

Bienvenue!



Mon cheminement



UNIVERSITÉ
LAVAL

2000-04

Bacc. en génie informatique

Carnegie Mellon

2004-06

M.S. en robotique

Carnegie Mellon

2006-11

Ph.D. en robotique



Disney Research
Pittsburgh

2011-13

Post-doc chez Disney



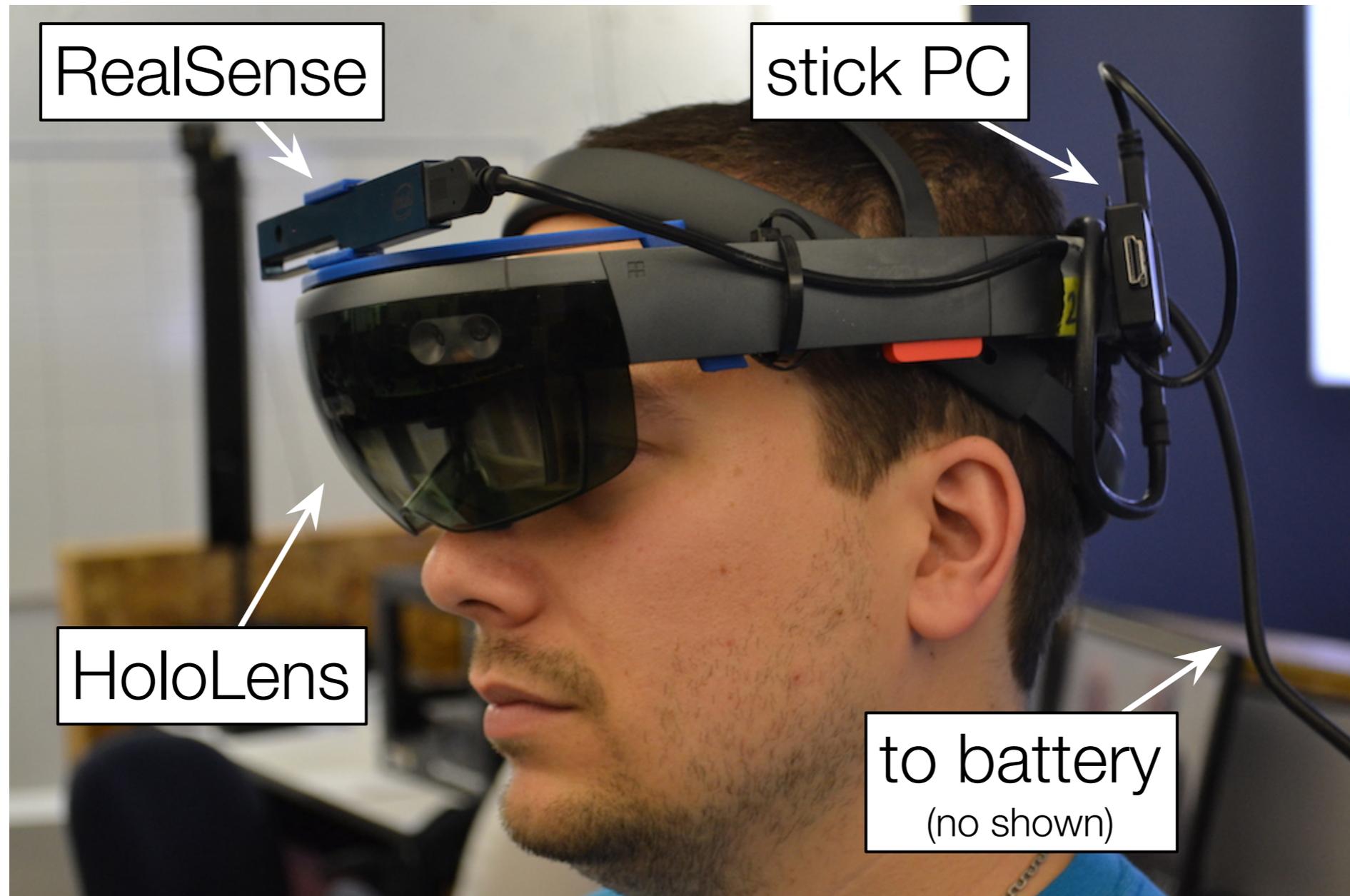
UNIVERSITÉ
LAVAL

2013-...

Prof. en GEL-GIF

Réalité augmentée & intelligence artificielle

<https://vimeo.com/175462212>



Éclairage extérieur à partir d'un visage



Éclairage extérieur à partir d'une photo



Éclairage intérieur à partir d'une photo



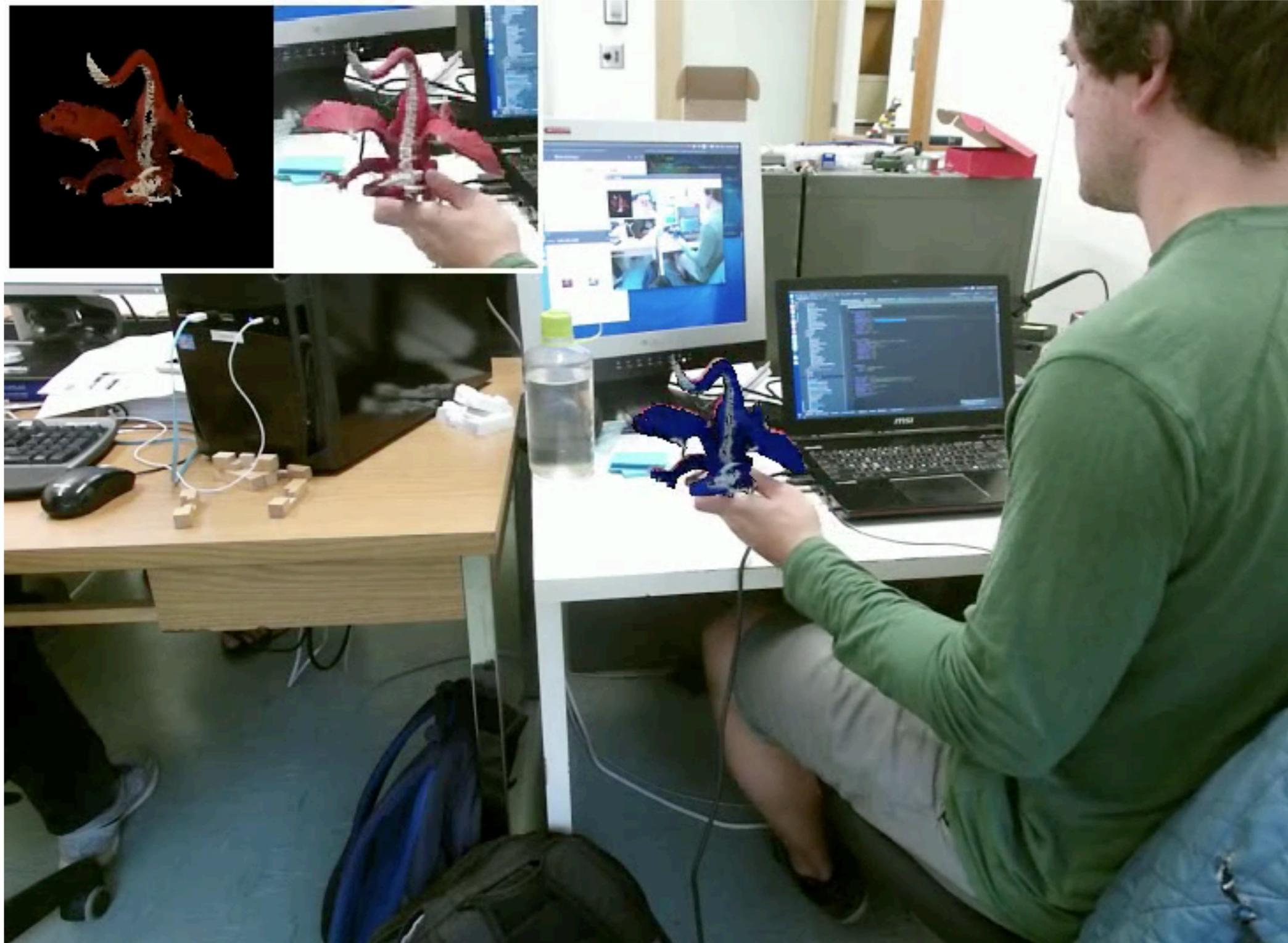
Éclairage intérieur à partir d'une photo



Suivi d'objets en 3D (début 2017)



Suivi d'objets en 3D (fin 2017)



Suivi d'objets en 3D (maintenant)



