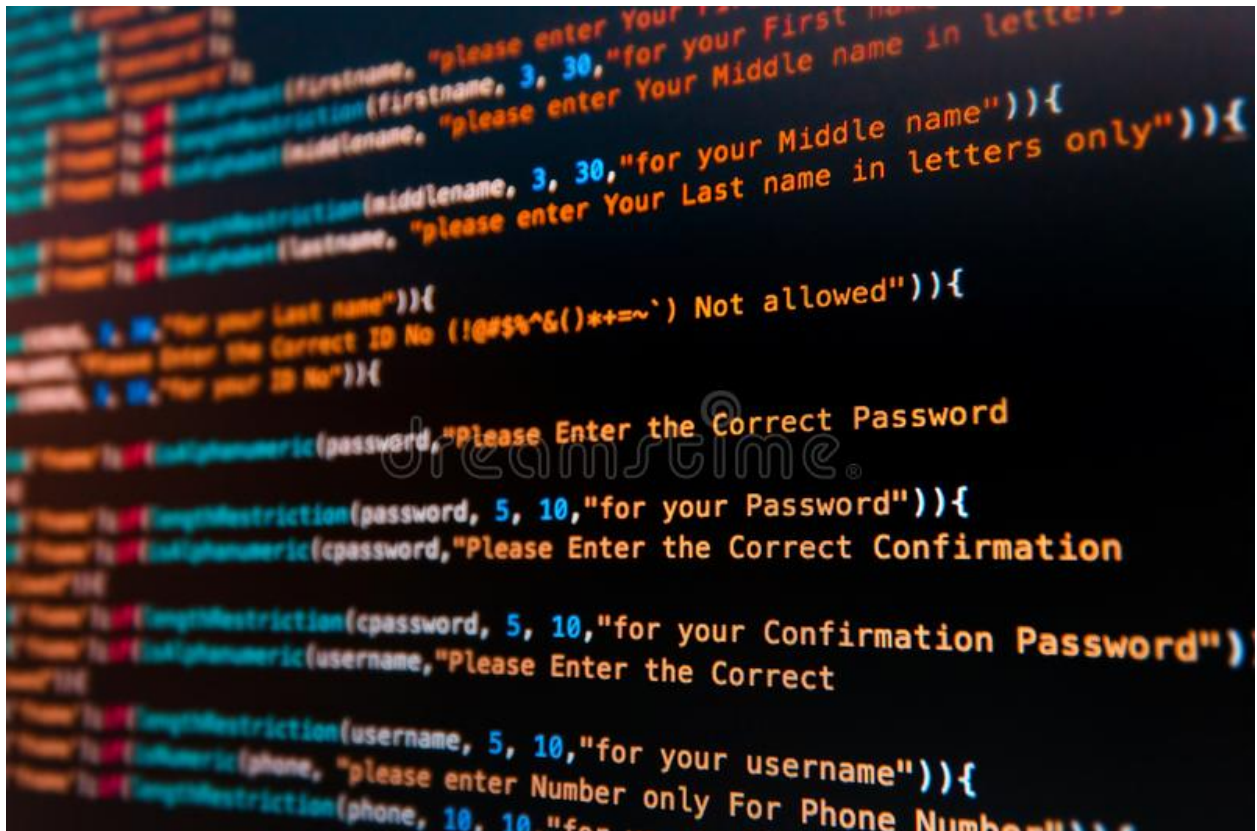


TP4 : Problème des lecteurs/rédacteurs



Hamza Mouddene

7 novembre 2020

1 Première implantation : Priorité aux rédacteurs

La première implantation consiste à donner la priorité aux rédacteurs, en utilisant un moniteur, deux conditions associés au moniteur, un boolean pour savoir s'il un rédacteur est en cours d'écriture, un entier pour connaître le nombre de rédacteur en attente et un compteur pour stocker le nombre de lecteurs actifs.

Un lecteur peut lire que s'il ne y'a ni de rédacteurs en attente ni d'écriture active, ainsi qu'un lecteur finit la lecture puis libère un rédacteur puisqu'on est sur une priorité aux rédacteurs. Un rédacteur peut commencer à écrire que s'il ne y'a ni d'écriture ni de lecture active, ainsi qu'un rédacteur termine d'écrire puis s'il ne y'a pas d'écriture active, il libère un lecteur.

2 Deuxième implantation : Priorité aux lecteurs

La deuxième implantation consiste à donner la priorité aux lecteurs, en utilisant un moniteur, deux conditions associés au moniteur, un boolean pour savoir s'il un rédacteur est en cours d'écriture, un entier pour connaître le nombre de rédacteur en attente et un compteur pour stocker le nombre de lecteurs actifs.

Un lecteur peut lire que s'il ne y'a pas d'écriture active, ainsi qu'un lecteur finit la lecture puis libère un rédacteur s'il ne y'a pas de lecteur actif. Un rédacteur peut commencer à écrire que s'il ne y'a ni d'écriture ni de lecture active, ainsi qu'un rédacteur termine d'écrire puis il libère un lecteur.

3 Troisième implantation : Équité

La troisième implantation consiste à faire une version équitable afin d'éviter la famine que ce soient des lecteurs ou des rédacteurs, en utilisant un moniteur, deux conditions associés au moniteur et deux compteurs pour stocker le nombre de lecteurs et de rédacteurs actifs.

Un lecteur peut lire que s'il ne y'a ni d'écriture active ni de lecteurs ou rédacteurs en attente, ainsi qu'un lecteur finit la lecture puis libère un rédacteur s'il ne y'a ni de lecteurs actifs ni d'écrivain en attente, sinon il libère lecteur. Un rédacteur peut commencer à écrire que s'il ne y'a ni lecture active ni d'écriture active, quand un rédacteur finit d'écrire il libère un lecteur. La particularité de cette version est que quand un rédacteur demande d'écrire et il satisfait pas la condition précédement citée on bloque un lecteur puis on vérifie que s'il y'a des lecteurs actifs on bloque le lecteur.

4 Conclusion

Les deux premières versions ont une limite, si on augment le nombre de redacteurs (resp. lecteurs) pour la strategie priorité aux rédacteurs (resp. priorité aux lecteurs), la partie non prioritaire est condamné à la famine, contrairement à la version équitable.

J'ai utilisé *getWaitQueueLength(Condition condition)* pour avoir le nombre de processus en attente correspondant à la condition associée au moniteur.