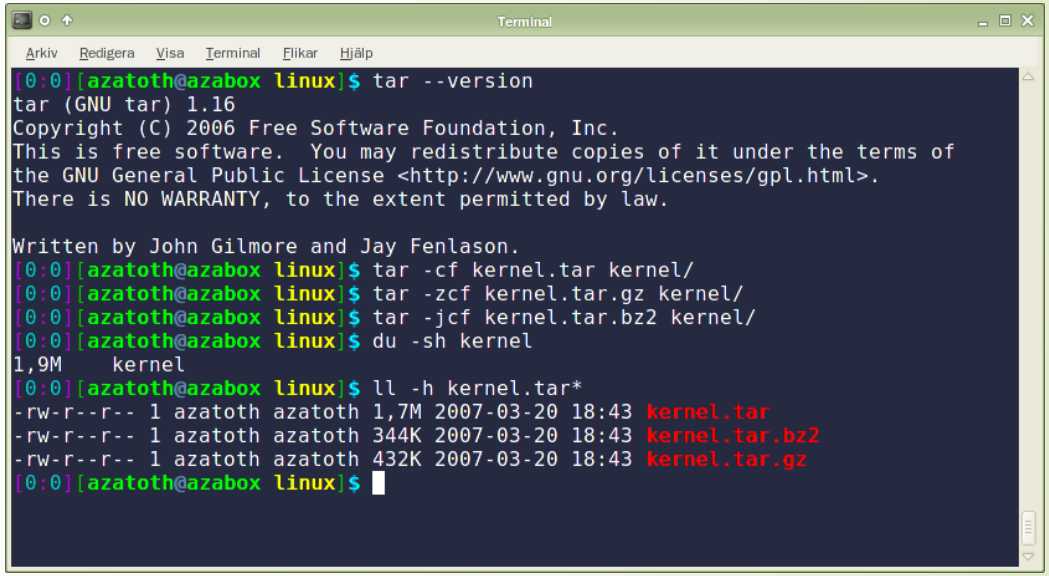
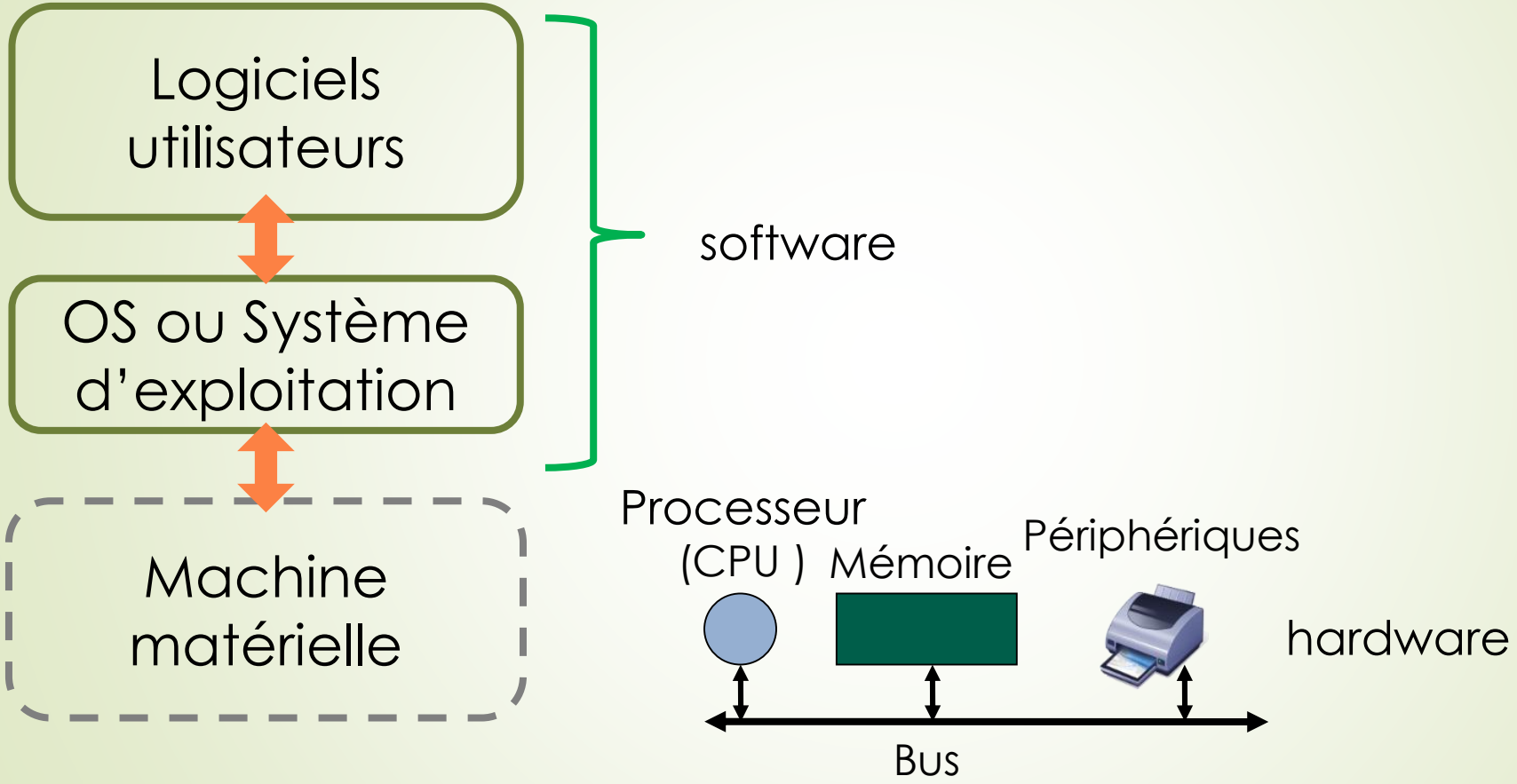
Introduction à la programmation



