

Rapport de Correction - Examen ASD S1 2025-2026

Fichier Source : doc20260115222526 .pdf **Nombre de Copies :** 24 (La copie N°23 est manquante dans le document source)

Analyse :**Algorithmique :**

- Boucles imbriquées `while (T != S) { while (i <= N) { ... } ... }`. Structure très lourde et incorrecte (boucle infinie si `T != S`).
- Boucle interne `for (J=0; J<Z; J++)` : Itère sur `Z` (valeur lue) ? Semble simuler quelque chose d'autre que la présence (peut-être un tableau de présences par séance ?).
- Logique complexe et confuse.

Notation :

Critère	Points	Commentaire
Lecture N, A, S	3 / 3	OK.
Initialisation	1 / 3	Beaucoup de variables inutiles.
Condition boucle	1 / 4	Boucles imbriquées inutiles et dangereuses.
Logique prés./abs.	2 / 4	Conditions présentes mais noyées dans une mauvaise structure.
Compteurs	2 / 3	OK.
Affichages inter.	0 / 2	Confus.
Affichage final	1 / 1	OK.

NOTE FINALE : 10 / 20**Feedback :**

- **Appréciation globale : Moyen -.** Pourquoi tant de boucles ? Restez simple.
-

Analyse :

Algorithmique :

- Boucle `for` simple.
- Logique interne correcte.
- Condition d'arrêt `if (N == i && N < S)` ? Test d'arrêt incorrect (`i` est l'indice, il vaudra `N` à la fin).
- Code de 'Stop Simulation' demande à l'utilisateur de resaisir `N`, `i`, `S` ? Contresens total.

NOTE FINALE : 12 / 20

Feedback :

- **Appréciation globale : Moyen.** Bonne structure mais fin incohérente (ressaisie des données).
-

Analyse :

Algorithmique :

- Boucle `for (... | | i != S)`. Condition arrêt fausse (| | continue si $i \neq S$, donc tant que $i \neq S$ la boucle tourne même si $i > N$).
- Compteurs : Initialise $C = N$. Décrémente si absent ($C = C - 1$). $M = N - C$ (absents).
- Logique cohérente (compte à rebours des présents).

NOTE FINALE : 14 / 20

Feedback :

- **Appréciation globale : Moyen / Bon.** Approche originale (compte à rebours).
-

Analyse :

Algorithmique :

- `int B` non initialisé.
- Boucle `for`.
- Calculs finaux : $B = B + 1$ (?? Une seule fois hors boucle). $n = N - B$.
- Les compteurs ne sont pas mis à jour **DANS** la boucle. Donc `B` ne compte rien (valeur poubelle + 1).

NOTE FINALE : 06 / 20

Feedback :

- **Appréciation globale : Insuffisant.** Les compteurs doivent être dans la boucle.
-

Analyse :

Algorithmique :

- Lit `x` (une seule fois) **avant** la boucle (dans `scanf ("%d", &x)` pour A ??). Confusion totale.
- Structure `if/else` avant la boucle.
- Boucle `while` infinie (`i` incrémenté une fois avant).

NOTE FINALE : 04 / 20

Feedback :

- **Appréciation globale : Très Insuffisant.**
-

Analyse :

Algorithmique :

- Même erreur : Lit `x` une seule fois avant la boucle.
- Traite le cas présent/absent une fois.
- Boucle `while` qui répète le résultat final ?
- `return 0` dans la boucle.

NOTE FINALE : 05 / 20

Feedback :

- **Appréciation globale : Insuffisant.** Ne traite qu'un seul étudiant.
-

Analyse :

Algorithmique :

- Boucle `while (N != 0 && K != S)`. `N` est constant, donc `N!=0` toujours vrai. Arrêt sur `K (présents) == S (seuil absents) ??` Confusion variables.
- Pas de compteur de boucle (boucle infinie, sauf si `K` atteint `S`).
- Logique interne correcte.

NOTE FINALE : 09 / 20

Feedback :

- **Appréciation globale : Fragile.** Boucle potentiellement infinie. Confusion sur la condition d'arrêt.
-

Analyse :

Algorithmique :

- Boucle `for` OK.
- Logique interne OK.
- Arrêt : `if (Y == S) i = N.` (Force la fin de boucle). Correct.
- Variables non initialisées `Y`, `E` (accumulateurs). Grave en C.

NOTE FINALE : 13 / 20

Feedback :

- **Appréciation globale : Moyen.** Attention à l'initialisation.
-

Analyse :

Algorithmique :

- Boucle `for (N = 0; N >= T)`. `N` (nombre étudiants) écrasé par 0. `T = N` initial.
- Condition `N >= T` (`0 >= N`) : Faux dès le début (sauf si `N=0`). La boucle ne s'exécute pas.
- Logique interne : `i == 0` (comparaison inutile), `i++` (sur variable non init).

NOTE FINALE : 03 / 20

Feedback :

- **Appréciation globale : Très Insuffisant.** Boucle ne démarre pas.
-

Analyse :

Algorithmique :

- Incrémentation `C = C + A` (si absent ?). Bizarre.
- Condition arrêt `if (S == S || ...)` : Toujours vrai.
- Pseudo-code.

