



Q1 : Le niveau de documentation des classes est-il approprié par rapport à leur complexité ? NOC, WMC

Q2 : La conception est-elle bien modulaire ?

Q3 : Le code est-il mature ? CBO, Temp total de l'exécution

Q4 : Le code peut-il être testé bien automatiquement ? TPC, NEC

– NOC (Nombre d'enfants) est le nombre de sous-classes directes subordonnées à la classe dans la hiérarchie des classes

Pour Q3, nous avons sélectionné deux métriques, CBO et Temps total de l'exécution.

1. CBO fait référence à "Coupling Between Objects", nous savons que plus une classe est couplée à d'autres classes, plus une modification de cette classe peut influencer de classes. Ensuite, CBO a joué un grand rôle en testant si le code est mature, si ce code peut avoir un CBO inférieur, la charge de travail des tests sera également réduite, ce qui montre déguisé que ce code a été modifié plusieurs fois et est relativement mature. Ensuite, cela n'affectera pas plusieurs aspects en raison de tâches de test fastidieuses.

2. Pour Temps total de l'exécution, lorsque nous testons, le résultat est très lent. Tout le monde veut obtenir une réponse le plus rapidement possible, ensuite si le temps d'exécution est lent, le code est moins mature et moins adapté aux besoins des gens.

Pour le Q4, nous avons choisi deux métriques, TPC et NEC.

1. Pour TPC, nous testons en fonction de la classe, donc au lieu de tester directement l'intégralité du code, nous pouvons juger si chaque partie peut être exécutée correctement grâce à des tests automatiques. S'il y a un bogue dans l'un d'entre eux, alors lorsque le code global est automatiquement testé, certains bogues seront trouvés.
2. Pour NEC, il est évident que si un code a beaucoup d'erreurs après son exécution, alors le code doit avoir une erreur dans un certain lien. Lors du test automatique, il a dû rencontrer des problèmes, ensuite, ce code ne fonctionne pas bien dans les tests automatiques.