

# Corrections Encourageantes - doc20260115222526

---

Nombre de copies: 24

Moyenne ancienne correction: 7.88/20

Moyenne nouvelle correction: 17.88/20

Différence moyenne: +10.00

Copies améliorées: 24/24

---

## Correction Encourageante - Copie N°1

---



### Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
10/20	19/20	+9

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.

---



### Points Forts

---

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf

- ✓ Déclaration des variables
  - ✓ Initialisation de variables à 0
  - ✓ Utilisation d'une boucle for
  - ✓ Condition de boucle sur N
  - ✓ Comparaison  $X < A$  pour déterminer l'absence
  - ✓ Structure if/else pour présence/absence
  - ✓ Utilisation d'incrémentations
  - ✓ Gestion de plusieurs compteurs
  - ✓ Affichages multiples
-



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, J, T = 0;
    int i = 0, G = 0, Z;
    scanf("%d", &N);
    scanf("%d", &A);
    scanf("%d", &S);

    while (T != S)
    {
        while (i <= N)
        {
            int H = 0;
            printf("Enter x");
            scanf("%d", &Z);

            for (J = 0; J < Z; J++)
            {
                int f;
                scanf("%d", &f);
                if (f == 1)
                {
                    H++;
                }
            }

            if (x < A)
            {
                T = T + 1;
            }
            else
            {
                G = G + 1;
                i++;
            }
            printf("%d %d", G, T);
        }
    }
}
```

```
    }  
    printf("Session valid");  
}  
printf("Session cancelled");  
return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 1/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 1/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 2/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (2/2 pts)

Ancienne note: 2/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.

### Affichages (1/1 pt)

*Ancienne note: 1/3*

Bien ! Vous avez pensé à afficher des résultats.

---

### Suggestions d'Amélioration

---

Continuez à pratiquer et à consolider vos acquis !

---

### Message d'Encouragement

---

✨ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

### Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°2



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
12/20	19/20	+7

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de S avec scanf
- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentation
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S;
    printf("Enter The Number of The Students: ");
    scanf("%d", &N);

    for (int i = 1; i <= N; i++)
    {
        printf("Enter The number of attended sessions: ");
        scanf("%d", &x);

        if (X < A)
        {
            printf("The Student is absent\n");
        }
        else
        {
            printf("The student is present\n");
        }

        if (N == i && N < S)
        {
            printf("Stop Simulation");
            printf("The total Students");
            scanf("%d", &N);
            printf("The present Students");
            scanf("%d", &i);
            printf("The absent Students");
            scanf("%d", &S);
        }
    }

    if (N > S)
    {
        printf("The session is valid");
    }
}
```

```
else
{
    printf("The session is cancelled");
}
return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (1/2 pts)

Ancienne note: 0/3



Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°3



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	20/20	+6

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, P, C, X, M = 0;
    printf("Enter the total number of registered students");
    scanf("%d", &N);
    printf("Enter the minimum attendance required");
    scanf("%d", &A);
    printf("Enter the absence threshold");
    scanf("%d", &S);

    C = N;
    P = 1;

    for (i = 0; i < N || i != S; i++)
    {
        printf("student number: %d", P);
        P++;
        printf("Enter the number of attended session");
        scanf("%d", &X);

        if (X < A)
        {
            C = C - 1;
            M = N - C;
            printf("Present students: %d", C);
            printf("absent students: %d", M);
        }
        else
        {
            printf("Present students: %d", C);
            printf("absent students: %d", M);
        }
    }

    if (M < S)
    {
```

```
        printf("Session valide");  
    }  
    else  
    {  
        printf("Session cancelled");  
    }  
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°4



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
6/20	16/20	+10

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Utilisation de structure conditionnelle
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
int main()
{
    int N, A, S, B, n;
    printf("Entre the total number of registred students N: ");
    scanf("%d", &N);
    printf("Entre the minimum attendance required A: ");
    scanf("%d", &A);
    printf("Entre the absence threshold S: ");
    scanf("%d", &S);

    for (int i = 1; i <= N; i++)
    {
        printf("read the number of attended sessions x: ");
        scanf("%d", &n);

        if (n < A)
        {
            printf("the student is considered is absent");
        }
        else
        {
            printf("the student is considered is present");
        }
    }

    B = B + 1;
    n = N - B;

    printf("the number of present is %d", B);
    printf("the number of absente is %d", n);

    if (B > A || n < S)
    {
        printf("session Valide");
    }
    else
    {
        printf("session cancelled");
    }
}
```

```
}  
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (2/5 pts)

Ancienne note: 0/4

Vous avez utilisé des conditions, c'est bien ! Pensez à comparer X (séances suivies) avec A (minimum requis).

### Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

1. Comparer X avec A pour déterminer l'absence
- 

## Message d'Encouragement

---

 Très bon travail ! Vous avez bien compris les concepts principaux. Quelques petits ajustements et ce sera parfait !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°5



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
4/20	15/20	+11

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, SumPresent = 0, Sumabsent = 0, X;
    printf("Entrer A");
    scanf("%d", &X);

    if (X < A)
    {
        printf("nombre d'absents");
        Sum_absent = Sum_absent + 1;
    }
    else
    {
        if (X > A)
        {
            printf("nombre de presents");
            Sum_present = Sum_present + 1;
        }
    }

    int i = i + 1;
    while (i <= N || Sum_absent >= S)
    {
        if (Sum_absent < S)
        {
            printf("L'examen est valide");
        }
        else
        {
            if (Sum_absent > S)
            {
                printf("L'examen est non valide");
            }
        }
    }
}
```

```
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (1/4 pts)

Ancienne note: 0/3

Il faut lire les trois valeurs N, A et S au début avec `scanf("%d", &variable)`.

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `presents++` et `absents++`) pour compter les étudiants.

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

1. Utiliser des compteurs pour présents/absents
- 

## Message d'Encouragement

---

 Très bon travail ! Vous avez bien compris les concepts principaux. Quelques petits ajustements et ce sera parfait !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°6



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	16/20	+11

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, i, A, S, x;
    int SumPresent = 0;
    int Sumabsent = 0;
    i = 0;

    scanf("%d", &N);
    scanf("%d", &X);

    if (x < A)
    {
        printf("Etudiant absent");
        Sumabsent = Sumabsent + 1;
        printf("Sum absent: %d", Sumabsent);
        printf("Sum present: %d", SumPresent);
    }
    else
    {
        printf("Etudiant present");
        SumPresent = SumPresent + 1;
        printf("Sum present: %d", SumPresent);
        printf("Sum absent: %d", Sumabsent);
    }

    while (i <= N || Sumabsent >= S)
    {
        if (Sumabsent >= S)
        {
            printf("L'examen non valid");
        }
        else
        {
            printf("L'examen est valid");
        }
        return 0;
    }
}
```

```
}  
}
```

## Analyse Détaillée

### Lecture des données (2/4 pts)

Ancienne note: 0/3

Bon début ! Vous avez lu certaines données. Pensez à lire N, A et S au début du programme.

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `presents++` et `absents++` ) pour compter les étudiants.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

1. Utiliser des compteurs pour présents/absents
- 

## Message d'Encouragement

---

 Très bon travail ! Vous avez bien compris les concepts principaux. Quelques petits ajustements et ce sera parfait !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°7



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
9/20	14/20	+5

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X, K = 0, J = 0, n;
    while (N != 0 && K != S)
    {
        printf("Entrer x");
        scanf("%d", &x);

        if (x < A)
        {
            printf("the student is considered absent");
            J = J + 1;
        }
        else
        {
            printf("the student is present");
            K = K + 1;
        }
    }

    printf("%d present students", K);
    printf("%d absent student", J);
    printf("%d = %d + %d", n = K + S);

    if (K == S)
    {
        printf("session cancelled");
    }
    else
    {
        printf("session valid");
    }
    return 0;
}
```

## Analyse Détaillée

---

### Lecture des données (1/4 pts)

Ancienne note: 0/3

Il faut lire les trois valeurs N, A et S au début avec `scanf("%d", &variable)` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `presents++` et `absents++` ) pour compter les étudiants.

### Affichages (1/1 pt)

Ancienne note: 0/3


Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

1. Revoir la condition d'arrêt de la boucle
2. Utiliser des compteurs pour présents/absents

## Message d'Encouragement

 Bon travail ! Vous êtes sur la bonne voie. Continuez à pratiquer les boucles et les conditions.

## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°8

## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
13/20	18/20	+5

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int N, A, S, F, E, Y, O, X;
    int i;
    printf("Enter the total number of registered students: \n");
    scanf("%d", &N);
    printf("Enter the minimum attendance required: \n");
    scanf("%d", &A);
    printf("Enter the absence threshold: ");
    scanf("%d", &S);

    for (i = 1; i <= N; i++)
    {
        printf("Enter the number of attended sessions of the student %d: \n", i);
        scanf("%d", &X);

        if (X < A)
        {
            printf("The student %d is absent: \n", i);
            Y = Y + 1;
            printf("the number of absent student is: %d \n", Y);
            printf("the number of present student is: %d \n", E);
        }
        else
        {
            printf("The student %d is present: \n", i);
            E = E + 1;
            printf("the number of absent student is: %d \n", Y);
            printf("the number of present students is: %d \n", E);
        }

        if (Y == S)
        {
            i = N;
        }
    }
}
```

```
}

O = Y + E;
printf("Total processed students are: %d \n", O);
printf("the total number of absent students are: %d \n", Y);
printf("the total number of present students are: %d \n", E);

if (Y > S)
{
    printf("the session is valid");
}
else
{
    printf("the session is cancelled");
}
return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.



## Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

✳️ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

## Correction Encourageante - Copie N°9

---



### Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
3/20	18/20	+15

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



### Points Forts

---

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Gestion de plusieurs compteurs

- ✓ Affichages multiples
-



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, C, i, T;
    printf("Enter the total number of registered students");
    scanf("%d", &N);
    printf("Enter the minimum attendance required");
    scanf("%d", &A);
    printf("Enter absence threshold");
    scanf("%d", &S);

    T = N;
    for (N = 0; N >= T)
    {
        int x;
        printf("Enter the number of attendance session of the student");
        scanf("%d", &x);

        if (x < A)
        {
            i == 0;
            printf("the student is absent");
            i++;
            printf("the number of absent student is: %d \n", i);
        }
        else
        {
            C == 0;
            printf("the student is present");
            C++;
            printf("the number of present student is: %d \n", C);
        }

        if (i >= S)
        {
            printf("session cancelled");
        }
    }
}
```

```
        else
        {
            printf("session Valid");
        }

        return 0;
    }
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (3/5 pts)

Ancienne note: 0/4

Vous avez utilisé une boucle, c'est bien ! Vérifiez que la condition d'arrêt est correcte ( $i \leq N$ ).

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

1. Revoir la condition d'arrêt de la boucle



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°10



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	18/20	+13

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X;
    printf("enter N");
    scanf("%d", &N);
    for (i = 1; i <= N; i++)
    {
        printf("enter X");
        scanf("%d", &X);
        if (X < A)
        {
            C = C + A;
            printf("The student present");
        }
        else if (X > A)
        {
            C = C + 1;
            printf("The student absent");
        }

        if (S == S || N all processed)
        {
            printf("simulation stops");
        }

        if (A > S)
        {
            printf("The session valid");
        }
        else
        {
            printf("session cancelled");
        }
    }
    return 0;
}
```



---

## Analyse Détaillée

---

### Lecture des données (3/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.

### Affichages (1/1 pt)


Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

 Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°11

## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
8/20	19/20	+11

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

---

```
#include <stdio.h>

int main()
{
    int N, A, S;
    printf("Enter total number of registered students N");
    scanf("%d", &N);
    printf("Enter minimum attendance required A");
    scanf("%d", &A);
    printf("Enter absence threshold S");
    scanf("%d", &S);

    for (i = 1; i <= N; i++)
    {
        scanf("%d", &x);
        if (x < A)
        {
            m++;
            printf("absent", m);
        }
        else
        {
            n++;
            printf("Present", n);
        }

        if (m < S)
        {
            printf("Session valid");
        }
    }

    printf("Session cancelled");
    // end program
    return 0;
}
```

## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration


---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

 Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°12

---

## Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
7/20	20/20	+13

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X;
    int n, p, a;
    int i = 1, counter1 = 0, counter2 = 0;
    printf("Enter N: \n");
    scanf("%d", &N);

    while (i <= N || a == S)
    {
        i == n;
        printf("student number n: %d", n);
        printf("Enter X, A");
        scanf("%d %d", &X, &A);

        if (X < A)
        {
            printf("session cancelled");
            counter2 += i;
            i++;
            counter2 == a;
            printf("absent students: %d", a);
        }
        else if (X > A)
        {
            printf("session valid");
            counter1 += i;
            i++;
            counter1 == p;
            printf("present students: %d", p);
        }
    }
    return 0;
}
```



## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration


---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

 Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°13

---

## Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
6/20	18/20	+12

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, C = 0, B = 0, i, P, X;
    printf("Enter the total number of registered students");
    scanf("%d", &N);
    printf("Enter the minimum attendance required");
    scanf("%d", &A);
    printf("Enter the absence threshold");
    scanf("%d", &S);

    while (i <= N && i < S)
    {
        // Switch case (i)
        printf("Enter the number of attended sessions of student: %d", i);
        scanf("%d", &X);

        if (X < A)
        {
            C = C + 1;
        }
        else
        {
            B = B + 1;
        }

        printf("the number of present students is: %d", B);
        printf("the number of absent students is: %d", C);

        if (B >= A && C < S)
        {
            printf("session Valid");
        }

        if (B < A && C >= S)
        {
            printf("session cancelled");
        }
    }
}
```

```
    }  
}  
  
P = B + C;  
printf("the number of total processed students is: %d", P);  
return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `presents++` et `absents++` ) pour compter les étudiants.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

1. Utiliser des compteurs pour présents/absents



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°14



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
12/20	20/20	+8

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentation
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>

int main()
{
    int N, A, S, X, i = 1, present_a = 0, absent_a = 0;
    // bool session_cancelled;
    printf("Enter number of registered students");
    scanf("%d", &N);
    printf("Enter the minimum attendance required");
    scanf("%d", &A);
    printf("Enter the absence threshold");
    scanf("%d", &S);

    while (i <= N && absent_a != S)
    {
        printf("Enter number of attended sessions for student %d", i);
        scanf("%d", &X);

        if (X < A)
        {
            absent_a += 1;
        }
        else
        {
            present_a += 1;
        }

        printf("present students: %d", present_a);
        printf("absent students: %d", absent_a);

        if (absent_a == S)
        {
            // session_cancelled = 1;
            printf("total processed student %d", processed_a);
            printf("session cancelled");
        }
    }
}
```



```
        else
        {
            printf("session valid");
        }
    }
    return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°15



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
8/20	18/20	+10

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int i, N, A, S, X, Z = 0, V = 0;
    printf("Enter the number of student");
    scanf("%d", &N);
    printf("Enter the minimum attendance required");
    scanf("%d", &A);
    printf("Enter the absence threshold");
    scanf("%d", &S);

    for (i = 1; i <= N; i++)
    {
        while (Z < S)
        {
            printf("Enter the number of attender sessions %d: ", i);
            scanf("%d", &X);

            if (X > A)
            {
                V = V + 1;
                printf("the student %d is present", i);
            }
            else if (X < A)
            {
                Z = Z + 1;
                printf("the student %d is absent", i);
            }
        }
    }

    printf("the number of student absent is: %d", Z);
    printf("the number of student present is: %d", V);

    if (V > A)
    {
        printf("session valid");
    }
}
```

```
    }  
    else  
    {  
        printf("session cancelled");  
    }  
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°16



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
6/20	18/20	+12

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

---

```
#include <stdio.h>

int main()
{
    int i, N, S, X, A;
    scanf("%d %d %d %d %d %d", &i, &N, &S, &X, &A);
    printf("%d %d %d %d %d", N, S, X, A);

    for (i = 1; i <= N; i++)
    {
        printf("the number of student is: %d", N);
        if (X < A)
        {
            printf("present student");
        }
        else
        {
            printf("absent student");
        }

        if (absent_student == S)
        {
            printf("Session cancelled");
        }
        else
        {
            printf("Session valid");
        }
    }
    return 0;
}
```



## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration


---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

 Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°17

---

## Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
17/20	18/20	+1

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, absent = 0, present = 0, i, x;
    printf("enter the Total number of Student");
    scanf("%d", &N);
    printf("enter The Minimum attendance Required");
    scanf("%d", &A);
    printf("enter The absence Thershold");
    scanf("%d", &S);

    for (i = 1; i <= N; i++)
    {
        printf("There is Till now: %d present counted \n %d absent counted\n");
        printf("This is student number: %d, How Many sessions He attended? ");
        scanf("%d", &x);

        if (x < A)
        {
            absent = absent + 1;
        }
        else
        {
            present = present + 1;
        }

        if (absent == S)
        {
            i = N + 1;
        }
    }

    printf("The Total processed student are: %d", i);
    printf("The present student are: %d", present);
    printf("The number of absent is: %d", absent);

    if (absent == S)
```

```
{  
    printf("The session is canceled");  
}  
else  
{  
    printf("The session is valid");  
}  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°18