GIF-4105/7105 Photographie Algorithmique

<https://vimeo.com/213144547>



crédit: Maxime Leclerc



crédit: Yannick Hold-Geoffroy

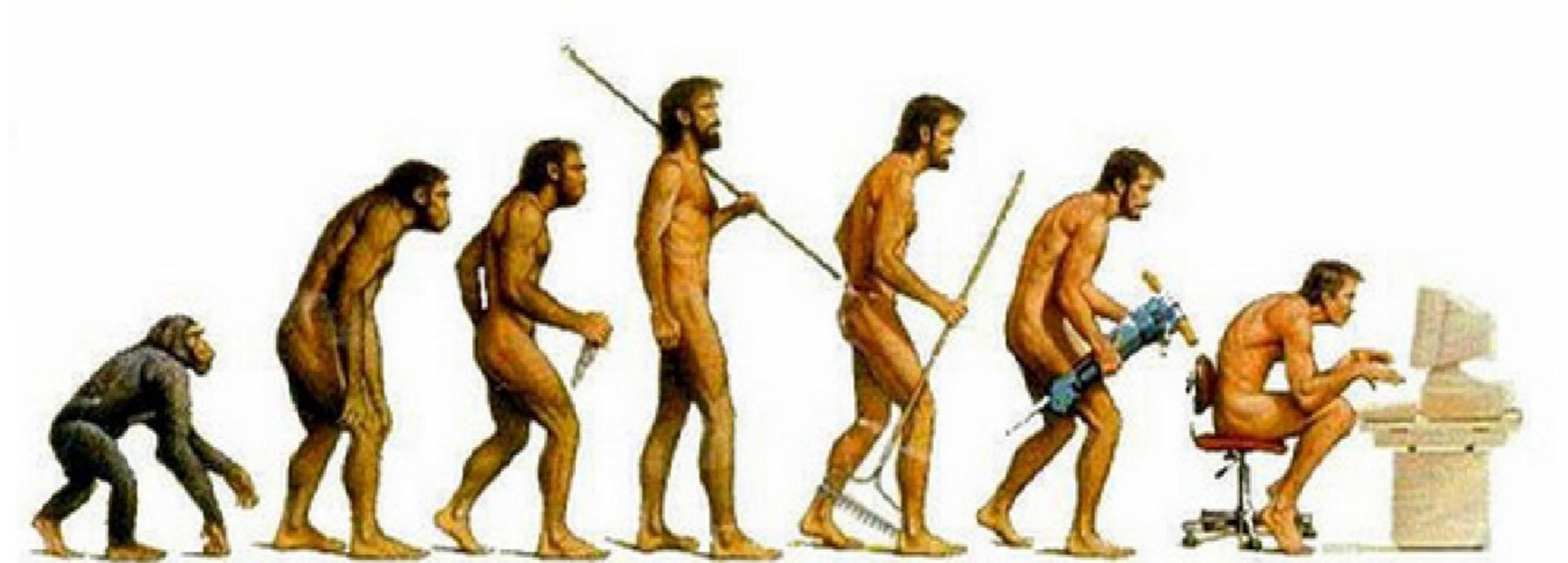


crédit: Cédric Tremblay

Crédits supplémentaires: « Niaisez votre prof »



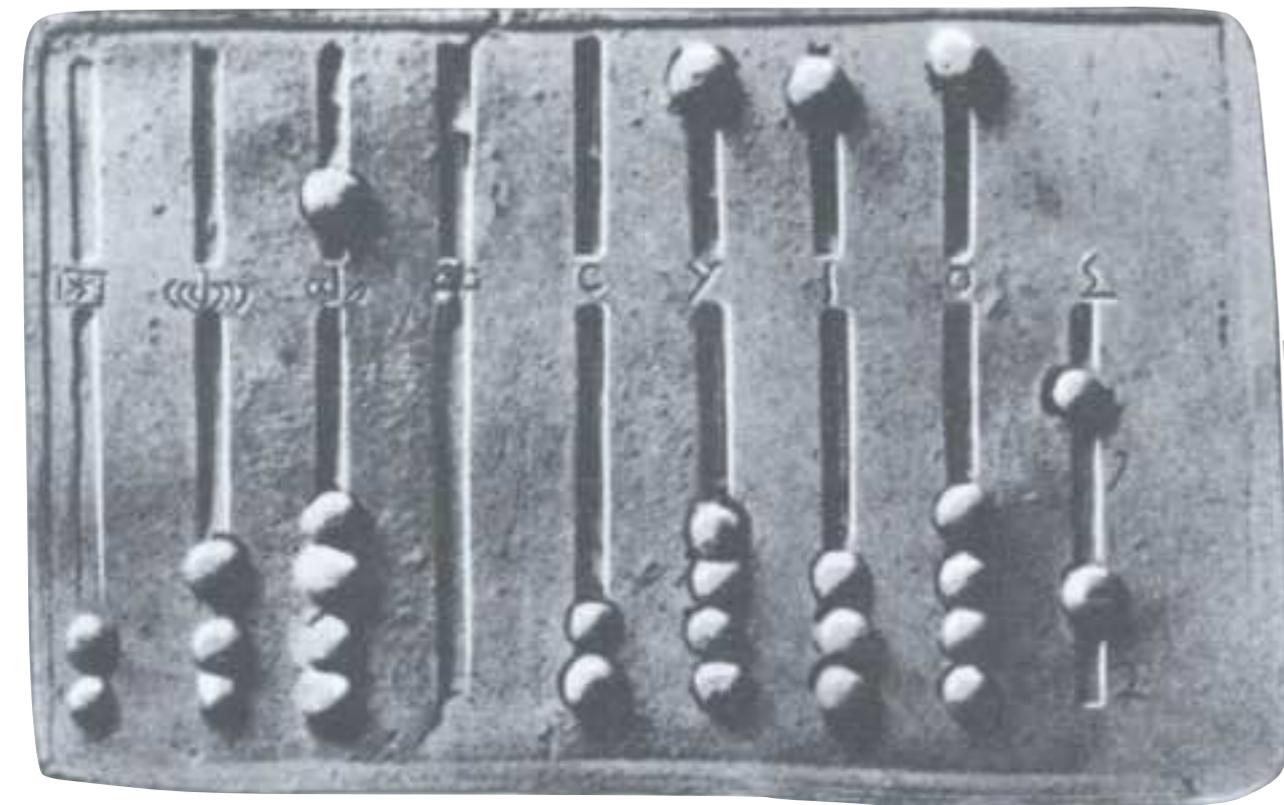
Bref historique des ordinateurs



Abaque (500BC—1500AD)

Grec/Romain

Chinois (boulier)



Capable de stocker des données et de faire des calculs sur ces données

Ordinateur

Stocke des données.

