

Séance 8

Algorithmes

[Algorithme producteur/consommateur : principe](#)

[Algorithme producteur/consommateur : synchronisation par sémaphores](#)

Algorithme producteur/consommateur : principe

Producteur : un thread qui dépose, à son rythme, des données dans un tampon mémoire.

Consommateur : un thread qui extrait ces données à son rythme.

Algorithme Producteur :

De manière répétitive

1. Attendre place libre dans tampon
2. Déposer valeur dans tampon

débloque attente Consommateur



Algorithme Consommateur :

De manière répétitive

1. Attendre valeur présente dans tampon
2. Retirer valeur du tampon

débloque attente Producteur



Algorithme producteur/consommateur : synchronisation par sémaphores

Un sémaphore pour gérer attente du producteur (places libres dans tampon) : **semP**

Valeur initiale : taille du tampon

Un sémaphore pour gérer attente du consommateur (valeur présente dans tampon) : **semC**

Valeur initiale : 0

Algorithme Producteur :

De manière répétitive

1. **-1 sur semP** (=> attente si sémaphore nul)
2. Déposer valeur dans tampon
3. **+ 1 sur semC**

Algorithme Consommateur :

De manière répétitive

1. **-1 sur semC** (=> attente si sémaphore nul)
2. Retirer valeur du tampon
3. **+1 sur semP**