NOTE FINALE : 05 / 20

Feedback :

- **Appréciation globale : Insuffisant.**
-

Analyse :

Algorithmique :

- Lecture `x` sans `&`. (`scanf ("%d" , x) -> Crash`).
- Variables `m`, `n` non initialisées.
- Structure correcte.

NOTE FINALE : 08 / 20

Feedback :

- **Appréciation globale : Fragile.** Erreur pointeur `scanf`.
-

Analyse :

Algorithmique :

- `while (i <= N || a == S). (|| continue si a == S).`
- Comparaisons pseudo-code `counter1 == p.` (Comparaison sans effet).
- Incrémentation `counter2 += i.` Ajoute l'indice étudiant ??
- Confus.

NOTE FINALE : 07 / 20

Feedback :

- **Appréciation globale : Insuffisant.**
-

Analyse :

Algorithmique :

- Commentaire `// Switch case`.
- Boucle `while`. `i` non initialisé. `i` jamais incrémenté (boucle infinie).
- Code logique interne correct.

NOTE FINALE : 06 / 20

Feedback :

- **Appréciation globale : Insuffisant.** Boucle infinie.
-

Analyse :

Algorithmique :

- Boucle `while` correcte (`absent_a != S`).
- Logique correcte.
- Affichage propre.
- Petit souci : `i` n'est incrémenté nulle part. Boucle infinie sur le 1er étudiant.

NOTE FINALE : 12 / 20

Feedback :

- **Appréciation globale : Moyen.** Oubli d'incrémenter le compteur de boucle.
-

Analyse :

Algorithmique :

- Double boucle `for` (étudiants) + `while` (tant que $Z < S$??).
- Le `while` à l'intérieur demande de saisir des sessions tant que le seuil global d'absence n'est pas atteint ?
- Confusion entre seuil global et saisie individuelle.
- Si un étudiant a $X > A$, \forall augmente. Le `while` ne s'arrête que si $Z < S$. Si l'étudiant est présent, Z ne change pas -> Boucle infinie sur la saisie du même étudiant ?

NOTE FINALE : 08 / 20

Feedback :

- **Appréciation globale : Fragile.** Risque de boucle infinie.
-

Analyse :

Algorithmique :

- Lecture groupée bizarre `scanf(..., &i...)`. Écrase `i`.
- Boucle `for`.
- Ne lit pas `x` dans la boucle. Utilise la valeur lue au début (n'importe quoi).

NOTE FINALE : 06 / 20

Feedback :

- **Appréciation globale : Insuffisant.**
-

Analyse :

Algorithmique :

- Boucle `for`.
- Arrêt `if (absent == S) i = N + 1`. Sortie de boucle correcte (hack).
- Logique correcte.
- Variable `x` lue correctement.

NOTE FINALE : 17 / 20

Feedback :

- **Appréciation globale : Très Bon.**
-

Analyse :

Algorithmique :

- Lecture confuse (`scanf` successifs écrasant les vars).
- Utilise `x` non lu dans le `if`.

NOTE FINALE : 05 / 20

Feedback :

- **Appréciation globale : Insuffisant.**
-

Analyse :

Algorithmique :

- Commentaires en Arabe/Français.
- Lectures correctes.
- Compteurs z , M .
- Condition finale `if (z == 5)`. Valeur magique 5 ? (Peut-être S vaut 5 dans son esprit ?).
- Condition boucle sur N uniquement.

NOTE FINALE : 13 / 20

Feedback :

- **Appréciation globale : Moyen.** Constante magique inexpliquée.
-

Analyse :

Algorithmique :

- Condition `while (i < N && i != S)`. Utilise l'indice `i` au lieu du compteur d'absents. Arrêt prématuré si on arrive à l'étudiant numéro `S`.
- Calcul `p = p + i`. Somme des indices ? Faux.

NOTE FINALE : 08 / 20

Feedback :

- **Appréciation globale : Fragile.**
-

Analyse :

Algorithmique :

- Boucle `while (N > 0)`. Boucle infinie (N ne change pas).
- Boucles `for` imbriquées bizarres.
- Demande de saisir le nombre d'absents ? (Contresens).

NOTE FINALE : 04 / 20

Feedback :

- **Appréciation globale : Très Insuffisant.**
-

Analyse :

Algorithmique :

- Variable 5 en nom de variable ? (`int N, A, 5...`) Syntaxe illégale.
- Variable `random` utilisée non init (non, init à 1).
- `if (X < A) : if (random) ->` condition toujours vraie. Compte les absents comme présents ("attended") ? Logique inversée et étrange.
- Tente de simuler de l'aléatoire ?

NOTE FINALE : 06 / 20

Feedback :

- **Appréciation globale : Insuffisant.** Code confus.
-

Analyse :

Algorithmique :

- Lecture chaotique. Ecrasement variables.
- Boucle `while (X < A)`. Condition sur variable d'entrée non modifiée -> Boucle infinie ou jamais exécutée.
- Incrémente `s` ?

NOTE FINALE : 03 / 20

Feedback :

- **Appréciation globale : Très Insuffisant.**
-

Analyse :

Algorithmique :

- Boucle `while (x != A)`. Condition d'arrêt étrange (tant que la saisie != seuil ?).
- Incrémentation `i` dans les deux sens (+1 ou -1).
- Affichage final douteux.

NOTE FINALE : 05 / 20

Feedback :

- **Appréciation globale : Insuffisant.**