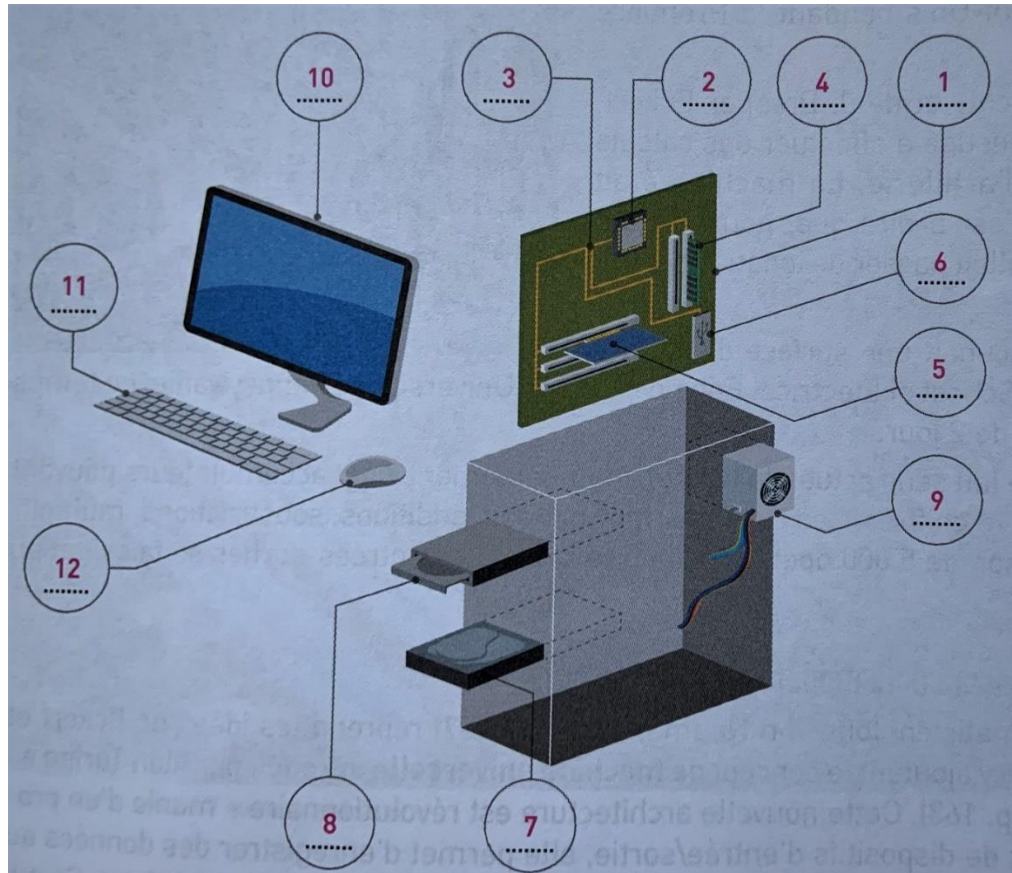


---

*Architectures Von Neumann  
&  
Organisation de la mémoire*

---

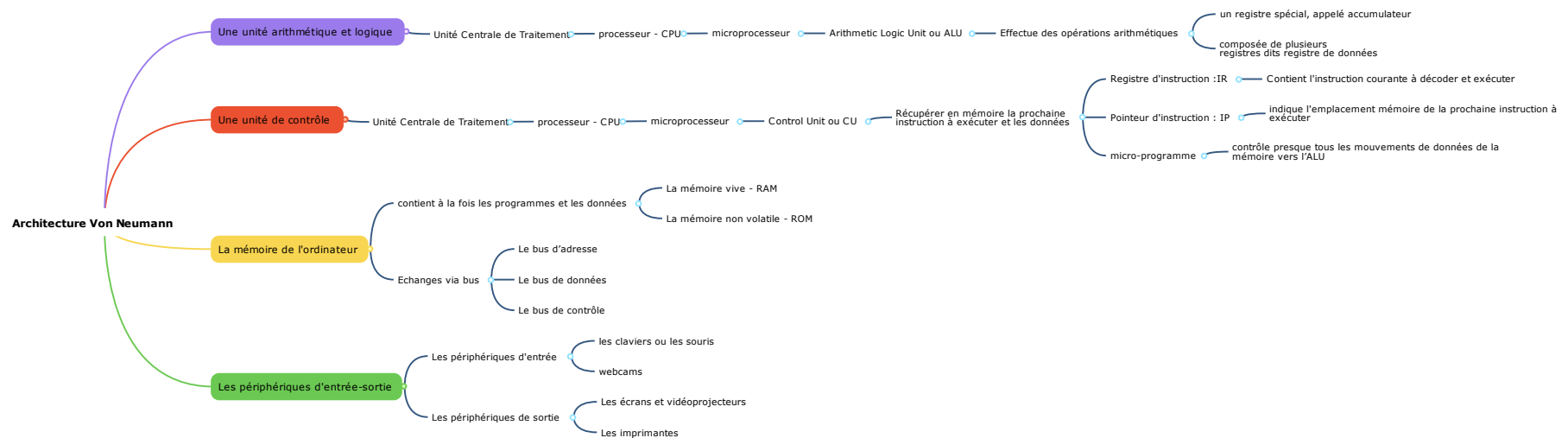
**Exercice 1 :** Compléter le schéma des principaux composants de l'ordinateur.



