

Exercice 1

Question 1

Réponse : `int quotient_par_soustraction(int a, int b)`

- 1 pt si juste et au bon endroit
- Enlever 0,5 pt par erreur de type (double au lieu de int, void au lieu de int pour le retour) pas de point en moins sur les noms des variables
- Enlever 0,5 pt si pas au bon endroit dans le programme

Question 2

Réponse : La compilation fonctionne mais l'édition de lien échoue car la fonction `quotient...` n'a pas de définition

- 1 point
- 0,25 point si pas de justification

Question 3 sur 2 pt

- Si la méthode emploie la division 0pt
- S'il n'y a pas de boucle ou trop de boucles : 0pt
- Si c'est par une autre méthode que la soustraction (sans employer la division) : on ne donne que 0,5 pt.
- Si exactement une boucle et méthode par soustraction et le bon quotient est trouvé (à des erreurs syntaxiques marginales, type oubli de point virgule) : 2 points (pas de points en moins si c'est un `for ()` dans cet exo)
- Si il manque un élément structurel important autre que la boucle pour que ça fonctionne (valeur de retour, condition de boucle) on enlève 1 pt
- Si le quotient trouvé est $q - 1$ ou $q + 1$: on enlève 0,25 pt
- Si la valeur retournée n'est pas déclarée : on enlève 0,25 pt

Exercice 2.1 sur 3,5 pt

- Si squelette de programme correct : 0,25pt
- Si `define TAILLE 4` (ou autre entier) : 0,25 pt
- Si ça ne fonctionne pas du tout, mais que la structure du programme respecte : boucle de saisie, saisie, boucle de comptage, on donne 1pt.
- Sinon
- 1,5 pt pour la saisie des entiers (affichage, `scanf` dans boucle `for`, affichage, `scanf`)
- 1,5 pt pour le nombre d'occurrences calculé par une boucle `for`
- On enlève 0,5 pt par boucle `while` au lieu de `for`
- Si le nombre d'occurrences est calculé à un près on enlève 0,25pt
- Si les déclarations sont fausses ou inexistantes : on enlève 0,5pt si ça concerne le tableau (et d'autres variables), 0,25 pt si une seule variable autre que le tableau
- Si il y a une erreur de syntaxe importante dans un `scanf` : on enlève 0,5pt
- Si il y a une erreur de syntaxe importante dans le `printf` du résultat : on enlève 0,5pt

Exercice 2.2 sur 3,5 pt

- Si squelette de programme correct : 0,25pt
- Si il n’y a pas de boucle ou trop de boucles 0pt pour le reste
- Si utilise un tableau et fonctionne : seulement 1 pt
- Si ça ne fonctionne pas du tout, on note la structure : c’est une boucle (0,5pt) dont le corps fait de la saisie (0,5pt) et du comptage avec if (0,5pt).
- Si le programme fait ce qui est demandé avec une unique boucle on part sur 3,25pt
- Si les déclarations sont fausses ou inexistantes : on enlève 0,5pt si ça concerne plusieurs variables, 0,25 pt si une seule variable.
- Si la boucle n’est ni un while ni un do while (donc un for) : on enlève 0,5pt
- Si la condition de boucle est fausse on enlève 1pt ou 0,5 pt si c’est presque bon
- Pas de comptage on enlève 1,5.
- Si les résultats du comptage sont faux, on enlève 0,5pt
- Si il y a une erreur de syntaxe importante dans un scanf : on enlève 0,5pt
- Si il y a une erreur de syntaxe importante dans le printf du résultat : on enlève 0,5pt

Exercice 3 sur 4 pt

- 2 pt pour le tableau du main, 2pt pour le tableau de puissance(4, 3)
- On enlève 1 point si le for ne se termine pas sur un i valant 3.
- On enlève 0,5 pt par colonne ayant une erreur : numéro de ligne faux, variable avec une mauvaise valeur ou variable oubliée, affichage erroné, ...
- On enlève 0,5 pt si la valeur de retour est incorrecte.

Exercice 4 sur 5 pt

- Un point par exercice tout juste. On admet les prototypes avec seulement n comme avec n et m aux deux dernières questions.
- 0,25 si seul le prototype de la fonction est juste.
- si le prototype est faux : on mets 0 et on arrête là pour les trois premières questions on note sur 0,5 pt maximum les deux dernières.
- Pour chacune des deux dernières question, si le calcul est faux mais qu’il y a deux for imbriqués on ajoute 0,25pt.