœthe, Ted Lemon, Walter Liœn, Todd Little, Joseph A. Martin, Murray S. Mazer, James T. McCartney, Jœl McCormack, Steven Miller, Jeffrey Mogul, Francois Morain, David Mostardi, A. E. Mossberg, Jeff E. Nelson, Marc Nozell, Tom Patterson, Nigel Poole, Eric D. Postpischil, Dennis Racca, Jon Reeves, Brian Reid, Herman

te Riele, John Riley, John Robinson, Eduardo Santiago, Richard Schedler, Michael Sclafani, Mark Shand, Bob Silverman, Al Simons, Kiran Somalwar, Bill Sommerfeld, Bob Souza, Alan Stanier, John T. Stapleton, Jorge Stolfi, Geof Stone, Richard Swan, Ed Taranto, Bob Thomas, Gary Thomas, Dennis Ting, Win Treese, Percy Tzelnic, Brick Verser, Paul Vixie, David Wall, Dick Wilkins, Dik Winter, Ted Wobber, Frank M. Woodall, Jr.

Calcul fait

- début mi-février 1990 à Bellcore sur 35 machines (Ds3100 et Sparc)
- mi-mars: réseau de Firefly de SRC (triplement de la puissance totale)
- fin mars: aide du réseau Internet (700 Ds3100, pour un total de 340 années de Mips)

1990 est vieux.

Factorisation – bis

Plus grand nombre factorisé: $(12^{151}-1)/11$ avec 162 chiffres décimaux.

Plus grand nombre *ordinaire* factorisé: RSA 130, avec quelques sites. (Arjen K. Lenstra, Bellcore, 11 avril 1996), $11527.00

RSA-130 =
1807082088687404805951656164405905566278102516769401349170127021450056662540244048387341127590812303371781887966563182013214880055 =
3968599945959745429016112616288378606757644911281006483255515724 3
×
4553449864673597218840368689727440886435630126320506960099044599

RSA-129 factorisé en 5000 mips années, 8 mois, 600 volontaires de 20 pays. Matrice 569466 × 524338 réduite à une matrice dense 188614 × 188160 par élimination gaussienne, 4.13 GO, 45h de MasPar MP-1 16K. $100.00

Factorisation de C120: 61 machines (Richard Wackerbarth, 5 mars 1996) avec l'aide de
Damien.Doligez@inria.fr, HJSmith@ix.netcom.com, Michel.Kern@inria.fr, Paul.Zimmermann@loria.fr, etc

Démonstration automatique ou assistée

Matériel

vérification d'un protocole de communications entre les caches de la machine multi-processeurs GigaMax, Encore. *McMillan* (CMU), 1992

idem pour un chip de communications déja réalisé et qui ne fonctionnait pas. *Fujita*, Fujitsu, 1993

conception hiérarchique du Power PC, IBM, 1992 ($\simeq$ vieille méthode Bull). Aussi chez DEC.

automates de Buchi pour spécifier et vérifier un circuit de gestion de communications. *Kurshan*, (ATT Bell Labs)

Pour le logiciel, c'est plus dur

preuves des programmes concurrents

preuves des gros programmes ...

preuves de protocoles de communication: "Bounded Retransmission Protocol", (Philips Research), en 1994.

axiomatisation en TLA de protocole de contrôle de flux pour des lignes série très rapide, (Bull recherche), 1994.

Les domaines

langages de programmation
validation des programmes
démonstration automatique
parallélisme et communication
logique mathématique
langages formels et compilation
bases de donnée

algorithmique
algorithmique géométrique
algorithmique parallèle
analyse d'algorithmes
codes et cryptographie
zero knowledge algorithms
calcul formel

systèmes d'exploitation
matériel et architecture
CAO
vision
robotique

L'informatique traite des calculs sur des structures de données autres que numériques.