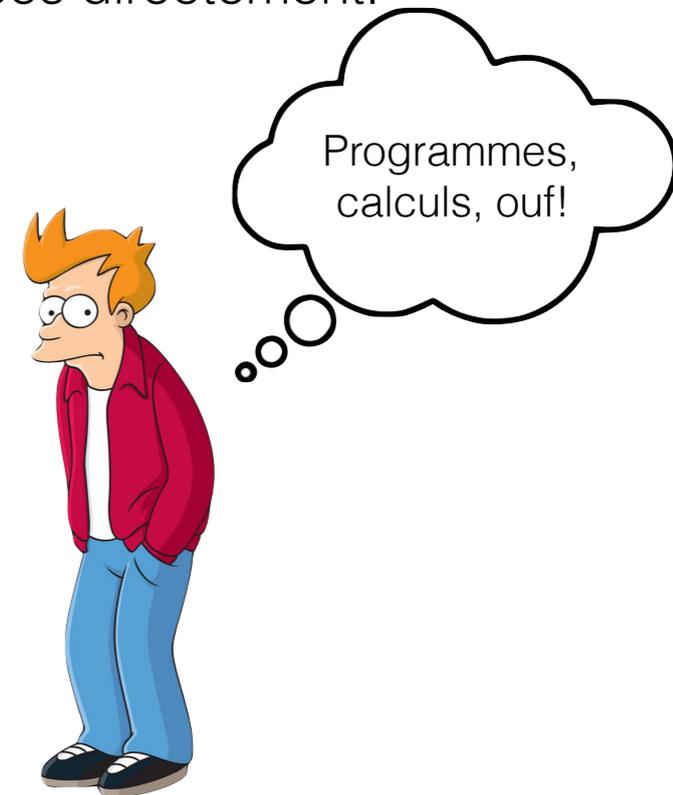
Mémoire

Stocke des données



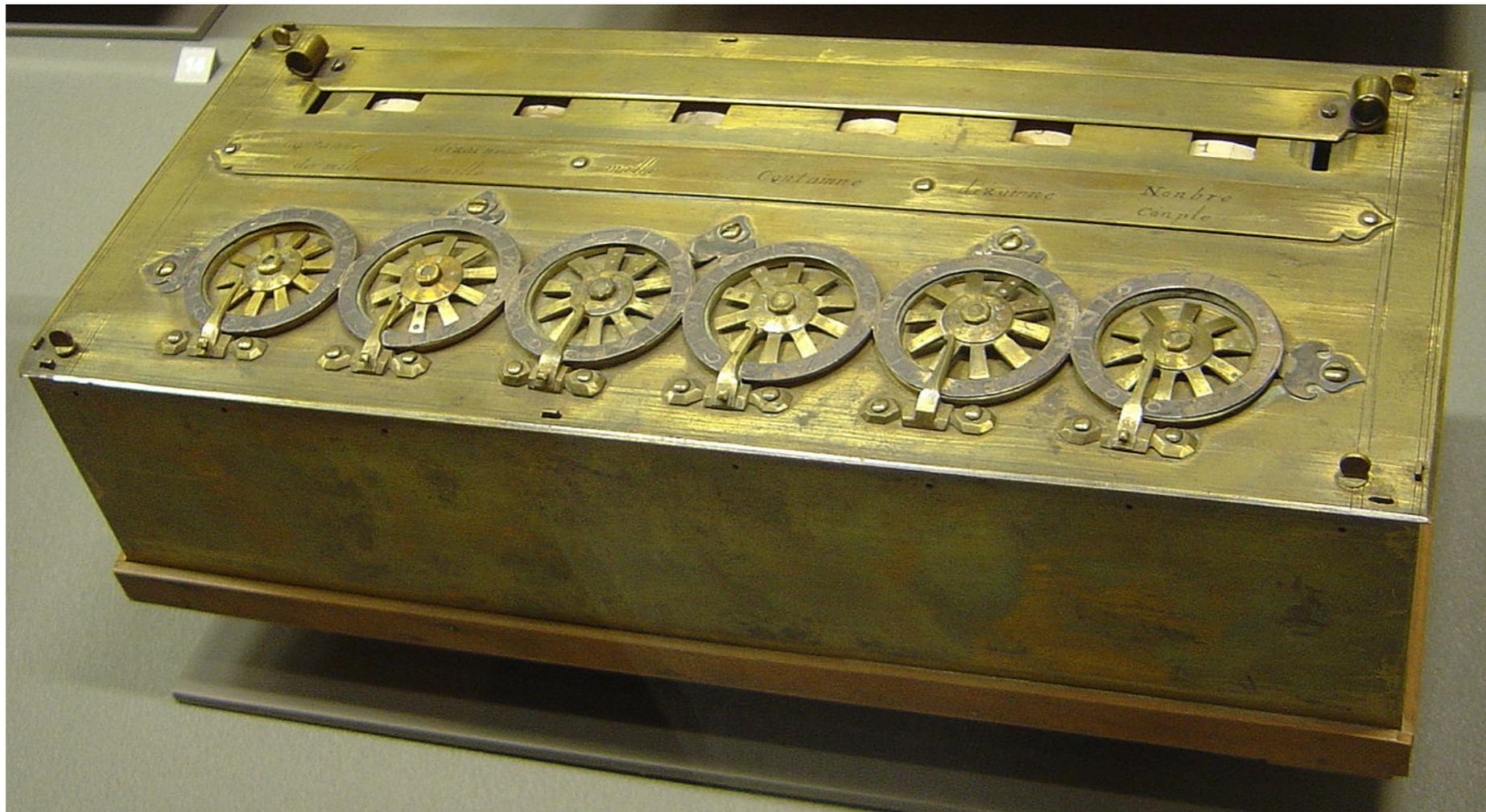
Utilisateur (nous)

Interagit avec l'ordinateur en manipulant les données directement.



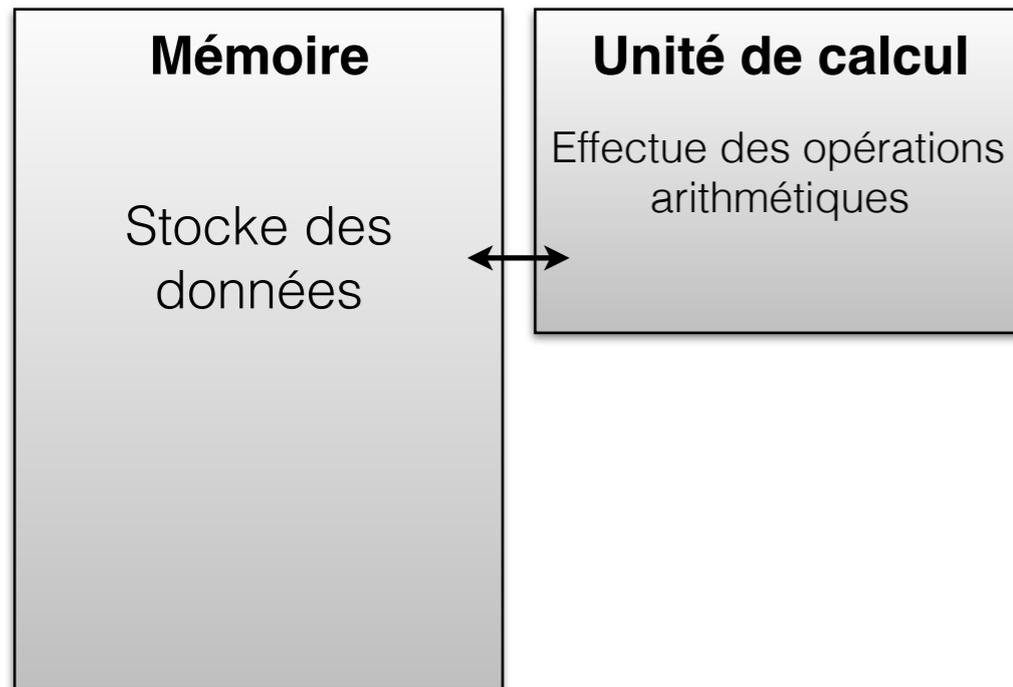
Pascaline (Blaise Pascal, 1642)

«machine arithmétique» conçue par Blaise Pascal à l'âge de 19 ans!
additions, soustractions, multiplications et divisions



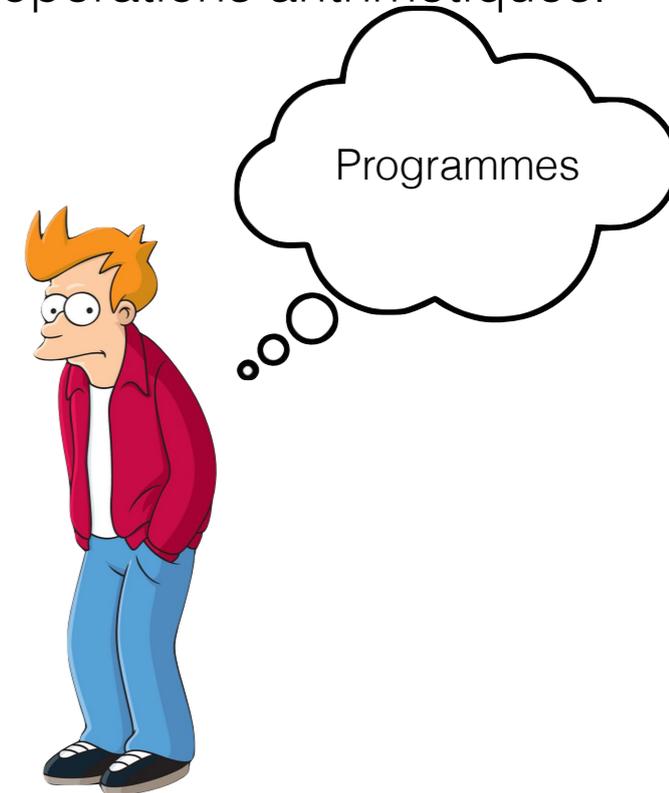
Ordinateur

Stocke des données.
Effectue des opérations arithmétiques simples.



Utilisateur (nous)

Interagit avec l'ordinateur en manipulant les données et les opérations arithmétiques.



Métier à tisser semi-automatique (1801, Joseph Marie Jacquard)

Les motifs à tisser pouvaient être programmés par cartes perforées!

Première fois qu'un « programme » pouvait être « enregistré »



Machine analytique (Charles Babbage, 1837)

