

Corrections Encourageantes -

doc20260115221630

Nombre de copies: 28

Moyenne ancienne correction: 6.25/20

Moyenne nouvelle correction: 14.39/20

Différence moyenne: +8.14

Copies améliorées: 28/28

Correction Encourageante - Copie N°1



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
11/20	14/20	+3

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;
    int p = 0, b = 0;
    printf ( " enter total number of registered students " );
    scanf ( " %d ", & N );
    printf ( " enter minimum attendance required " );
    scanf ( " %d ", & A );
    printf ( " enter absence threshold " );
    scanf ( " %d ", & S );
    while ( p != N && b != S ) {
        printf ( " enter the number of attended sessions x " );
        scanf ( " %d ", & x );
        if ( x < A ) {
            b = b + 1 ;
        }
        else {
            p = p + 1 ;
        }
        printf ( " %d , Presents " , " %d absent" , p , b )
        if ( b == S ) {
            printf ( " session cancelled " );
        }
        else {
            printf ( " session valid " );
        }
    }
    return 0 ;
}
```

Analyse Détailée

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°2

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
7/20	14/20	+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

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Comparaison $X < A$ pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentation
- ✓ Affichages multiples



Code Soumis

```
# include <stdio.h>
# include <stdlib.h>
int N, A = 5, S, x;
printf ( " Enter total number of registered students: " );
scanf ( " %d ", & N );
printf ( " Enter absence threshold : " );
scanf ( " %d ", & S );
for ( int i = 1 ; i <= 20 ; i ++ ) {
printf ( Enter the number of attended sessions : " );
scanf ( " %d ", &x );
if ( x < A ) {
printf ( " student is absent. " );
else
printf ( " student is present." );
}
if ( S > N ) {
printf ( " the student is in the list of the absent students." );
else
printf ( " the student is in the list of the present students." );
} }
printf ( " Student number is : " );
printf ( " present student is : " );
printf ( " absent students is :" );
if ( S > N ) {
printf ( " the Session is cancelled. " );
else
printf ( " the Session is valid. " );
```



Analyse Détailée



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 (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 (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. Revoir la condition d'arrêt de la boucle
-

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°3

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'incrémentation
 - ✓ Affichages multiples
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, x ;
    int i, count 1 = 0, count 2 = 0 ;
    printf ( " Enter N, A, S : " ) ;
    scanf ( " %d %d %d ", &N, &A, &S ) ;
    for ( i = 1 ; i <= N ; i ++ ) {
        scanf ( " %d " , &x ) ;
        if ( X < A ) {
            printf ( " %d ", i ) ;
            printf ( " the student absent " ) ;
            count 1 = count 1 + 1 ;
            printf ( " Absent student : %d ", count 1 ) ;
        }
        else {
            printf ( " %d ", i ) ;
            printf ( " the studen present " ) ;
            count 2 = count 2 + 1 ;
            printf ( " present student = %d ", count 2 ) ;
        }
        if ( count 1 >= S ) {
            printf ( " Simulation stops " ) ;
        }
        if ( count 1 < S ) {
            printf ( " Session valid " ) ;
        }
        else {
            printf ( " Session cancelled " ) ;
        }
    }
    return 0 ;
}
```

Analyse Détailée

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°4

Comparaison des Notes

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

- ✓ Messages utilisateur avant saisie
- ✓ 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'incrémantation
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



Code Soumis

```
# include <stdio.h>
int main ( ) {
int N, A, S;
int Student [M], d = 0, L = 0, i = 0 ;
printf ( " enter the total number of registered student N " ) ;
printf ( " enter the minimum attendance required A " ) ;
printf ( " enter the absente thershold S " ) ;
scanf ( " %d %d %d " , & N, & A, & S ) ;
for ( i = 0 ; i < N ; i ++ ) {
scanf ( " %d " , & Student [i] = x ) ;
if ( x < A ) {
printf ( " the student is absent % d, Student [i] ) ;
d += 1 ;
} else {
printf ( " the student is persent % d, Student [i] ) ;
L += 1 ;
}
printf ( " % d " , Student [i] ) ;
printf ( " d = %d " , d ) ; // عدد الطلبة الحاضرين
printf ( " L = %d " , L ) ; // عدد الطلبة الغائبين
if ( d > S ) {
printf ( " sissan valid " ) ;
} else {
printf ( " sisson cancelled " ) ;
}
return 0 ;
}
```