1. La carte mère
2. Le microprocesseur
3. Le bus

4. La mémoire vive
5. La carte graphique
6. Les ports d'entrée-sortie
7. Le disque dur
8. Le lecteur/graveur de CD/DVD
9. L'alimentation électrique
10. L'écran
11. Le clavier
12. La souris

**Exercice 2 :** Construire une carte mentale détaillée des éléments constitutifs de l'Architecture Von Neumann.



### Exercice 3

Pour chaque question, 4 propositions de réponses sont faites. Une seule est correcte. Laquelle ?

1. Un processeur peut atteindre 3,5 GHz. Cela correspond à combien de cycles d'horloge par seconde ?  
**a) 3 500 000 000**  
b) 3 500 000  
c) 3 500 000 000 000  
d) 3 500
  
2. Dans l'architecture de Von Neumann, l'UAL est généralement représentée par :  
a) Un « N »  
**b) Un « V »**  
c) Un « C »  
d) Un « U »
  
3. Quel est le rôle de l'Unité Arithmétique et Logique dans un processeur ?  
a) La gestion interne du processeur  
**b) Faire les calculs**  
c) Interfacer les différents périphériques  
d) Définir la base arithmétique
  
4. Parmi les éléments suivants, lequel n'est pas un périphérique d'entrée sortie ?  
a) La souris  
b) L'imprimante  
c) Le clavier  
**d) Le transistor**

5. La mémoire RAM ?

- a) Ne fonctionne qu'en mode lecture
- b) Ne fonctionne qu'en mode écriture
- c) Conserve les données en cas de coupure de l'alimentation
- d) **Perd les données en cas de coupure de l'alimentation**

6. A quoi sert l'horloge du processeur ?
- a) Elle séquence les instructions à exécuter.**
  - b) Elle donne l'heure exacte.
  - c) Elle envoie des ordres de lecture de données.
  - d) Elle décide des instructions à exécuter.
7. Dans l'architecture de Von Neumann, le processeur contient :
- a) L'unité de commande et l'unité arithmétique et logique**
  - b) L'unité de commande et la RAM
  - c) Seulement l'unité arithmétique et logique.
  - d) Seulement l'unité de commande
8. Parmi les 4 composants informatiques ci-dessous, lequel est le plus rapide ?
- a) RAM
  - b) Disque dur
  - c) Mémoire cache
  - d) Registre**
9. Pour stocker les résultats intermédiaires, l'ALU utilise ...
- a) L'accumulateur**
  - b) Le tas
  - c) La pile
  - d) Les registres

10. Le processeur est composé ...

- a) De l'unité de contrôle et de la mémoire principale
- b) D'une mémoire principale et de l'unité arithmétique et logique
- c) **De l'unité de contrôle et de l'unité arithmétique et logique**
- d) D'un accumulateur et d'entrées/sorties

11. Le composant du CPU chargé de comparer le contenu de deux données est ...
- a) L'unité arithmétique et logique**
  - b) L'unité de contrôle
  - c) La mémoire
  - d) Les registres
12. L'unité qui permet de séquencer, décoder, traduire chaque instruction et générer les signaux d'activation nécessaires pour l'ALU et d'autres unités s'appelle ...
- a) L'unité arithmétique
  - b) Le CPU
  - c) L'unité logique
  - d) L'unité de contrôle**
13. Physiquement, les registres sont :
- a) Des composants sur la carte mère de l'ordinateur
  - b) A l'intérieur du processeur**
  - c) Une partie séparée de la RAM
  - d) De la mémoire cache



**Exercice 4** : Gestion du segment de pile - **Compléter les textes à trous, puis compléter le schéma représentant la pile** de la feuille d'exercices (Exercices Von Neumann).

