1.Les structures de données utilisées dans le code sont producteur et consommateur.  
2. L'accès aux données partagées (comme buffer et count) est protégé en utilisant des mécanismes de synchronisation, notamment des verrous (mutex) et des variables de condition (condition variables). Les mutex garantissent qu'une seule thread peut accéder à la fois à certaines parties critiques du code, tandis que les variables de condition permettent aux threads de signaler et d'attendre des événements spécifiques.  
3.Les risques incluent les conditions de concurrence où plusieurs threads peuvent tenter d'accéder/modifier les mêmes données simultanément, ce qui peut causer des résultats imprévisibles ou incorrects. Les problèmes comme les deadlocks (blocages) peuvent survenir si les threads attendent indéfiniment les uns les autres. De plus, les problèmes de performance peuvent émerger si la synchronisation est trop fine et entraîne des frais généraux excessifs