Une technologie exponentielle

	gain	période
processeur vitesse	$\times 1.35$	1 an
capacité mémoire	$\times 4$	3 ans
coût/octet disques	$\times 3/4$	1 an

Actuellement,

Processeur à 500 MHz, 4M transistors,
RAM 16 Mbits, 60ns
Disques 2.5" à 1GO, 3.5" à 4GO (5KF)
Réseau à 12 M machines

Les réseaux — défi des années 90

courrier électronique
échanges de programmes
consultation de bibliothèques
minitel planétaire
sécurité et confidentialité

Evolution des systèmes informatiques

Quelques grands pas

66: GE-645	mémoire virtuelle, Multics
70: pdp11	Unix
72: alto	bitmap, souris, wysiwyg
74: ethernet	réseau entre les altos
75: dover	impression laser, Press
78: Vax et Unix-BSD	Unix avec mémoire virtuelle
82: Sun Microsystems	Station personnelle Unix
84: uucp	Interconnexion à longue distance (IP)
88: Machines RISC	IBM 801, Titan, SPARC+Mips
91: Portables, format A4	Poids et autonomie
93: Mosaic, Internet	Grand public

Quelques noms

Organick	Multics
Thompson	Unix
Ritchie	Unix
Lampson	Alto
Thacker	Alto
Metcalfe	Ethernet
Warnock	Press, Postscript
Bell	pdp11, Vax
Joy	Unix BSD, SunOS
Leffler	Unix + réseau
Hennessy	RISC, Mips
Patterson	RISC, Berkeley

Le *dynabook* d'Alan Kay

"The Learning Research Group at Xerox PARC is concerned with all aspects of the communication and manipulation of knowledge. We design, build, and use dynamic media which can be used by human beings of all ages. Several years ago, we crystallized our dreams into a design idea for a personal dynamic medium the size of a notebook (the **Dynabook***) which could be owned by everyone and could have the power to handle virtually all of its owner's information-related needs..."*

Alan Kay et Adele Goldberg,
Xerox Palo Alto Research Center,
Computers, Mars 1977

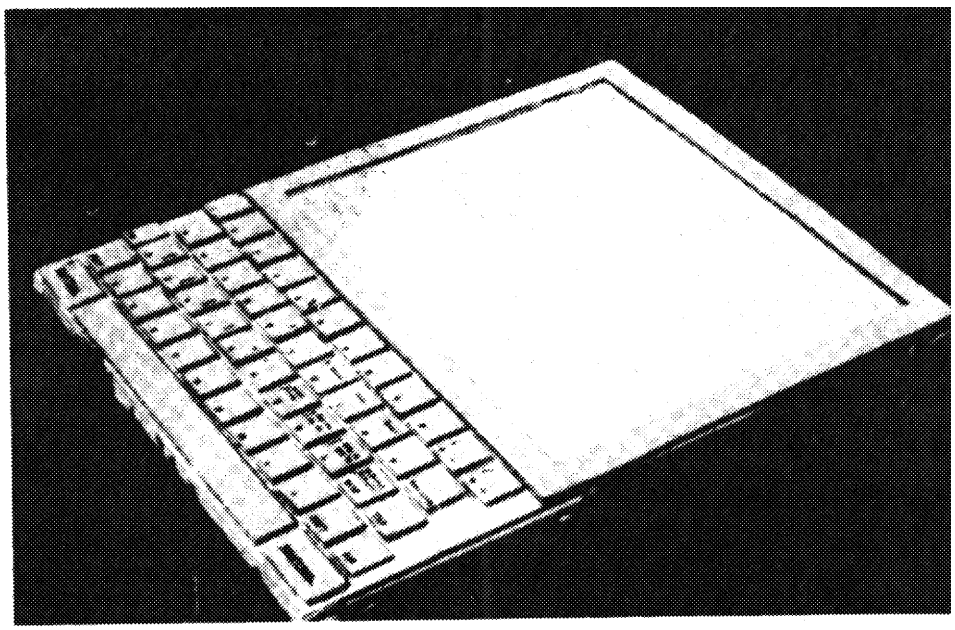


Figure 2. Mock-up of a future Dynabook.

Les portables à l'X

Bons informaticiens (vs ENS, ETH, MIT)

Réseau dans les chambres

Communication accrue (courrier, panneaux électroniques)

Service impression commode

Développer le service fichiers et stations Unix

Pas une mauvaise affaire

Format du cours

Développer le travail personnel

Formation initiale très hétérogène des élèves

$\simeq$ MIT, sauf que l'X bénéficie du pensionnat.