L'ordinateur
Visite rapide

Architecture en trois couches



Architecture de Von Neumann

John von Neumann
(1903-1957)

Le processeur

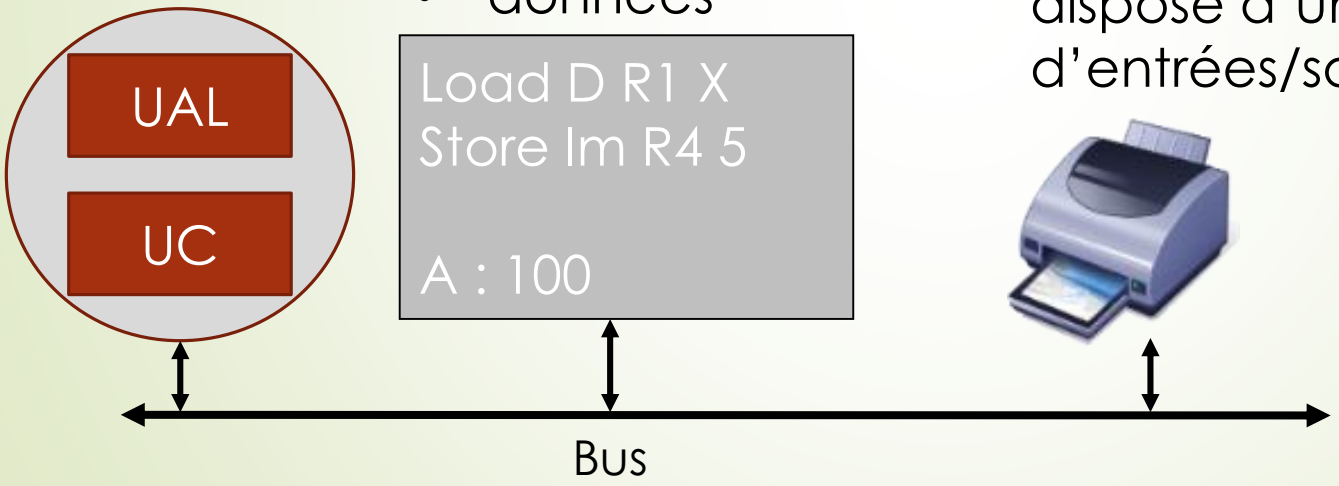
exécute les instructions

La mémoire centrale

contient le programme à exécuter :