Analyse Détailée



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 (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

 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
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

- ✓ 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, X, absent, present ;
    printf ( " Enter the students number ", N ) ;
    scanf ( " %d ", & N ) ;
    printf ( " Enter the minimum attendance required ", A ) ;
    scanf ( " %d ", & A ) ;
    printf ( " Enter the absence threshold ", S ) ;
    scanf ( " %d ", & S ) ;
    For ( i = 1 ; i <= n ; i ++ ) {
        scanf ( " %d ", & x ) ;
        if ( x < A ) {
            absent = absent + 1 ;
        } else {
            present = present + 1 ;
        }
        printf ( " Present students are: %d ", & present ) ;
        printf ( " Absent students are: %d ", & absent ) ;
        if ( N = absent + present || absent = S ) {
            printf ( " stop the program " ) ;
        }
        if ( absent >= S ) {
            printf ( " Session cancelled " ) ;
        } else {
            printf ( " Session valid " ) ;
        }
    }
    return 0 ;
}
```



Analyse Détailée



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 (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

 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	15/20	+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

- ✓ 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
 - ✓ Utilisation d'incrémentation
 - ✓ Affichages multiples
-



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int N, A, S, X, alesent, present
    printf ( " Enter total number of registered students: " N );
    scanf ( "%d", & N );
    printf ( " Enter the minimum attendance required \ m ", A );
    scanf ( "%d", & A );
    printf ( " Enter alesence threshold \ n ", S );
    scanf ( "%d", & S );
    for ( i = 1 ; i <= n ; i ++ ) {
        scanf ( "%d" . & x );
        if ( x < A ) {
            alesent = alesent + 1 ;
        } else {
            present = present + 1 ;
        }
        printf ( " present student are: %d " , present );
        printf ( " alesent student are: %d " , alesent );
        if ( N = alesent + present || present ) {
            printf ( " stop the program " ) ;
        }
        if ( alesent > S ) {
            printf ( " Session concelled " ) ;
        } else {
            printf ( " Session valid " ) ;
        }
    }
    return 0 ;
}
```



Analyse Détailée



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 (4/5 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

 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
7/20	19/20	+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
 - ✓ Messages utilisateur avant saisie
 - ✓ 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

```
# includ <stolin.h>
int main ( ) {
int N, A, S;
int x;
int present = 0, absent = 0;
int i = 0;
printf ( " Enter total numbers: " );
scanf ( "%d", &N );
printf ( " Enter minimum attendance : " );
scanf ( "%d", & A );
printf ( " Enter absence threshold: " );
scanf ( "%d", & S );
while ( i < N && absent < S )
{ printf ( " Enter attended session for student" );
scanf ( "%d", &x );
if ( x < A ) {
absent ++
else {
present ++ ;
i ++ ;
printf ( " Prossed = %d, i );
printf ( " preset = %d ; present );
printf ( " abset = %d ; abset );
if ( absence >= S )
printf ( " session canceled " )
else
{ printf ( " session valid " );
return 0;
}
```



Analyse Détailée



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 (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°8

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
1/20	13/20	+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