Futur dans le monde informatique

Les fichiers et les processeurs délocalisés

Atteignables partout et à tout moment

Le matériel informatique de l'X

Machines Unix

Vax 9000, Digital, 50 mips, `poly`

10 alphas, 75 mips

12 suns IPX, 20 mips, `voitures`

17 hp 9000, 60 mips, `poissons`

5 PowerMac,

2 Silicon Graphics,

1 machine passerelle vers l'Internet, `sil`

32 terminaux X

Autres

PB 190, avec ThinkPascal et ThinkC, et émulateur terminal

PC Toshiba, avec TurboPascal et TurboC++, et émulateur terminal

Quelques recommandations

Sécurité réseau

CERT (*Computer Emergency Response Team*), dépend de la NSA (*National Security Agency*), fait la police sur le réseau.

Le réseau n'est opérationnel que si une auto discipline (simple) est respectée.

Centre Informatique de l'X

Pas des *punching balls*. Climat de confiance. Très difficile à 450 (900). Si pbs, on coupe l'Internet.

Impression ($\simeq$ 10-20 centimes la page). Ne pas faire des impressions inutiles. Utiliser les photocopieuses.

Il faut arriver à créer la notion d'élève *root*. Réalisable si confiance, mais ...

Elèves délégués

Un délégué par PC
Repas ensemble à la fin octobre
2 délégués info pour la promo

Enseignement de l'informatique à l'X

1. Tronc Commun, obligatoire
   1er trimestre, 1ère année;
   Projet informatique,
   2ème semestre, 1ère année

2. Majeure 1, optionnelle
   1er trimestre, 2ème année

3. Majeure 2, optionnelle
   2ème trimestre, 2ème année

Contenus

TC: algorithmes et programmation
Cori et Lévy

Maj 1: mathématiques et informatique
Vuillemin
algèbre des codes, architecture, logique,
vision et synthèse, complexité, ML

Maj 2: informatique
Cousot
compilation, systèmes, bases de données,
sémantique, algorithmique
Prolog, biologie, calcul parallèle

Plan du TC

1. 3 initiations: lundi 9 Sep – mardi 24 Sep.

2. 10 petites classes: jeudi 26 Sep – jeudi 12 Déc.

3. 10 travaux dirigés: lundi 30 Sep – mardi 17 Déc.

4. contrôle HC 2h: mercredi 13 Nov.

5. composition de classement 4h: lundi 6 Jan,
   Cœfficient 10.
   Théorique.

6. projet de programmation, 2ème semestre,
   amphi de présentation: jeudi 19 Déc?
   500–1000 lignes de Pascal ou C ou Caml,
   $\sim$20 sujets différents, cœfficient de difficulté,
   Cœfficient 8.
   Pratique, travail personnel.
   Souvent pluridisciplinaire.

Le contenu du Tronc Commun

1. Structures de données élémentaires (tableaux, listes, files)
2. Arbres
3. Graphes
4. Récursivité
5. Modularité
6. Analyse syntaxique élémentaire