Premier ordinateur à usage général

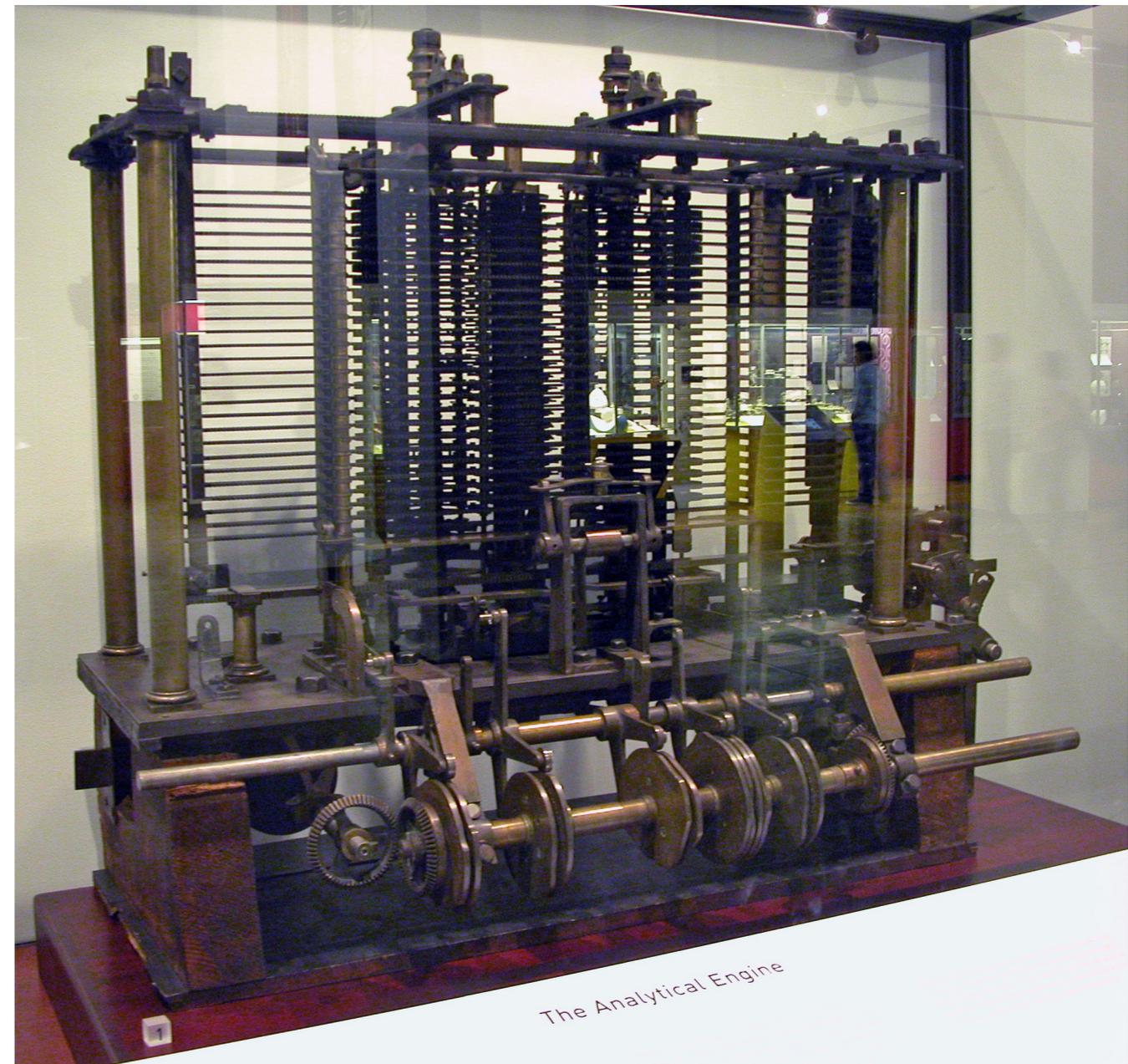
Incluait:

unité de calcul centrale

mémoire

programmes (cartes perforées)

chiffres décimaux



version préliminaire, la version complète
n'a jamais été construite

Qui a été le premier
programmeur?

Ahem, programmeuse

Ada Lovelace (1815–1852)

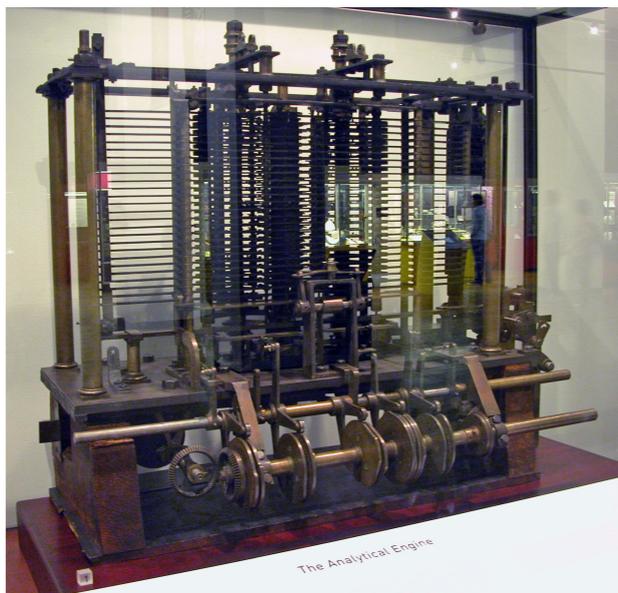
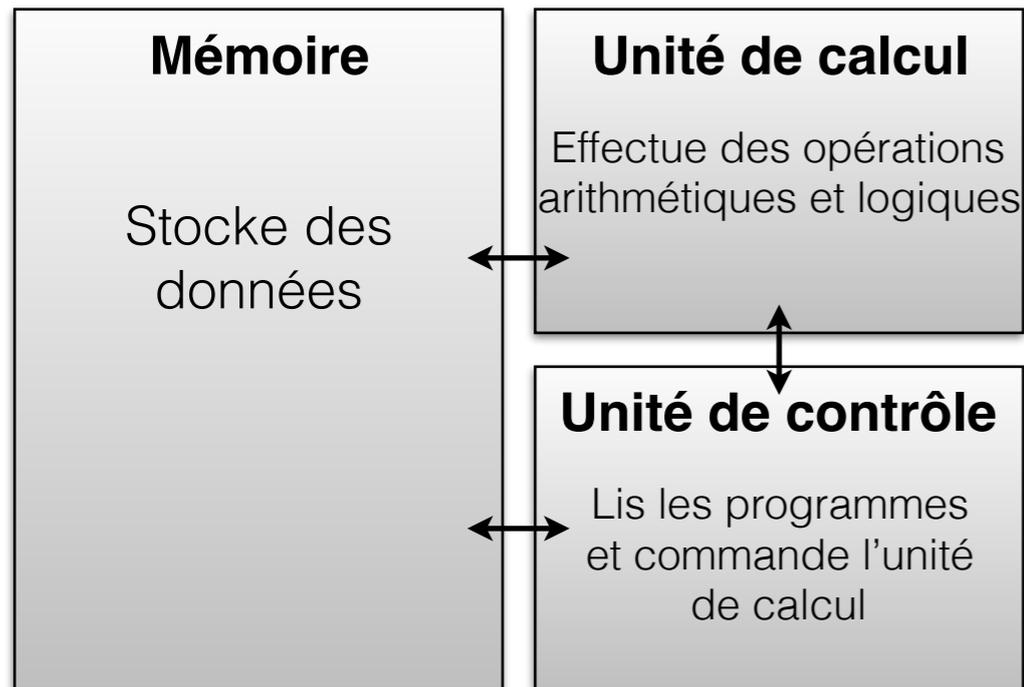


Ordinateur

Stocke des données.

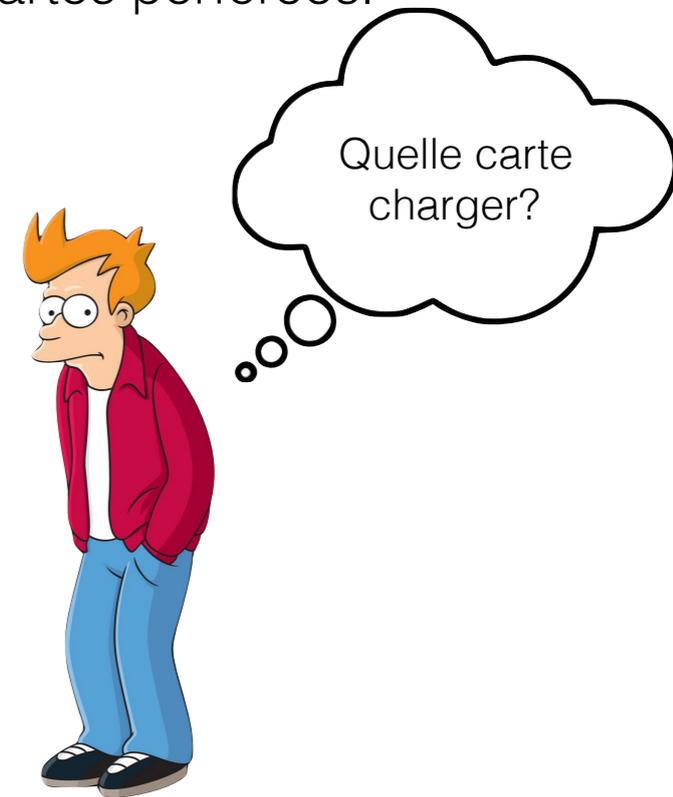
Effectue des opérations arithmétiques et logiques.

Effectue des calculs selon un programme.



Utilisateur (nous)

Interagit avec l'ordinateur en chargeant des cartes perforées.