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ 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 ;
    printf ( " Enter the number of attended sessions: " );
    scanf ( " %d ", & X );
    if ( X < A )
        printf ( " the student is absent " );
    else
        printf ( " the student is present " );
    for ( N = the totale number of registered students ) {
        student number ;
        present students = N - absent students ;
        absent students = N - present students;
    }
    if ( X > S )
        printf ( " session valid " );
    else
        printf ( " session cancelled " );
    return 0;
}
```

Analyse Détailée

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 (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

 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°9

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
- ✓ Messages utilisateur avant saisie
- ✓ 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 <stdia.h>
int main ( )
{
    int N, A, S;
    int x, B = 0, P = 0;
    printf ( " enter the number of registered students" );
    scanf ( " %d", &N );
    printf ( " enter the minimum attendance required" );
    scanf ( " %d", & A );
    printf ( " enter the absence thresholds" );
    scanf ( " %d", & S );
    for ( i = 0 ; i < N || B == S ; i ++ ) {
        printf ( " enter the number of attendance of student %d:", i );
        scanf ( " %d", & x );
        if ( x < A )
            B ++ ;
        else
            P ++ ;
        printf ( " number of students %d \n present student: %d \n absent student "
        printf ( " total of pressed students is: %d ", i );
        printf ( " present students are: %d ";
        printf ( " abesent student are : %d , B );
        if ( B < P )
            printf ( " in session valid" );
        else
            printf ( " in session cancelled" );
    return 0 ;
```



Analyse Détailée



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 (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°10

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	17/20	+3

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
 - ✓ Utilisation d'incrémentation
 - ✓ Gestion de plusieurs compteurs
 - ✓ Affichages multiples
-



Code Soumis

```
include < stdio.h >
int main ( ) {
int N, A, S :
int present = 0 ; absent = 0
int i = 0 ;
printf ( Enter N ( total students ) : " ) ;
Scanf ( " %d ", & N ) ;
printf ( " Enter A ( minimum attendance ) : " ) ;
Scanf ( " %d ", & A ) ;
printf ( " Enter S ( absent threshold ) : " ) ;
Scanf ( " %d ", & S ) ;
while ( i < N && absent < S ) {
i ++ ;
printf ( " Student & d% - attended session
Scanf ( " %d ", & x ) ;
if ( X < A ) {
absent ++ ;
} else {
present ++ ;
}
printf ( " Step % d : \n ", i ) ;
printf ( " present = % d \n ", present ) ;
printf ( " absent = % d \n ", absent ) ;
printf ( " Final Results : \n " ) :
printf ( " Processed students = % d \n ", i ) ;
printf ( " present students = % d \n ", present ) ;
printf ( " absent students = % d \n ", absents ) ;
if ( absent >= S {
printf ( " Session concelled \n " ) ;
} else {
printf ( " Session valid \n " );
}
return 0 ;
```

Analyse Détailée

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 (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

 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°11

Comparaison des Notes

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

- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Utilisation de structure conditionnelle
- ✓ Utilisation d'incrémentation
- ✓ Gestion de plusieurs compteurs



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int N, A, S, n = 0, m = 0;
    Print f ( " entre the number A, N, S ) ;
    scanf ( " %d %d %d", & A, & N, & S ) ;
    for ( i = 1 ; i <= N ; i ++ )
        scanf ( " %d ", & n ) ;
    if ( n < A ) {
        n ++ ;
        print f ( " absent %d ", n ) ;
    } else
        m ++
    print f ( " present %d ", m ) ;
    Sum 1 = Sum 1 + 1 ;
} else {
    print f ( " student mes %d " ) ;
    Sum 2 = Sum 2 + 1 ;
}
if ( Sum 1 > Sum 2 ) {
    print f ( " session cancelled " )
} else { print f ( " session valid " ) }
return 0 ;
}
```



Analyse Détailée



Lecture des données (0/4 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 (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 (2/2 pts)

Ancienne note: 0/3

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

Affichages (0/1 pt)

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

Suggestions d'Amélioration

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

Message d'Encouragement

 Vous avez compris plusieurs concepts importants. Avec un peu plus de pratique sur les boucles et les compteurs, vous allez progresser rapidement !



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
3/20	14/20	+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

- ✓ 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, x, A, S, L = 0, M = 0;
    Printf ( type a number total ) ;
    Scanf ( "%d", & N ) ;
    Printf ( type a number of A ) ;
    Scanf ( "%d %d", & A, & M );
    while ( L > S && L >= N ) {
        Printf ( " the Number of attinded " );
        Scanf ( "%d", & x );
        if (x < A) {
            L = L + 1 ;
        }
        else {
            M = M + 1 ;
        }
        Printf ( " the student is absent " );
        Printf ( " the student is presnt " );
        if ( L > N ) {
            Printf ( " Session valid " );
        }
        else {
            Printf ( " Session cancelled " );
        }
    }
    return 0 ;
}
```