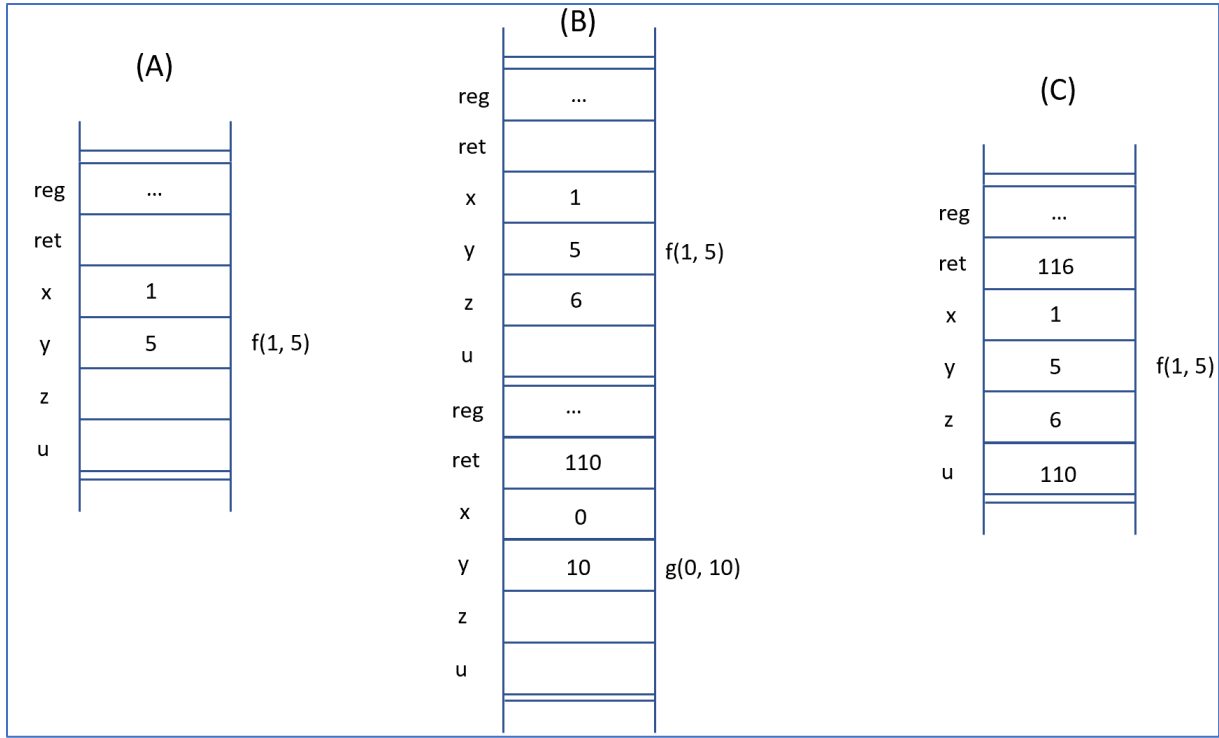
Considérons les 2 fonctions suivantes et observons la gestion de la pile lors de l'appel **f(1, 5)** :

```
def g(x, y) :  
    return 100 + y  
  
def f(x, y) :  
    z = x + y  
    u = g(x - 1, y * 2)  
    return z + u
```

La pile (A) représente l'environnement lors de l'appel à f(1, 5). Les premières cases mémoires contiennent respectivement une sauvegarde des registres du microprocesseur (reg), un espace pour la valeur de retour de la fonction (ret), les valeurs des paramètres x (1) et y (5), ainsi qu'un espace pour la variable locale z (qui pour le moment ne contient aucune valeur).

La pile (B) représente l'environnement lors de l'appel g(x - 1, y \* 2). Pour réaliser cet appel, un nouvel environnement est alloué sur la pile avec une sauvegarde des registres et les valeurs des arguments x(0) et y (10) de g. Cet appel se termine en renvoyant la valeur 100 + y (110) qui est stockée dans son espace de retour (ret).

La pile (C) montre enfin l'environnement de f(1, 5) après le retour de g(0, 10). On voit que l'espace alloué pour l'appel à g a été supprimé de la pile, que la valeur de retour (110) a été récupérée et stockée dans la case mémoire de la variable locale u et que la somme z + u (116) est stockée dans l'espace de retour.



Gestion du segment de pile