- instructions
- données

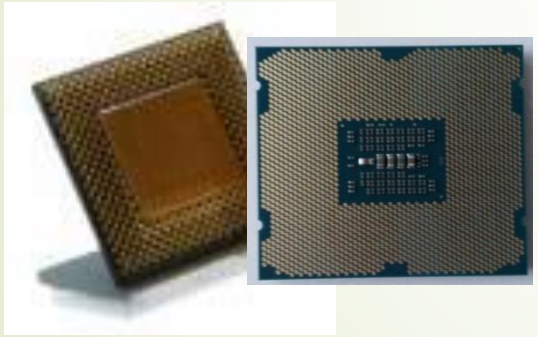
L'ordinateur dispose d'unités d'entrées/sorties



Le processeur

- Le **processeur** (**CPU**, pour *Central Processing Unit)* est le cerveau de l'ordinateur. Il permet de manipuler, des données et des instructions codées sous forme binaires.
- Le **processeur** est un circuit électronique cadencé au rythme d'une horloge interne qui envoie des impulsions, appelées « **top** ». La **fréquence d'horloge**, correspond nombre d'impulsions par seconde. Elle s'exprime en Hertz (Hz).

  - Ordinateur à 2 GHz → l'horloge envoie 200 000 000 000 battements par seconde.



Circuits électroniques composés de millions de transistors placés dans un boîtier comportant des connecteurs d'entrée-sortie, surmonté d'un ventilateur.

→ **circuit intégré** ou **puce**

Les mémoires de l'ordinateur

le cnam

Une « **mémoire** » est un composant électronique capable de stocker temporairement des informations

- Une mémoire est caractérisée par :
  - Sa capacité, représentant le volume global d'informations (en bits) que la mémoire peut stocker (par exemple 1 Goctets, soit 230 octets, soit 230 * 8 bits.
  - Son temps d'accès, correspondant à l'intervalle de temps entre la demande de lecture/écriture et la disponibilité de la donnée.
- L'ordinateur contient différents niveaux de mémoire, organisés selon une hiérarchie mémoire.

Les grandeurs de l'ordinateur
Capacité – bit - octet

le cnam

1 octet = 8 bits (byte)	Avant 1998	Après 1998
Kilooctet (Ko)	2^{10} octets = 1024 octets	1000 octets
Mégaoctet (Mo)	2^{20} octets = 1024 Koctets	1000 Koctets
Gigaoctet (Go)	2^{30} octets = 1024 Moctets	1000 Moctets

Multiples de l'octet : préfixes SI et mésusages

Nom	Symbole	Valeur	Mésusage[2]
kilooctet	ko	10^3	2^{10}
mégaoctet	Mo	10^6	2^{20}
gigaoctet	Go	10^9	2^{30}
téraoctet	To	10^{12}	2^{40}
pétaoctet	Po	10^{15}	
exaoctet	Eo	10^{18}	
zettaoctet	Zo	10^{21}	
yottaoctet	Yo	10^{24}	

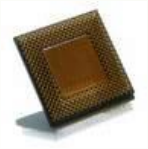
Multiples de l'octet : préfixes binaires

Nom	Symbole	Valeur
kibioctet	kio	2^{10}
mébioctet	Mio	2^{20}
gibioctet	Gio	2^{30}
tébioctet	Tio	2^{40}
pébioctet	Pio	2^{50}
exbioctet	Eio	2^{60}
zébioctet	Zio	2^{70}
yobioctet	Yio	2^{80}

Les mémoires de l'ordinateur

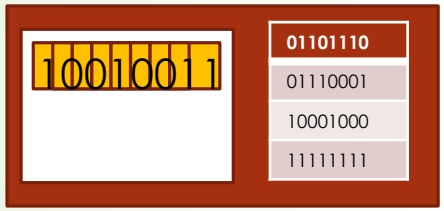
le cnam

- L'ordinateur contient différents niveaux de mémoire, organisés selon une **hiérarchie mémoire**.