Analyse Détailée



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°13



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	12/20	+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

- ✓ 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



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, f = 0, r = 0, i = 1, Xi
    print f ( " Total number of registered student: " ) ;
    scanf ( " %d ", & N ) ;
    print f ( " minimum attendance required: " ) ;
    scan f ( " %d ", & A ) ;
    Print f ( " absencé threshold: " ) ;
    scan f ( " % ", & S ) ;
    while ( r >= S && i >= N ) {
        print f ( " the student %d ", i ) ;
        Scanf ( " %d ", & X ) ;
        if ( X < A ) {
            r = r + 1 ;
        else f = f + 1 ;
        print f ( " number of present %d " , f )
        print f ( " number of absent %d ", r ) ;
        if ( A > f && r > S ) {
            print f ( " session cancelled " ) ;
        else
            print f ( " Session valid " ) ;
        return 0 ;
    }
```

Analyse Détailée

Lecture des données (0/4 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 (0/1 pt)

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

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°14

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
2/20	13/20	+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
- ✓ Comparaison $X < A$ pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Affichages multiples



Code Soumis

```
# include < STDiO.h >
int main ( ) {
    int N, M = 0, L = 0, S, X, A ;
    printf ( " enter The number
N, M, L, S, X and A " ) ;
    Scanf ( " %d ", & N ) ;
    Scanf ( " %d " , & M ) ;
    Scanf ( " %d ", " %d ", & L, & X ) ;
    Scanf ( " %d ", " %d ", & S, & A ) ;
    while ( X = A ) {
        if ( X = A ) {
            M = N ;
            printf ( " present students " ) ;
        else if ( X < A )
            M = 0 ;
            printf ( " absent Students " ) ;
        }
        if ( M = N ) {
            M = N - L ;
            printf ( " %d ", M ) ; }
        else {
            L = N - M ;
            printf ( " %d ", L ) ;
        }
        if ( X < S ) {
            printf ( " session valid \n " ) : ;
        else
            printf ( " session concelled \n "
        }
        return 0 ;
    }
```

Analyse Détailée

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 (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

 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°15

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
1/20	14/20	+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

- ✓ Messages utilisateur avant saisie
- ✓ 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'incrémentation
- ✓ Affichages multiples



Code Soumis

. The correct of the exercise : - examination attendance monitoring

```
# include <stdio.h>
int main ( ) {
    int n, a, S ;
    printf ( " enter number : " ) ;
    for ( j = 0 ; j <= x, j ++ ) {
        printf ( " read the number of attended session x " ) ;
        if ( x < A )
            printf ( " The student is considered absent " ) ;
        else
            printf ( " The student is present " ) ;
    }
    if ( x == 0 )
        printf ( " all N student are processed or the number of absent
student rechears " ) ;
    while ( " N >= x " )
        present student = N - S
        absent student = N - A
        if ( S > 5 )
            printf ( " session concelled " )
        else {
            printf ( " session valid " ) .
        }
    Scanf ( " total number of student present and absent : % d \n " ) ;
    return 0 ;
}
```



Analyse Détailée



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 (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 (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. Revoir la condition d'arrêt de la boucle
-

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°16

Comparaison des Notes

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

- ✓ 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'incrémentation



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, x, i ;
    Print f ( " Enter the number of registered Students \n " ) ;
    Scanf ( " %d ", & N ) ;
    for ( i = 1 ; i <= N ; i ++ ) {
        Scanf ( " %d ", & x ) ;
        If ( x < A ) {
            Print f ( " the student is considered absent " ) ;
        else
            Print f ( " the student is present " ) ;
        }
    }
    If ( x == S ) {
        Print f ( " Session cancelled " ) ;
    else
        Print f ( " Session Valid " ) ;
    }
    return 0 ;
}
```

Analyse Détailée

Lecture des données (0/4 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 (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 (0/1 pt)