Langages de programmation

Pascal, Wirth

C, Kernighan et Ritchie

ML / Caml, Milner

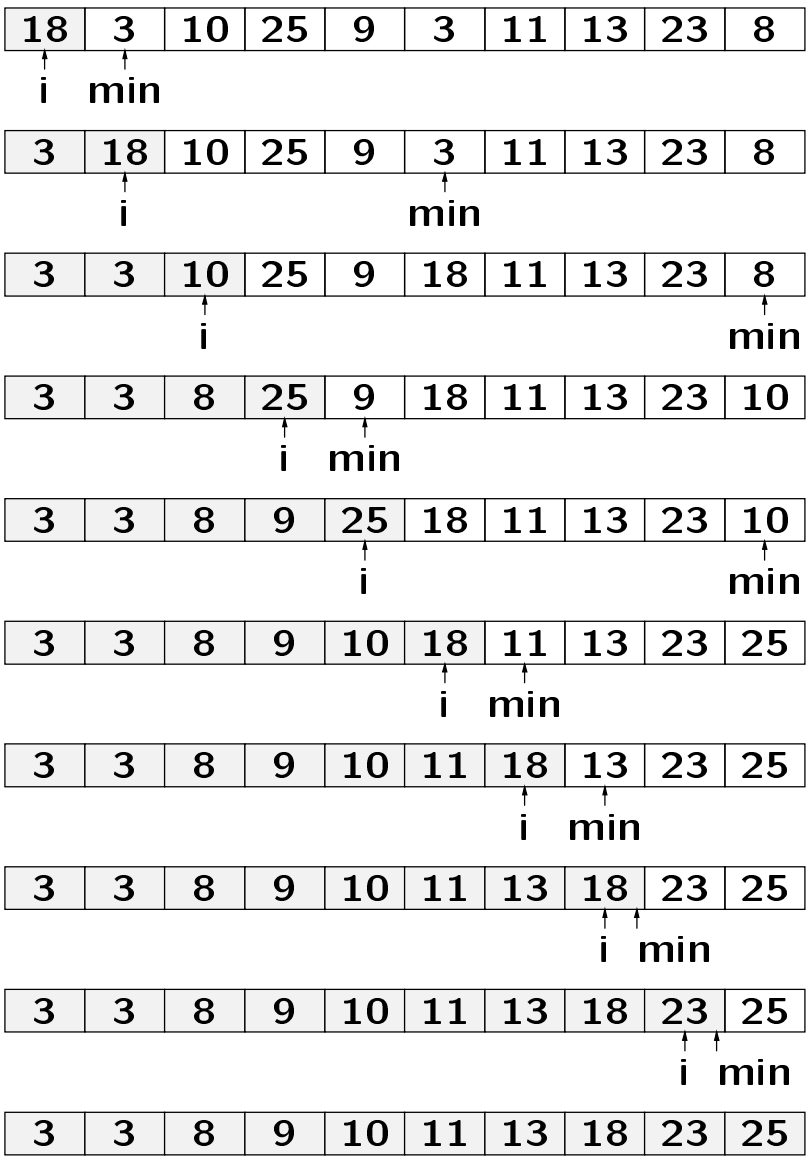
Programmation

1. Itération — Tableaux, Listes
2. Récursivité — Arbres, Graphes
3. Modularité – Gros programmes
4. Exploration – Arbre de recherche
5. Analyse syntaxique
   Chaînes de caractères $\Longrightarrow$ Arbres

Maxime du TC

Le matériel passe, le logiciel reste

Le tri par sélection (programme itératif)



Le tri par sélection (bis)

Algorithme

Pour $i = 1$ à N faire

> **Chercher l'emplacement min de l'élément minimum dans $[i + 1, N]$,**
>
> **Echanger $a[i]$ et $a[min]$.**