Mémoires vives : mémoires **volatiles** :

Mémoires de masse : mémoires **permanentes**



REGISTRES
N bits (32, 64)
1 nanoseconde

Mémoires Caches
Koctets (L1,L2)
5 nanosecondes

Mémoires Centrales
Goctets
10 nanosecondes

Mémoires de masse
500 Goctets - Toctets
5 millisecondes

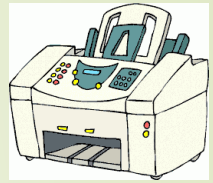
Au plus près du cpu — Capacité, vitesse — Au plus loin du cpu

Mémoire volatile : le contenu de la mémoire n'existe que si il y a une alimentation électrique (typiquement les mémoires caches et mémoire centrale)
Mémoire permanente, de masse : mémoire de grande capacité dont le contenu demeure même sans alimentation électrique (typiquement le disque dur)

Périphériques de l'ordinateur

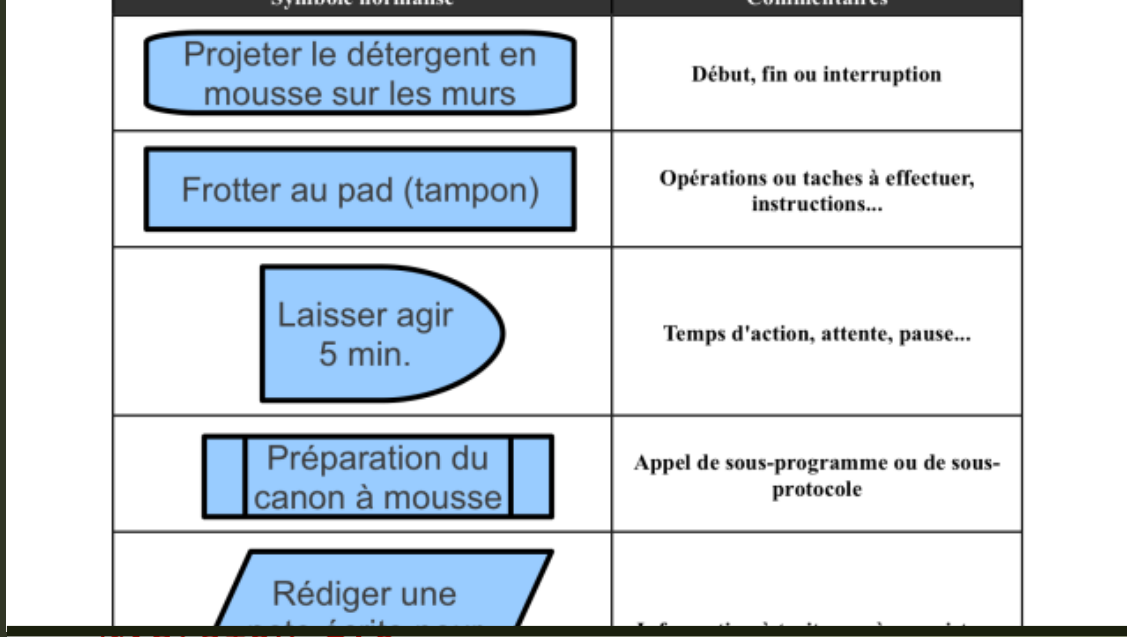
le cnam

Un périphérique est un matériel électronique pouvant être raccordé à un ordinateur par l'intermédiaire de l'une de ses **interfaces d'entrée-sortie** (interface VGA, HDMI, USB, RJ45.), le plus souvent par l'intermédiaire d'un **connecteur**. L'interface d'entrées-sorties est pilotée par un driver (pilote d'entrées-sorties)



- On distingue habituellement les catégories de périphériques suivantes :
  - **périphériques de sortie**: ce sont des périphériques permettant à l'ordinateur d'émettre des informations vers l'extérieur, tels qu'un écran, une imprimante..
  - **périphériques d'entrée** : ce sont des périphériques capables uniquement d'envoyer des informations à l'ordinateur, par exemple la souris, le clavier, etc.
  - **périphériques d'entrée-sortie** : ce sont des périphériques capables d'envoyer des informations à l'ordinateur et permettant également à l'ordinateur démettre des informations vers l'extérieur, par exemple le modem, le disque dur

Programmes, langages, types et niveaux de langage




```
 * @author john doe <doe.j@example.com>
 */
HelloButton()
{
  JButton hello = new JButton( "Hello, wor
  hello.addActionListener( new HelloBtnList

  // use the JFrame type until support for t
  // new component is finished
  JFrame frame = new JFrame( "Hello Button"
  Container pane = frame.getContentPane();
  pane.add( hello );
  frame.pack();
```

Programme, langage

le cnam

1.