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

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°17

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
3/20	10/20	+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

- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Comparaison $X < A$ pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentation
- ✓ Gestion de plusieurs compteurs



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, i = 1 ;
    int X, present = 0, absent = 0 ;
    print f ( " total number of registered student ", N ) ;
    Scanf ( " %d ", & N ) ;
    print f ( " minimum attendance required ", A ) ;
    Scanf ( " %d ", & A ) ;
    print f ( " absence threshold ", S ) ;
    Scanf ( " %d ", & S ) ;
    if ( X < A ) {
        absent ++ ;
    else
        present ++ ;
    print f ( " Step %d : \n ", i ) ;
    print f ( " present student %d : \n ", present ) ;
    print f ( " Absent student %d : \n ", Absent ) ;
    i ++ ;
    }
    print f ( " present student %d / n ", present ) ;
    print f ( " absent student %d / n ", absent ) ;
    print f ( " Session valid " ) ;
    print f ( " Session cancelled " ) ;
    return 0 ;
}
```



Analyse Détailée



Lecture des données (0/4 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 (0/5 pts)**

Ancienne note: 0/4

Il faut utiliser une boucle `for` ou `while` pour traiter chaque étudiant. Exemple:

```
for(i=1; i<=N; i++)
```

 **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 (0/1 pt)**

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

 **Suggestions d'Amélioration**

1. Ajouter une boucle pour traiter N étudiants
-

Message d'Encouragement

 Vous avez compris plusieurs concepts importants. Avec un peu plus de pratique sur les boucles et les compteurs, vous allez progresser rapidement !

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
11/20	15/20	+4

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

- ✓ Déclaration des variables
 - ✓ Initialisation de variables à 0
 - ✓ Utilisation d'une boucle while
 - ✓ Condition de boucle sur N
 - ✓ 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 ;
    int Present = 0, Apsent = 0;
    int i = 0 ;
    scanf ( " %d ", & N ) ;
    scanf ( " %d ", & A ) ;
    scanf ( " %d ", & S ) ;
    while ( i < N & Apsent < S ) {
        Scanf ( " %d " & X ) ;
        if ( X < A )
            I Apsent ++ ;
        else
            Presert ++ ;
        i ++ ;
        Print f ( " Step %d : Preset = %d Apset = %d \n " ,
        i, Preset, Apset ) ; }
    Printf ( " total stadets : %d \n " , i ) ;
    Printf ( " Preset stadets : %d \n " , preset ) ;
    Printf ( " Apsent studets : %d \n " , Apsent ) ;
    if ( Apsent >= S ) {
        Printf ( " Session canceled " ) ; }
    else {
        Printf ( " Session valid " ) ; }
    Return 0 ;
}
```



Analyse Détailée



Lecture des données (0/4 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 (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

 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°19

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
13/20	19/20	+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
 - ✓ 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, P = 0, a = 0;
    Scanf ( " %d %d %d ", &N, &A, &S ) ;
    for ( int i = 1, i <= N ; i ++ ) {
        printf ( " Entre the number of attended sessions %d ", i ) ;
        Scanf ( " %d ", & x ) ;
        if ( x < A ) {
            a ++ ;
            printf ( " abrent Students is %d \n ", a ) ;
        } else { P ++ ;
            printf ( " present Students is %d \n ", P ) ;
        }
        if ( a == S ) {
            printf ( " Session cancelled " ) ;
            return 0 ;
        }
        printf ( " total processed Students is %d \n ", i ) ;
        printf ( " present Sudents is %d \n ", P ) ;
        printf ( " abrent Sudents is %d \n ", a ) ;
        if ( a <= S ) {
            printf ( " Sessio Valed " ) ;
        } else { printf ( " Session cancelled " ) ;
        }
        return 0 ;
    }
```



Analyse Détailée



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°20



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
13/20	14/20	+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

- ✓ 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
 - ✓ Utilisation d'incrémentation
 - ✓ Gestion de plusieurs compteurs
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, X, student Number = 1, Attendence = 0, Absence = 0 ;
    // N is the number total, A is the min of classes, X the number of classes t
    Print f ( " Enter the number total of students: " ) ;
    scanf ( " %d ", & N ) ;
    Print f ( " Enter the minimum classes attended by a single student: " ) ;
    Scanf ( "%d", & A ) ;
    Print f ( " Enter the absence thrushold : " ) ;
    Scanf ( " %d ", & S ) ;
    while ( student Number <= N && absence < S ) {
        Print f ( " how many classes did the student attend " ) ;
        scanf ( " %d ", & x ) ;
        if ( x < A ) { Printf ( " the student is counted as absent " ) ;
            absence ++ ; }
        else { Print f ( " the student is present " ) ; Print f ( " %d were presen
            Attendence ++ ; Printf ( " %d were absent \n ", absence ) ;
        }
        Print f ( " student number : %d ", studen Num ) ; if ( absence < S ) {
            Print f ( " %d are present \n ", Attendence ) ; Print f ( " session valid
            Print f ( " %d are absent \n ", absence ) ; else { Print f ( " session inv
            student Number ++ ; return 0 ;
        } // the end of the while loop.
    }
```