Programmes



ABC (John V. Atanasoff, 1939)

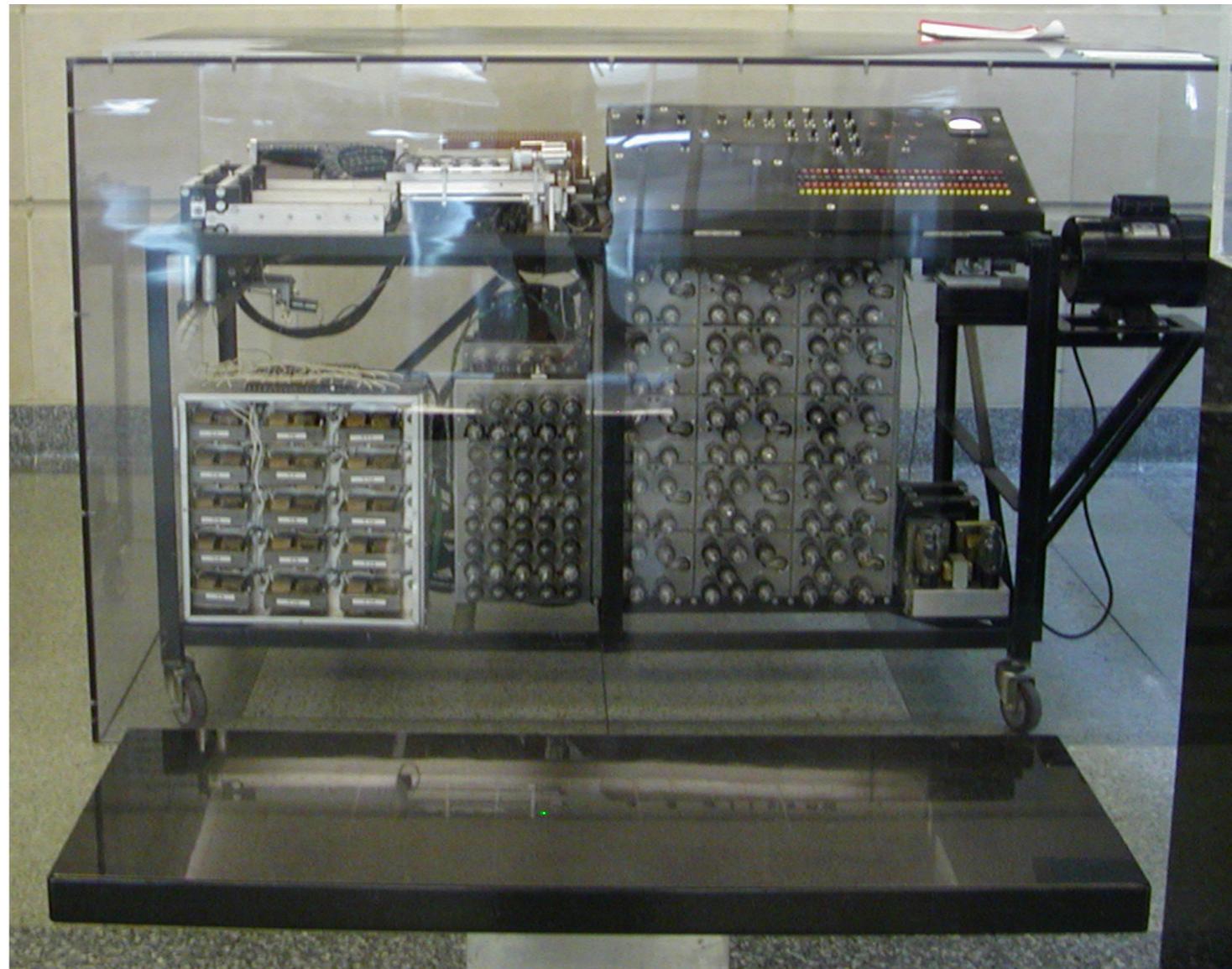
Premier ordinateur électronique

Innovations:

Tubes à vide (au lieu d'engrenages)

Représente les nombres en **binaire**

Séparation entre calculs et mémoire

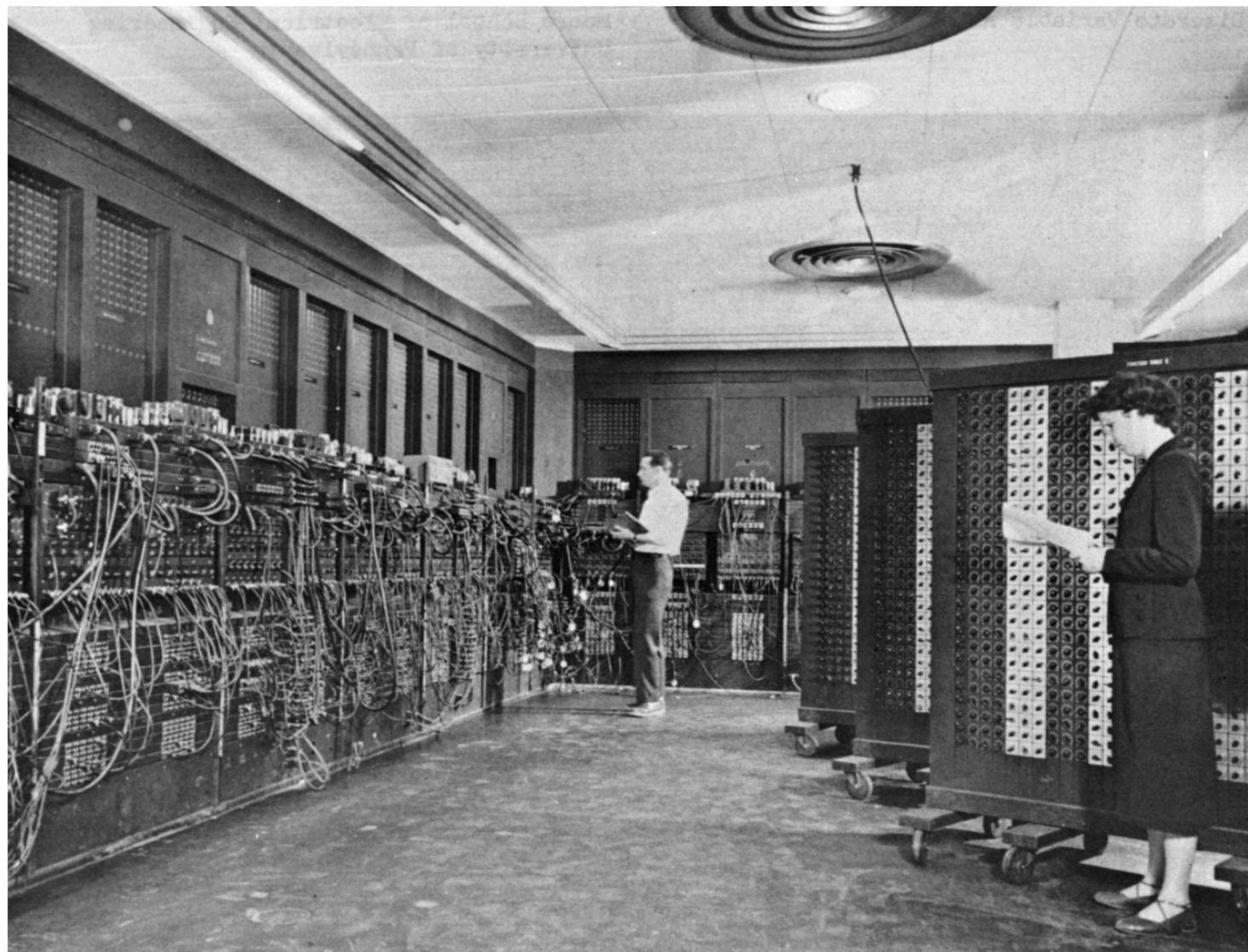


ENIAC (Maulchy & Eckert, 1946)

calculs de balistique durant la 2e Guerre Mondiale

les panneaux à gauche étaient les programmes: il fallait programmer manuellement en branchant et débranchant ces connexions

18,000 tubes à vide, 15,000 pieds carrés, 30 tonnes, 140 kwatts puissance



Premier «bug»?

9/9

0800 Antan started

1000 " stopped - antan ✓

1300 (032) MP - MC ~~1.482147000~~ { 1.2700 9.037 847 025
 2.130476415 } 9.037 846 995 connect
 (033) PRO 2 2.130476415 4.615925059(-2)
 connect 2.130676415

Relays 6-2 in 033 failed special speed test
 in relay " 11.000 test.

Relays changed

1100 Started Cosine Tapc (Sine check)

1525 Started Mult + Adder Test.

1545  Relay #70 Panel F
 (moth) in relay.

First actual case of bug being found.

~~1630~~ 1630 Antan started.

1700 closed down.