J'ai un problème à résoudre !

b

a Périmètre ?

2.

J'écris une solution !

➔ ALGORITHME

Périmètre := 2a + 2b

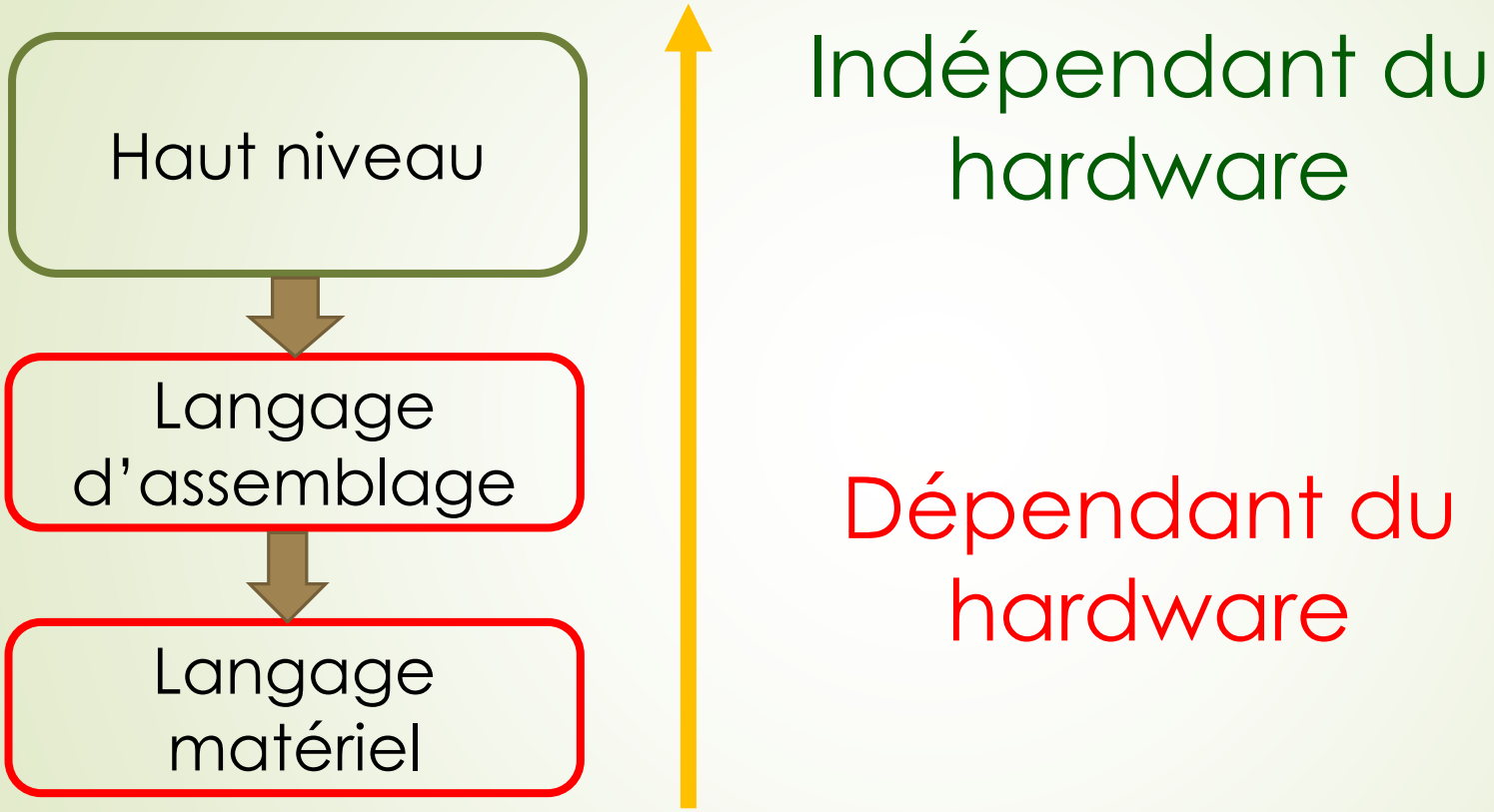
3.

