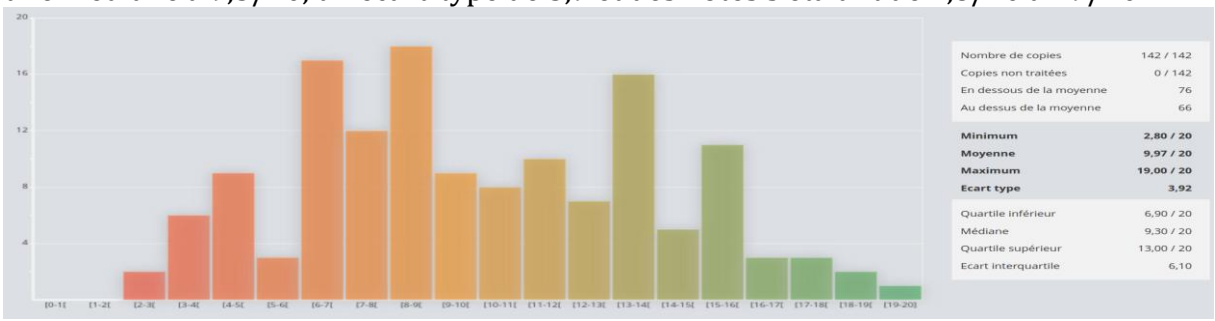
L'épreuve portait sur la résolution d'un problème célèbre : le *problème du voyageur de commerce* et de sa résolution par un *algorithme génétique*.
Après avoir remarqué l'impossibilité d'une résolution exacte en des temps raisonnables par des algorithmes dits de force brute, ou naïfs, on construit une méthode efficace de résolution, approchée, et en temps raisonnable, par une approche dite *métaheuristique* et connue sous le nom *d'algorithme génétique*.
Cette classe de méthode, inspirée des processus biologiques mis en œuvre au niveau du génome dans la théorie de l'évolution, est basée sur l'évolution d'une population, génération après génération, par croisements, mutations, puis sélection, afin de mieux en mieux satisfaire à un critère d'évolution. Elle est utilisée pour la résolution de problèmes (NP)-difficiles, dont le problème du voyageur de commerce est un bon exemple.

Le déroulé de l'épreuve est satisfaisant : malgré sa longueur, de nombreux candidats sont parvenus à terminer le sujet, et les statistiques donnent, une moyenne proche de 10/20, une médiane à 9,3/20, un écart-type de 3,9 et des notes s'étalant de 2,8/20 à 19/20.



Comme habituellement le niveau des candidats en algorithme est très variable ; dénotant une pratique de la programmation très inégale. Le niveau des candidats est en moyenne celui attendu. On pourrait juste déplorer que les évolutions récentes du programme officiel ne soient pas toujours bien digérées, notamment la structure de donnée des dictionnaires (ou collections) pourtant essentielle car incontournable pour la résolution algorithmique de problèmes complexes.

Dans le détail :
La première partie portait sur la compréhension du problème de voyageur de commerce à l'aide d'un exemple. Ne nécessitant pas de code, mais seulement la compréhension de la problématique, cette première partie a été bien réussie.
La seconde partie portait sur l'implémentation des données. Elle a été assez bien réussie. La représentation du problème à l'aide d'une matrice des distances inter-villes a été bien comprise (Q.3). La complétion du code de la fonction qui renvoyait la longueur d'un parcours (Q.4.) a été souvent erronée à cause de problèmes d'indiçage. Dans la constitution d'une population initiale au sein d'un dictionnaire, la fait que la clé d'un dictionnaire ne puisse être que de type non-mutable (entier, flottant, chaine de caractères…) excluant ainsi les listes (réponse attendu Q.5.1) est mal su des candidats. Par contre le phénomène d'aliasing pour les listes (Q.5.2) est connu de nombreux candidats.
Dans la troisième et dernière partie on implémentait l'algorithme génétique.
Cette partie n'a été bien réussie que par les meilleurs candidats. Le bon code du tri par insertion croissant (Q.6) n'a été trouvé que par une minorité de candidats. Nombreux ont reconnu à la place celui du tri décroissant.

Les complétions des codes pour la sélection d'une population de reproducteurs, la recherche d'une tournée minimale (recherche de min), le croisement, et la mutation (échange de 2 variables) ont donné lieu à une réussite variable selon que le candidat avait ou non gardé le fil de l'implémentation. La dernière question qui demandait de compléter le code de l'algorithme par des instructions fournies a été relativement bien réussie.

En résumé, cette année encore, les candidats sont parvenus dans l'ensemble et en un temps très court (45') à s'accaparer une problématique complexe, et à implémenter une résolution délicate. Le niveau démontré par les candidats a été jugé globalement satisfaisant et conforme aux attentes. Par contre le maniement des dictionnaires demeure chez certains candidats encore assez perfectible.