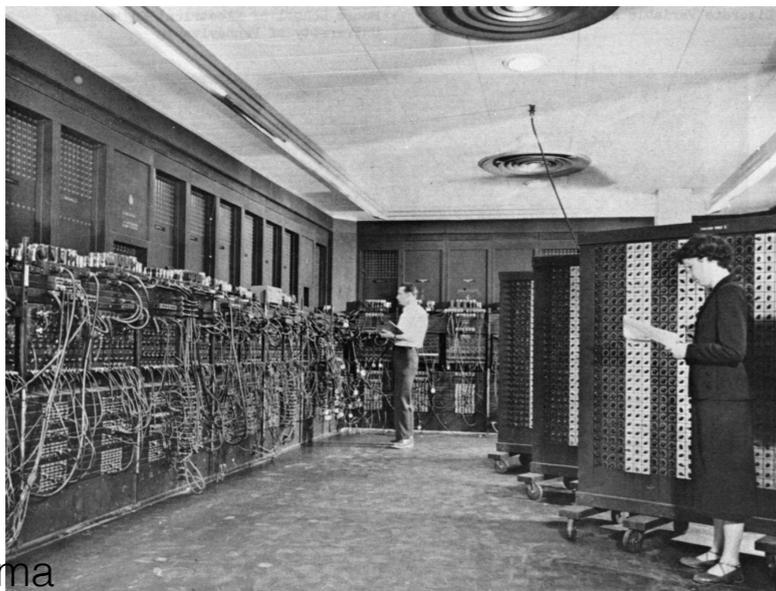
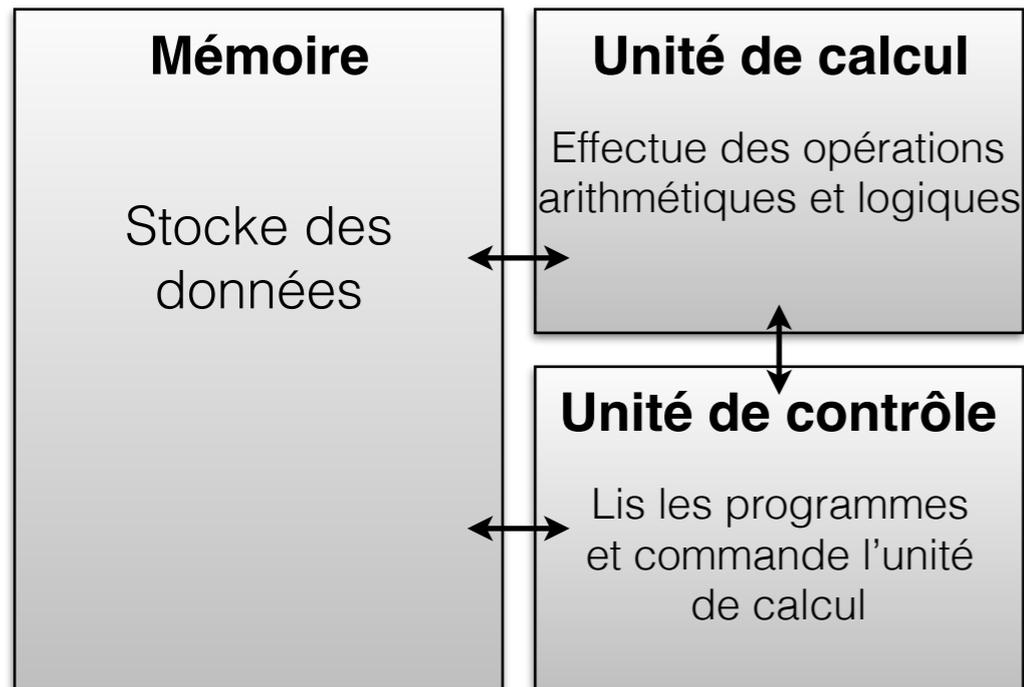
Relay
 2145
 Relay 3376

Ordinateur

Stocke des données.

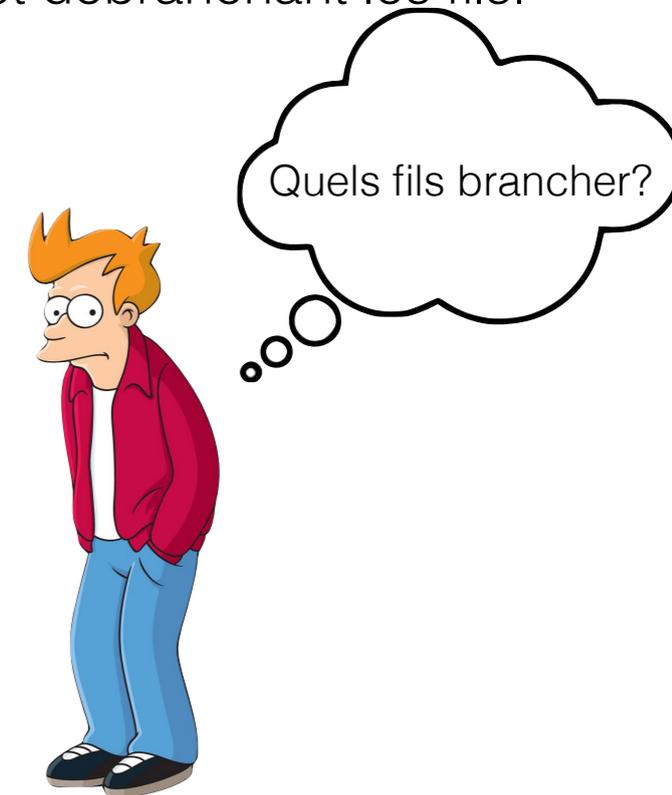
Effectue des opérations arithmétiques et logiques.

Effectue des calculs selon un programme.

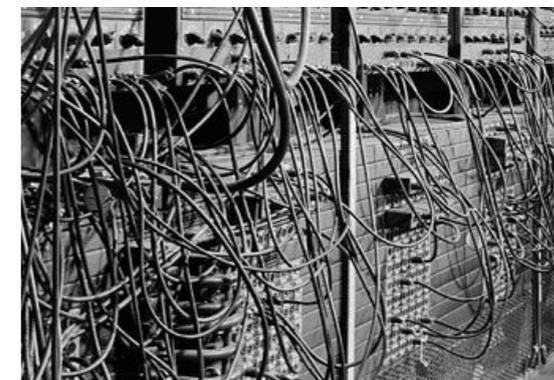


Utilisateur (nous)

Interagit avec l'ordinateur en branchant et débranchant les fils.



Programmes



Architecture « von Neumann »

John von Neumann



- 4 composantes principales:
 - mémoire
 - unité de calcul arithmétique et logique (ALU)
 - unité de contrôle (CU)
 - équipement d'entrées et sorties (I/O)
- La mémoire contient les données ET les programmes
- Implémentations initiales: EDVAC et IAS (avec tubes à vide)
- Architecture toujours en utilisation aujourd'hui!

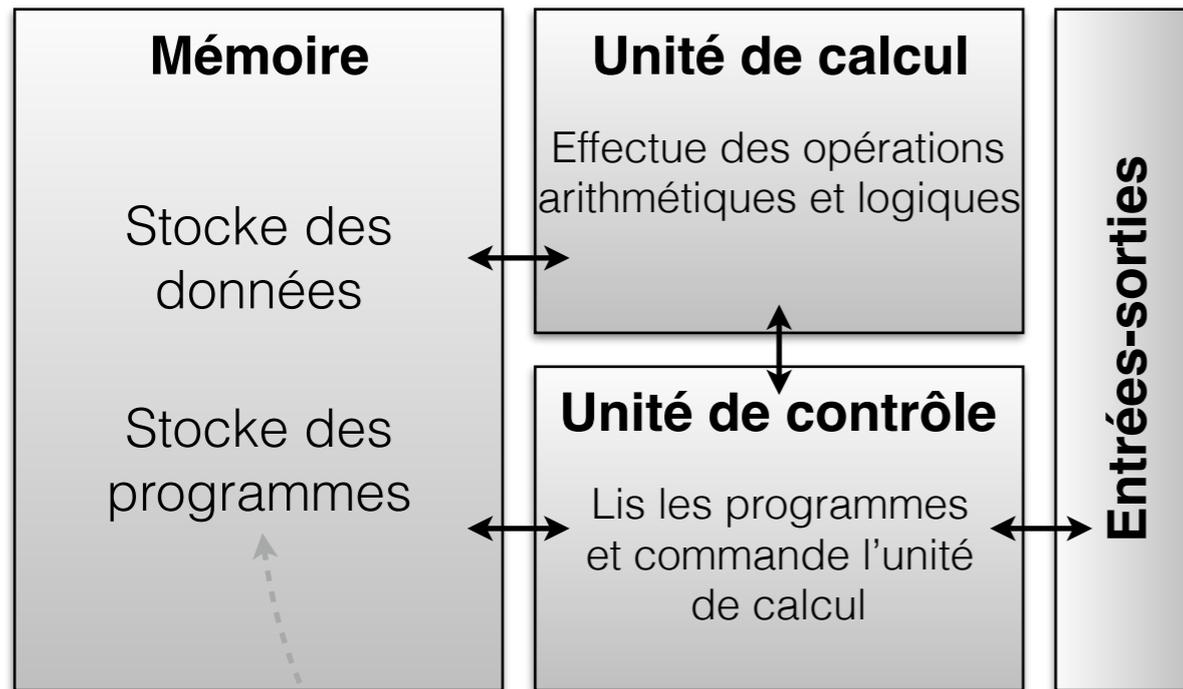
Ordinateur

Stocke des données.

Effectue des opérations arithmétiques et logiques.