En utilisant un langage de programmation, je code la solution pour la faire exécuter par l'ordinateur

➔ PROGRAMME constitué d'instructions

```
main(){
int a, b, perimetre;
        read (a); read (b);
        perimetre := (2 * a) + (2 * b);
        affiche (périmètre);
}
```


Trois niveaux de programmation

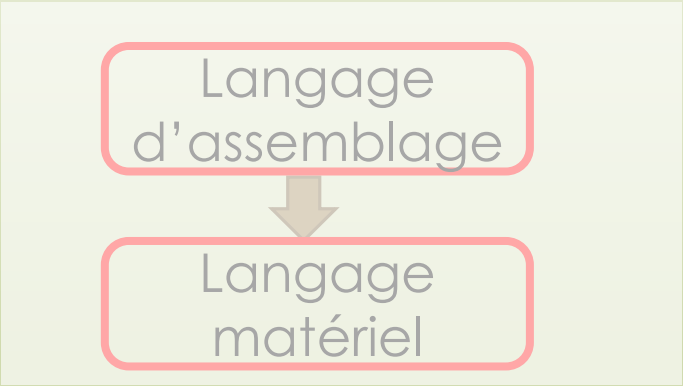


Programmation haut niveau

le cnam

Haut niveau

```
i = 2; j = 3;
while (i<2){j = j +i;}
```



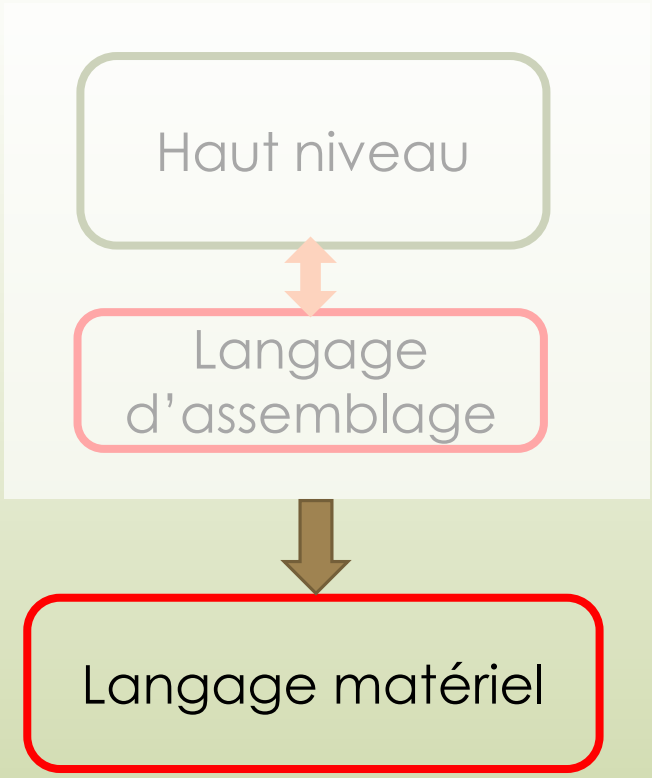
Instructions de haut niveau :

- Proches du raisonnement humain
- Indépendantes de la machine

→ Facilité de programmation

- Fortran (1954)
- Cobol (1959)
- C (1970)

