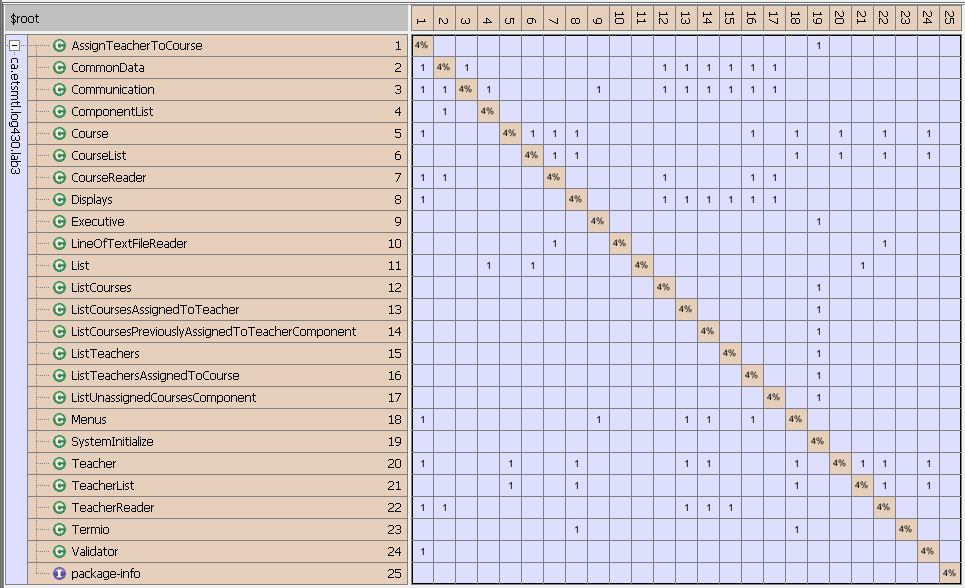
b)

Pour faire des modifications au code d’un projet dont le style d’architecture est l’invocation implicite, il faut bien comprendre le fonctionnement du patron de conception logiciel « Observer ». Dans notre cas, la classe qui est « Observable » est la classe « Communication ». Cette classe sert à écouter pour enregistrer n’importe quelle modification de l’état de l’application. Lorsqu’une modification est détectée, un message est envoyé à tous les composants qui se sont enregistré à la liste des notifications. Selon la nature de l’événement, les composants impliqués lanceront les fonctions correspondantes à la logique d’affaire de l’application.

Pour respecter l’architecture d’invocation implicite, toutes nos modifications au code existant devaient respecter ce modèle. Pour ce faire, nous avons donc

c)

La matrice DSM du système original :



La matrice DSM du système modifié :