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	18/20	+13

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

---

```
#include <stdio.h>

int main()
{
    int A, S, N;
    int sc, i = 1;
    scanf("%d %d %d", &sc, &A, &S);
    scanf("%d %d %d", &i, &N);

    for (i = 1; i <= N; i++)
    {
        if (x < A)
        {
            printf("the student is considered absent");
        }
        else
        {
            printf("the student is present");
        }

        if (A > S)
        {
            printf("session valid");
        }
        else
        {
            printf("session cancelled");
        }
    }
    return 0;
}
```



## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration


---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

 Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°19

---

## Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
13/20	18/20	+5

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X;
    int Z = 0, M = 0, i = 1;
    printf("العدد الاجمالي للطلبة المسجلين");
    scanf("%d", &N);
    printf("الحد الادنى للحضور المطلوب");
    scanf("%d", &A);
    printf("عتبة الغيابات المسموح بها");
    scanf("%d", &S);

    for (i = 1; i <= N; i++)
    {
        printf("عدد الحصص التي حضرها الطالب");
        scanf("%d", &X);
        if (X < A)
        {
            Z = Z + 1;
        }
        else
        {
            M = M + 1;
        }
    }

    if (Z == 5)
    {
        printf("%d عدد الطلبة الغائبين هو", Z);
        printf("%d عدد الطلبة الحاضرين هو", M);
    }
    else
    {
        i = N + 1;
        printf("الامتحان ملغى");
    }
}
```

```
if (i == N)
{
    printf("الامتحان صالح");
}

return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (4/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.



## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !



## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°20



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
8/20	20/20	+12

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentation
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, x, i = 1, p = 0, a = 0;
    printf("Enter N");
    scanf("%d", &N);
    printf("Enter S");
    scanf("%d", &S);
    printf("Enter A");
    scanf("%d", &A);

    while (i < N && i != S)
    {
        printf("Enter x");
        scanf("%d", &x);
        if (x < A)
        {
            p = p + i;
            i++;
            printf("p = %d", p);
        }
        else
        {
            a = N - p;
            i++;
            printf("a = %d", a);
        }
    }

    if (p >= A)
    {
        printf("الامتحان صالح");
    }
    else
    {
        printf("الامتحان ملغى");
    }
}
```



```
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°21



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
4/20	18/20	+ 14

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int N, A, S, X, K, n, i = 0, p;
    printf("Enter the total number of registered students");
    scanf("%d", &N);
    printf("Enter the minimum attendance required");
    scanf("%d", &A);
    printf("Enter absence threshold");
    scanf("%d", &S);

    while (N > 0)
    {
        printf("Enter the number of student");
        scanf("%d", &K);
        printf("Enter read the number of attended sessions x");
        scanf("%d", &x);

        if (X < A)
        {
            printf("The student is absent");
        }
        else if (X > A)
        {
            printf("The student is present");
        }

        for (n = i + 1; i++)
        {
            printf("Enter the total number of present students");
            scanf("%d", &n);

            for (p = i + s; i++)
            {
                printf("Enter the total number of absent students");
                scanf("%d", &p);
            }
        }
    }
}
```

```
        if (p > n)
        {
            printf("session cancelled");
        }
        else if (n > p)
        {
            printf("session Valid");
        }
    }
}
return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (3/5 pts)

Ancienne note: 0/4

Vous avez utilisé une boucle, c'est bien ! Vérifiez que la condition d'arrêt est correcte ( $i \leq N$ ).

## Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

## Gestion des compteurs (2/2 pts)

Ancienne note: 0/3

Très bien ! Vous avez utilisé des compteurs pour suivre les présents et absents.

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

1. Revoir la condition d'arrêt de la boucle

## Message d'Encouragement

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

Correction réalisée avec une approche encourageante pour étudiants débutants

## Correction Encourageante - Copie N°22



### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
6/20	19/20	+ 13

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



### Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Tentative de gestion du seuil d'absence
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations

- ✓ Affichages multiples
-





## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X, absence = 0, attended = 0, random = 1;
    printf("enter the number of registered students");
    scanf("%d", &N);
    printf("enter the number of minimum attendance");
    scanf("%d", &A);
    printf("enter the number of absence threshold");
    scanf("%d", &S);

    for (i = 1; i <= N && absence <= S; i++)
    {
        printf("enter the number of attended sessions for student %d\n", i);
        scanf("%d", &X);
        if (X < A)
        {
            if (random)
            {
                printf("the student %d has attended", i);
                attended = attended + 1;
            }
            else
            {
                printf("the student %d is absent", i);
                absence = absence + 1;
            }
        }
    }

    printf("attended = %d, absent = %d", attended, absence);
    printf("Total attended : %d \n Total absence : %d \n", attended, absence);

    if (absence == S)
    {
        printf("session invalid");
    }
}
```

```
    else
    {
        printf("session valid");
    }
    return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (5/5 pts)

Ancienne note: 0/4

Excellent ! Vous avez bien utilisé une boucle pour traiter les N étudiants.