Programmation en langage machine

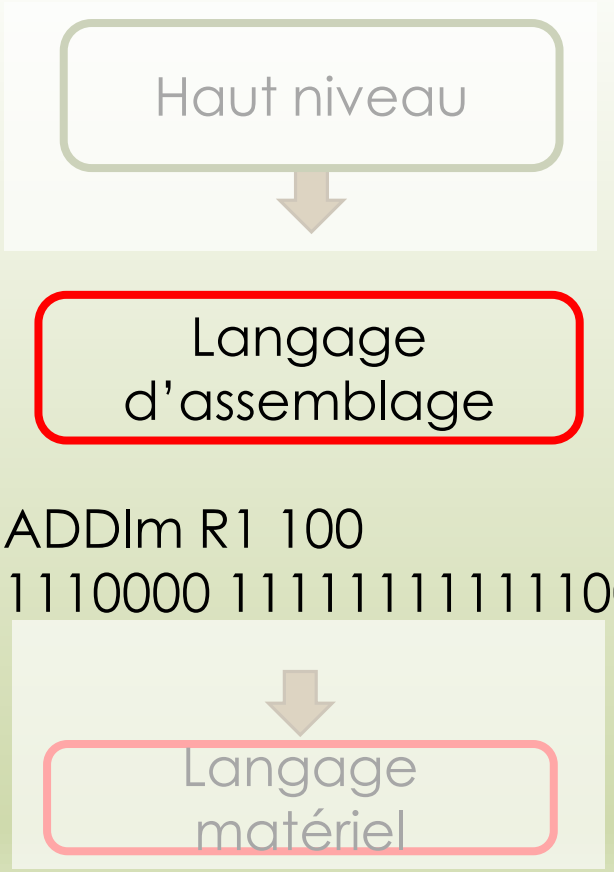


Instructions de bas niveau :

- Jeu d'instructions du processeur
- Dépendantes de la machine
- Chaines binaires

→ Programmation laborieuse des premiers temps de l'informatique

Programmation en langage d'assemblage

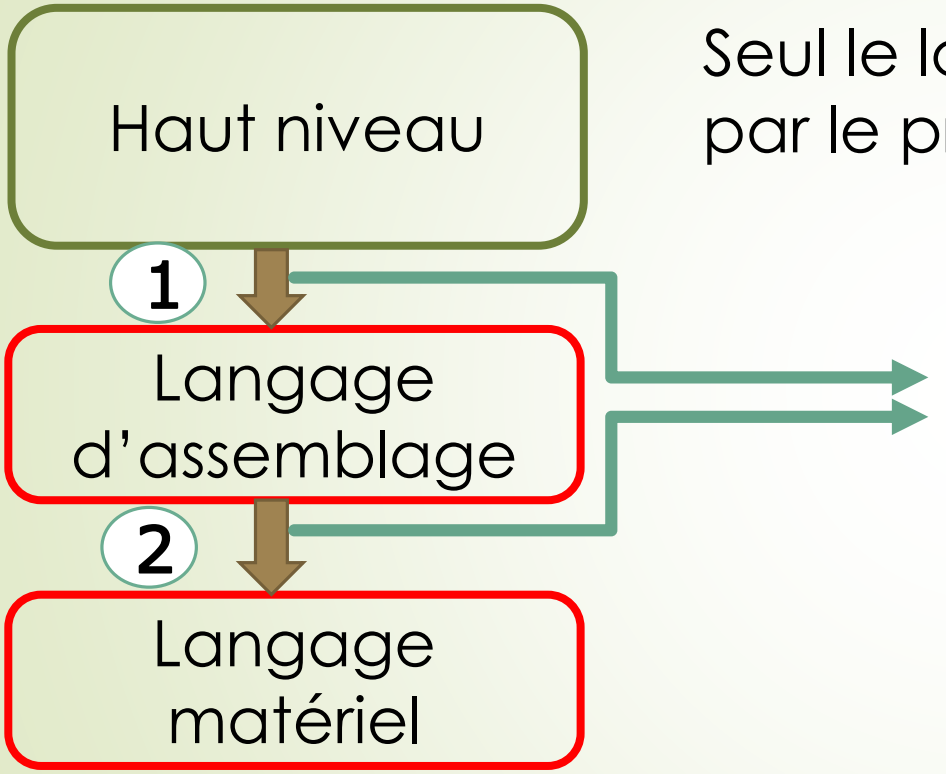


Instructions de bas niveau :

- Liées au jeu d'instructions du processeur
- Dépendantes de la machine
- Mnémoniques remplaçant les chaines binaires

→ Programmation de bas niveau
→ Premier langage remplaçant la programmation binaire (1949)