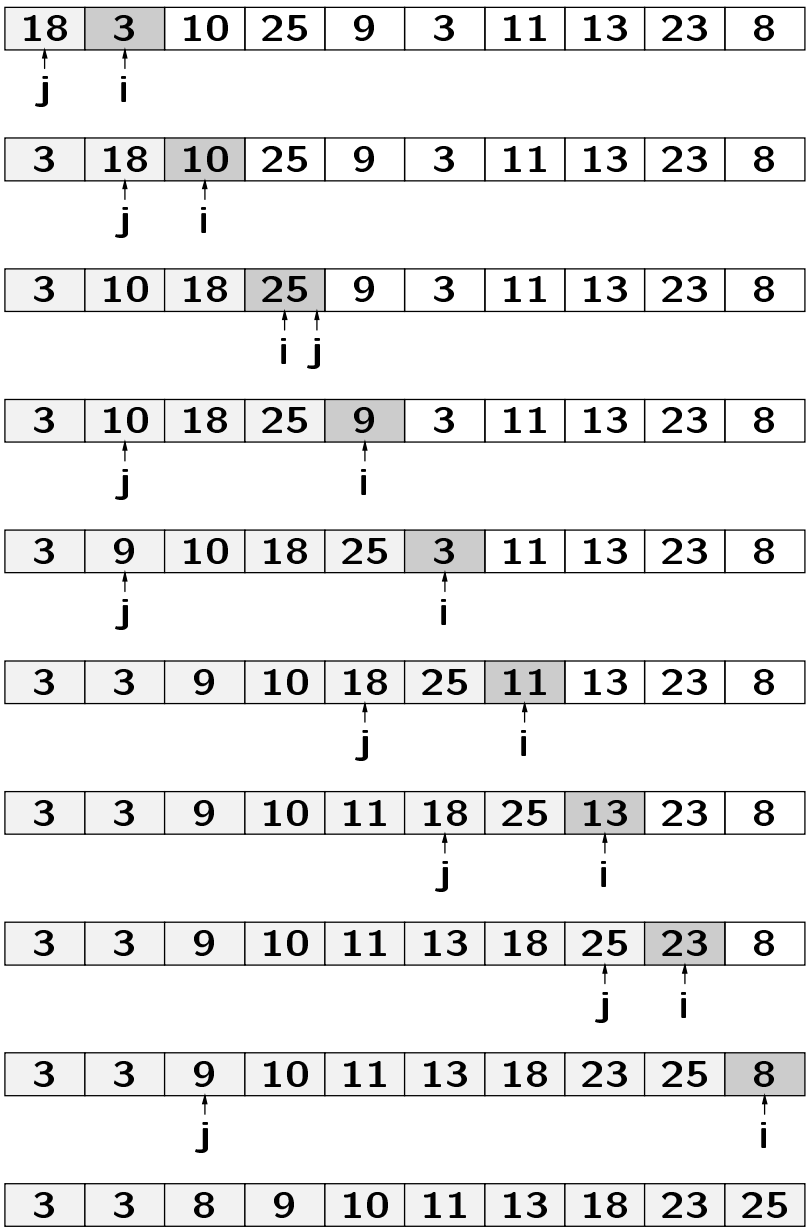
Correction

> **Début:** $\langle a[1] \rangle$ **trié**
>
> **A tout moment,** $\langle a[1], a[2], \cdots, a[i] \rangle$ **triés**
>
> **A la fin,** $i = N$ **et** $\langle a[1], a[2], \cdots, a[N] \rangle$ **triés**

Coût

> $N - 1 + N - 2 + \cdots + 1$ **comparaisons**
>
> $= N(N-1)/2$
>
> $\simeq O(N^2)$ **opérations**

Le tri par insertion



Le tri par insertion

Algorithme

Pour $i = 2$ à N faire

> **A partir de i, aller vers la gauche et trouver l'emplacement où insérer a[i] dans [1, $i-1$]**
>
> **Insérer a[i] en décalant vers la droite les éléments visités.**

Correction

> **Début: $\langle a[1] \rangle$ trié**
>
> **A tout moment, $\langle a[1], a[2], \cdots, a[i-1] \rangle$ triés**
>
> **A la fin, $i = N$ et $\langle a[1], a[2], \cdots, a[N-1] \rangle$ triés et après l'insertion de $a[N]$, on a $\langle a[1], a[2], \cdots, a[N] \rangle$ triés**

Coût

> **$1 + 2 + \cdots + N - 1$ comparaisons**
>
> **$= N(N-1)/2$**
>
> **$\simeq O(N^2)$ opérations**

Le tri par fusion (programme récursif)

18	3	10	25	9	3	11	13	23	8

3	18	10	9	25	3	11	13	8	23

3	18	9	10	25	3	11	8	13	23

3	9	10	18	25	3	8	11	13	23

3	3	8	9	10	11	13	18	23	25

Le tri par fusion (programme récursif)

Algorithme

Si $N = 0$ (tableau vide), ne rien faire.
Sinon:
1- Trier la moitié gauche du tableau a.
2- Trier la moitié droite du tableau a.
3- Fusionner les 2 résultats.

Correction

immédiate.

Coût

$C_N = 2C_{\lfloor N/2 \rfloor} + N$

$C_0 = 0$

$C_N \simeq N \log N$

Le gain de vitesse vient de la décomposition récursive du tableau a.

Les Arbres (structure de données récursive)

Définition

$$\textbf{Arbre} = \{\textbf{Vide}\} \uplus \textbf{Arbre} \times \textbf{Element} \times \textbf{Arbre}$$

Exemple d'algorithme sur les arbres

Taille (Vide) = 0
Taille (Arbre(g, n, d))
= 1 + Taille (g) + Taille (d)

Correction

immédiate.

Coût (N nœuds)

$$C_N = N$$

On parcourt un arbre récursivement en temps linéaire.

Itération / Récursion

- Programmes récursifs et structures de données récursives se correspondent.
- L'écriture récursive est presque toujours plus élégante.
- Induction mathématique et récursivité se correspondent souvent.
- La difficulté est toujours de trouver les structures récursives associées à des problèmes combinatoires (exemple: test de planarité des graphes, *depth first search*, Tarjan 72).
- Tous les langages modernes autorisent la récursivité.
- Certains permettent aussi l'écriture récursive des structures de données (par exemple Caml)