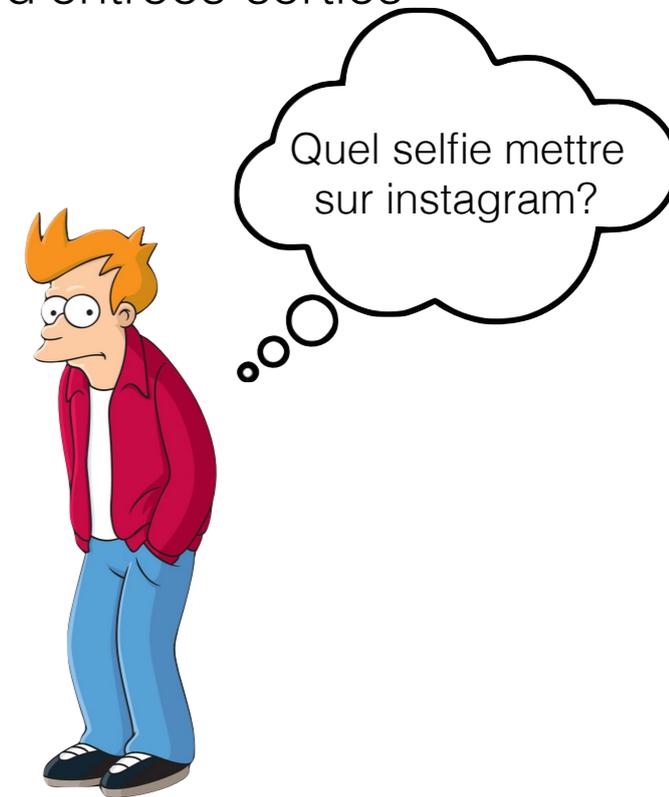
Effectue des calculs selon un programme.

Programme stocké en mémoire!



Utilisateur (nous)

Interagit avec l'ordinateur
avec les modules d'entrées-sorties



Architecture Von Neuman

Programmes

Tubes à vide

- Dispendieux
- Très fragiles
 - Temps moyen de fonctionnement de l'ENIAC: 5 heures!
- Énergivores, nécessitaient d'imposants systèmes de refroidissement

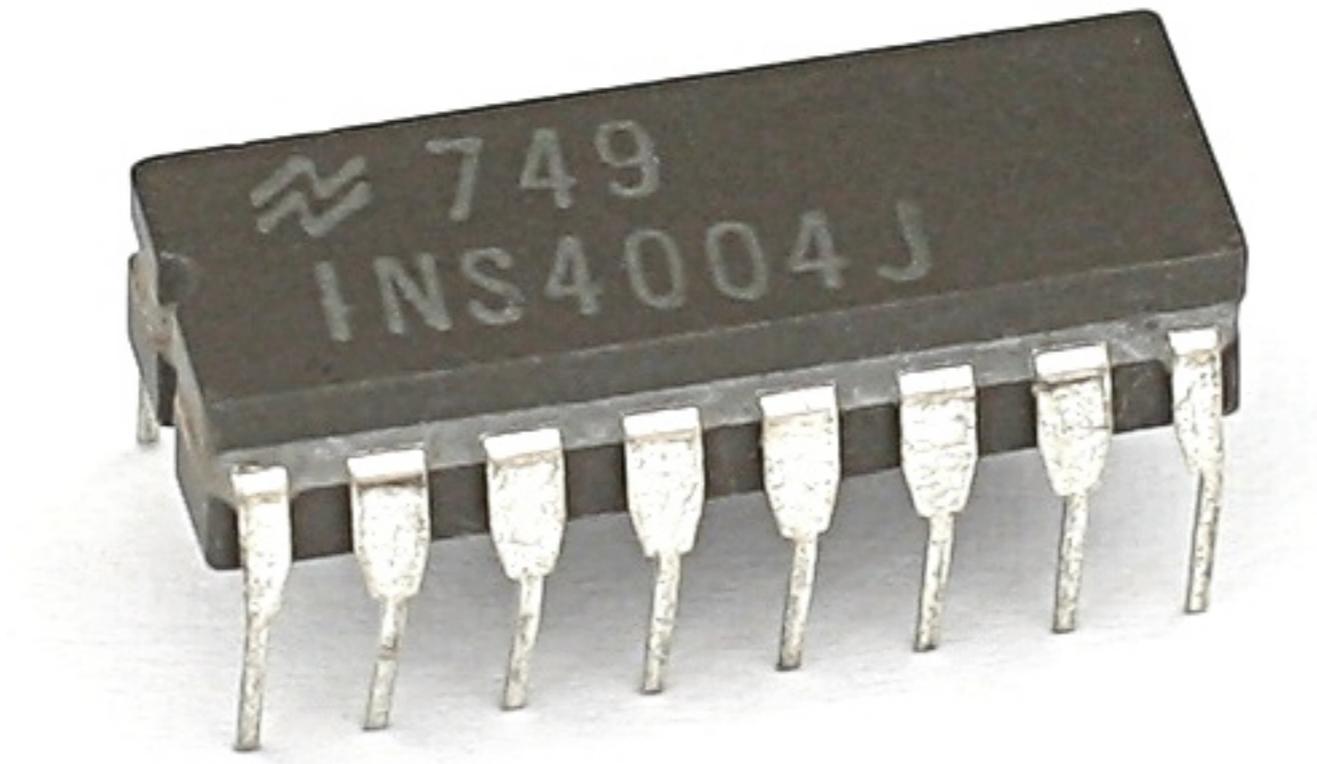
Transistor (Bardeen, Brattain, Shockley, 1947)

- Même fonctionnalité qu'un tube à vide
 - porte logique nécessaire aux calculs
- Plus rapide, plus compact, plus solide, moins dispendieux

premier transistor



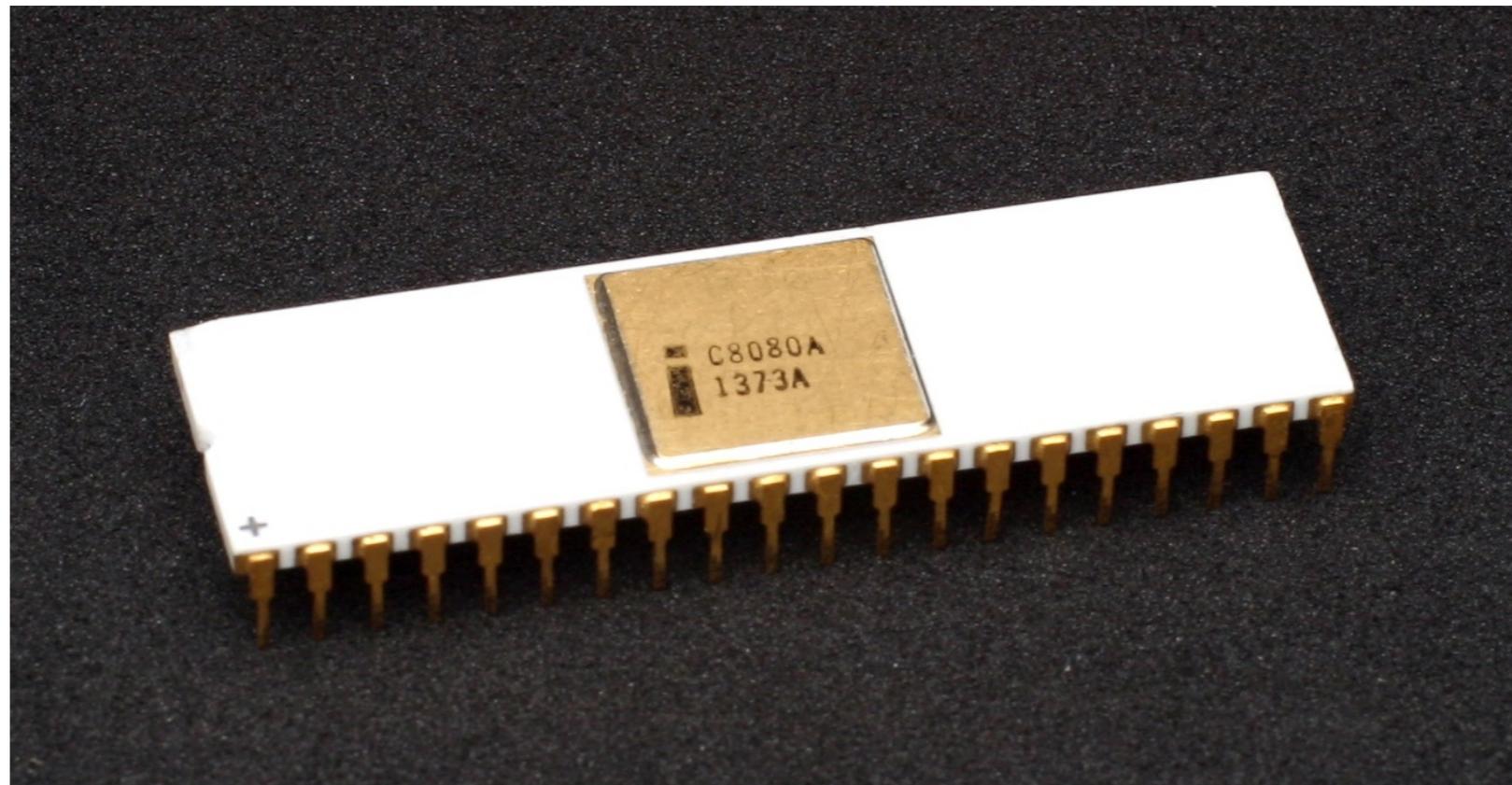
Premier microprocesseur: 4004 (Intel, 1974)



- Microprocesseur: circuit intégré comprenant unité de calcul, unité de contrôle, mémoire (limitée), et entrées-sorties
 - Vous vous rappelez von Neumann?
- CPU à 4 bits

Intel 8080 (1974)

- Premier micro-processeur réellement « tout usage »
- Vitesse d'horloge limite de 2 MHz