Traducteurs entre niveaux



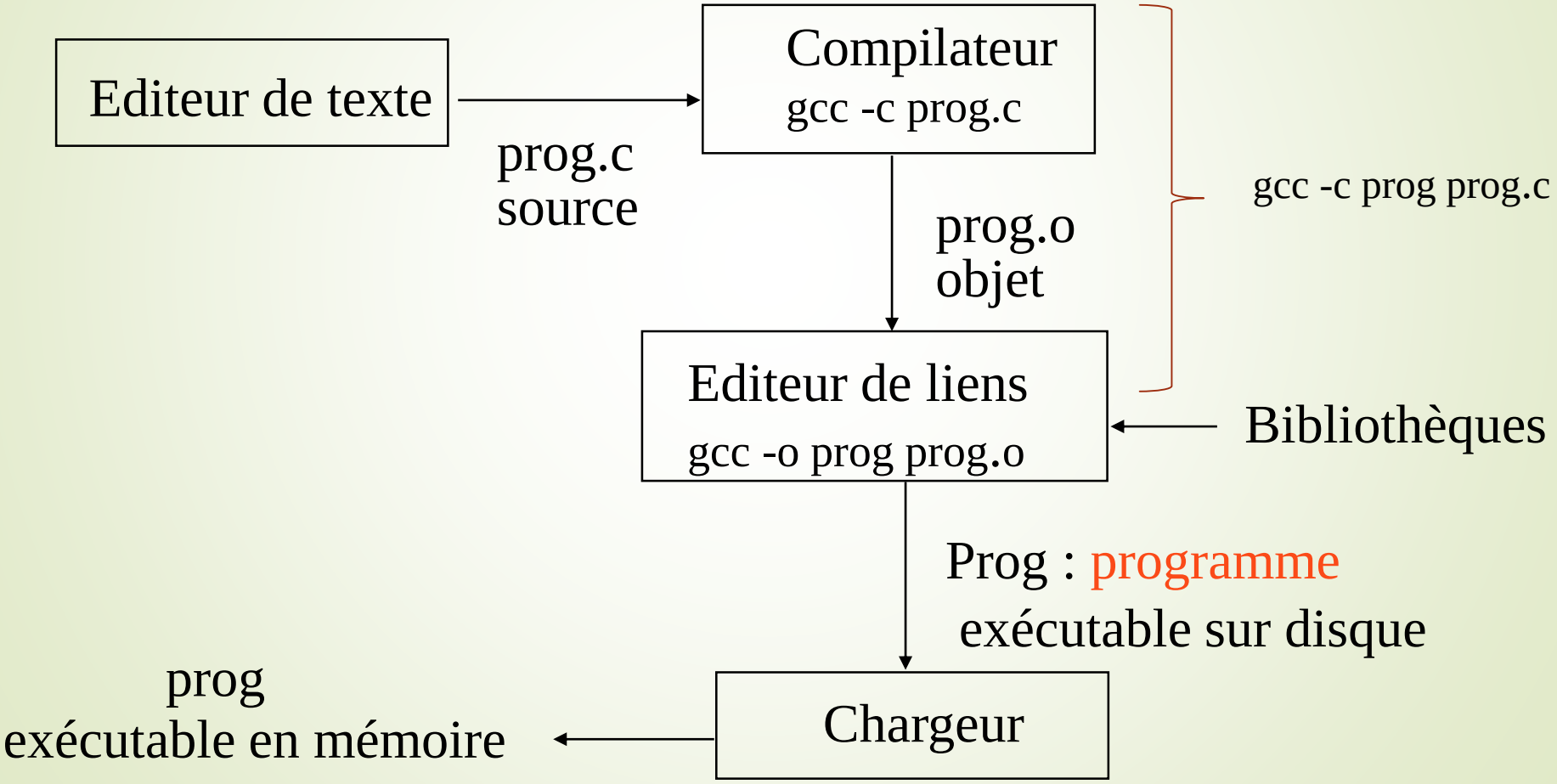
Seul le langage matériel est exécutable par le processeur

Traducteurs

1. Compilateur
2. Assembleur

le cnam

Construire mon premier langage en C



le cnam

Construire mon premier langage en C

Programme principal essai.c

```c
#include <stdio.h>
#include "util.h"

main(){

affiche_message("bonjour, nous sommes lundi matin ");

}
```

stdio.h

util.h

```c
int lire entier(){
float lire flottant(){
char lire car(){
void affiche entier(int i){
void affiche flottant (float f){
void affiche car (char c){
void affiche message (char *s){
void affiche car sautligne (char c){
void affiche entier sautligne(int i){
void affiche flottant sautligne (float f){
void ajoutesautdeligne() {
```

Construire mon premier langage en C