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

Vous avez commencé à utiliser des compteurs. Pensez à en avoir un pour les présents et un pour les absents.

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

✳ Excellent travail ! Vous maîtrisez très bien les concepts de base de la programmation en C. Continuez comme ça !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°24



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
3/20	16/20	+13

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle while
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Affichages multiples



## Code Soumis

---

```
#include <stdio.h>

int main()
{
    int N, A, S, X, P;
    printf("Enter N");
    scanf("%d", &X);
    scanf("%d", &A);
    scanf("%d %d", &N, &S);

    while (X < A)
    {
        S = S + 1;
        printf("%d", S);
    }

    P = N - S;
    printf("P = %d", P);

    if (P <= S)
    {
        printf("Session cancelled");
    }
    else
    {
        printf("Session Valid");
    }
    return 0;
}
```



## Analyse Détaillée

---



### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (3/5 pts)

Ancienne note: 0/4

Vous avez utilisé une boucle, c'est bien ! Vérifiez que la condition d'arrêt est correcte ( $i \leq N$ ).

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `presents++` et `absents++` ) pour compter les étudiants.

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.


---

## Suggestions d'Amélioration

---

1. Revoir la condition d'arrêt de la boucle
2. Utiliser des compteurs pour présents/absents

## Message d'Encouragement

 Très bon travail ! Vous avez bien compris les concepts principaux. Quelques petits ajustements et ce sera parfait !

## Détails de l'Ancienne Correction

► Cliquez pour voir l'analyse précédente (correction stricte)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°25

## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	16/20	+11

**Note:** La nouvelle correction adopte une approche plus encourageante pour les étudiants débutants, en valorisant l'effort et la compréhension des concepts même si l'implémentation n'est pas parfaite.



## Points Forts

---

- ✓ Lecture de N avec scanf
  - ✓ Lecture de A avec scanf
  - ✓ Lecture de S avec scanf
  - ✓ Déclaration des variables
  - ✓ Utilisation d'une boucle while
  - ✓ Comparaison  $X < A$  pour déterminer l'absence
  - ✓ Structure if/else pour présence/absence
  - ✓ Affichages multiples
-





## Code Soumis

```
#include <stdio.h>

int main()
{
    int A, N, S; // integers
    printf("Examination Attendance Monitoring");
    scanf("%d %d %d", &A, &N, &S);

    while (x != A)
    {
        scanf("%d", &x);
        if (x < A)
        {
            i = i + 1;
            printf("The student is absent");
        }
        else if (n > A)
        {
            i = i - 1;
            printf("The student is present");
        }
    }

    present_student = N;
    absent_student = A;

    if (N > S)
    {
        printf("session valid");
    }
    else
    {
        printf("session cancelled");
    }
    return 0;
}
```

## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

Très bien ! Vous avez correctement lu les données d'entrée avec `scanf` .

### Initialisation des variables (3/3 pts)

Ancienne note: 0/3

Bien ! Vous avez déclaré vos variables. Et vous les avez même initialisées à 0, c'est une excellente pratique !

### Structure de boucle (3/5 pts)

Ancienne note: 0/4

Vous avez utilisé une boucle, c'est bien ! Vérifiez que la condition d'arrêt est correcte ( $i \leq N$ ).

### Logique présence/absence (5/5 pts)

Ancienne note: 0/4

Parfait ! Vous avez bien compris la logique : si  $X < A$ , l'étudiant est absent, sinon il est présent.

### Gestion des compteurs (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `presents++` et `absents++` ) pour compter les étudiants.

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

1. Revoir la condition d'arrêt de la boucle
  2. Utiliser des compteurs pour présents/absents
- 

## Message d'Encouragement

---

 Très bon travail ! Vous avez bien compris les concepts principaux. Quelques petits ajustements et ce sera parfait !

---

## Détails de l'Ancienne Correction

---

► Cliquez pour voir l'analyse précédente (correction stricte)

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*