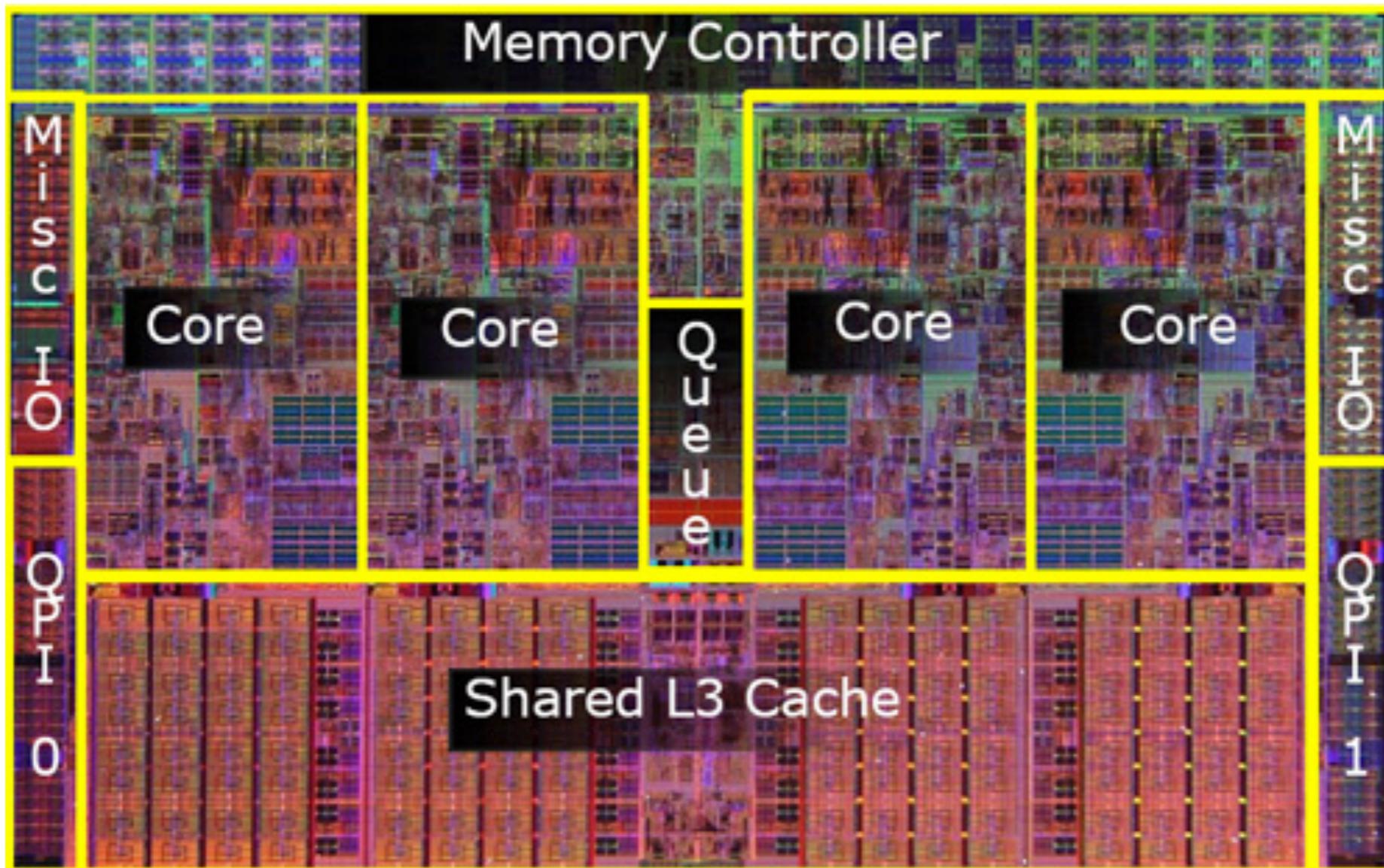
1971 — 2015: loi de Moore (1965)

- Gordon Moore, co-fondateur d'Intel, a établi que le nombre de transistors doublerait à tous les 2 ans
- Plus une prédiction, ou une observation, plutôt qu'une « loi »
- Plusieurs autres facteurs importants à considérer pour mesurer la performance
 - Vitesse d'horloge
 - Architecture (puces dédiées, multi-coeurs, etc.)
 - Améliorations logicielles
 - Économique
- Il y a une limite?
 - Semble toujours être dans les 5–10 prochaines années...
 - Effets quantiques limitent la miniaturisation des transistors, par contre de nouvelles (nano)-technologies compensent

Le cours d'OSA

Thème 1 : structure

Connaître la structure interne des ordinateurs



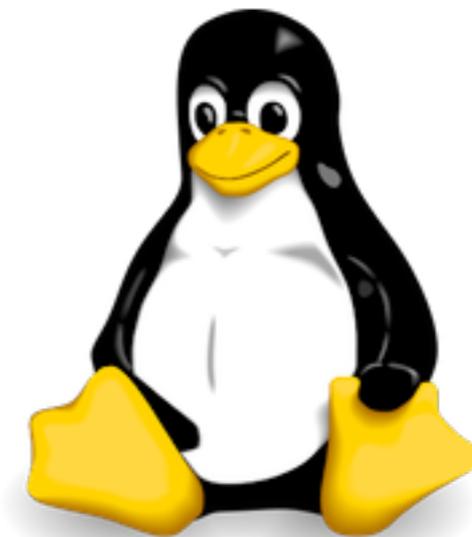
Thème 2: assembleur

Expérience concrète de programmation en langage assembleur
afin d'exploiter cette structure interne

Hello, world!

Thème 3: systèmes d'exploitation

Comprendre les principales fonctionnalités d'un système d'exploitation



Thème 4: entrées-sorties

Explorer les interactions entre le micro-processeur et le monde externe

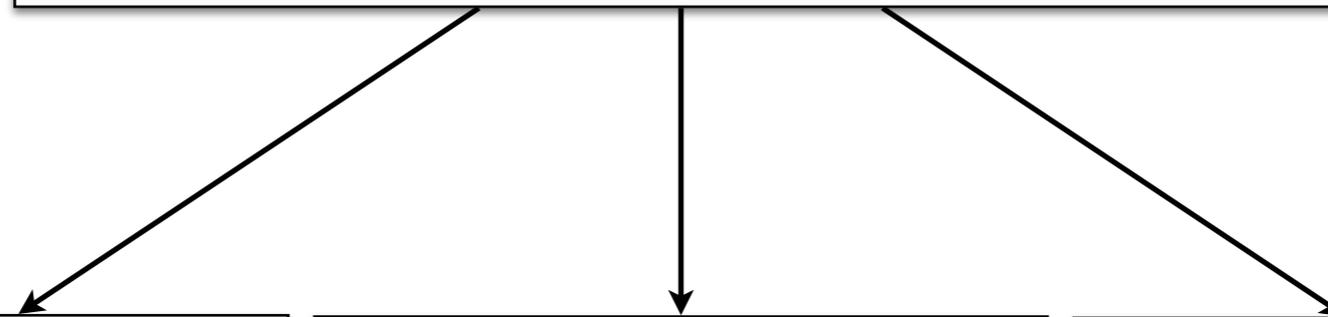


Pourquoi OSA?

- Programmeurs
 - Écrire des programmes plus performants (même si vous n'écrivez jamais d'assembleur)
- Utilisateurs
 - Mieux comprendre votre outil de travail

Pourquoi OSA?

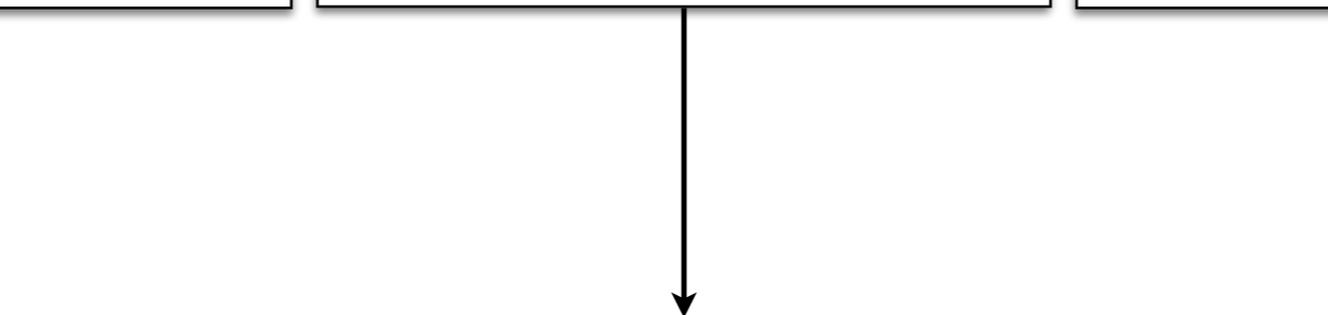
GIF-1001
Ordinateurs: structure et applications



GIF-3002
Systèmes micro-processeurs et interfaces

GLO-2001
Systèmes d'exploitation

GLO-2000
Réseaux pour ingénieurs



GIF-3000
Architecture des micro-processeurs