Analyse Détailée

Lecture des données (0/4 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 (2/2 pts)

Ancienne note: 0/3

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

Affichages (0/1 pt)

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

Suggestions d'Amélioration

1. Revoir la condition d'arrêt de la boucle
 2. Ajouter plus d'affichages pour suivre l'évolution
-

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°21

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
10/20	14/20	+4

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

- ✓ 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'incrémantation



```
# include < stdio.h >
int main ( ) {
    int N, A, S, X, i, present = 0, absent = 0, a ;
    print f ( " Enter the total number of registered students: " ) ;
    Scanf ( " %d ", & N ) ;
    print f ( " Enter the minimum attendance required: " ) ;
    Scanf ( " %d ", & A ) ;
    print f ( " Enter absent threshold: " ) ;
    Scanf ( " %d ", & S ) ;
    for ( i = 1, i <= N, i ++ ) {
        a = i ;
        print f ( " the student number %d \n ", i ) ;
        print f ( " present students %d \n ", present ) ;
        print f ( " absent students %d \n ", absent ) ;
        print f ( " How many attended sessions: \n " ) ;
        Scanf ( " %d ", & X ) ;
        if ( X < A ) {
            absent = absent + 1 ;
        else
            present = present + 1 ;
        if ( absent == S )
            i = N + 1 ;
        }
        if ( absent == S ) {
            print f ( " %d ", a ) ; // proceed students if we reach S
        else
            print f ( " %d ", N ) ; } // proceed students normally
        print f ( " present students %d ", present ) ;
        print f ( " absent students %d ", absent ) ;
        if ( present >= A && absent <= S )
            session valid ;
        else
            session invalid ;
    return 0 ;
```

Analyse Détailée

Lecture des données (0/4 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 (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 (0/1 pt)

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

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°22

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
8/20	17/20	+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

- ✓ Messages utilisateur avant saisie
- ✓ 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 >
int main ( ) {
    int N, A, S, X, i, count 1 = 0, count 2 = 0 ;
    Printf ( " enter The Number of register student : " ) ;
    Scanf ( " %d ", & N ) ;
    Printf ( " enter The minmumattendance required : " ) ;
    Scanf ( " %d ", & A ) ;
    Printf ( " enter absence Thre Shold : " ) ;
    Scanf ( " %d ", & S ) ;
    X = 6 ;
    for ( i = 1 ; i <= N || i == S ; i ++ ) {
        if ( A > X ) {
            count 1 ++ ;
            Printf ( " The student absent : " ) ;
        } else {
            Printf ( " The student Present : " ) ;
            count 2 ++ ;
        }
        if ( count 1 <= S ) {
            Printf ( " Session not valid " ) ;
        } else {
            Printf ( " Session valid " ) ;
        }
        Printf ( " tootal Processed Student is %d ", N ) ;
        Printf ( " tootol of student Present : %d ", count 1 ) ;
        Printf ( " total of tu dent absent : %d ", cont 2 ) ;
    }
    return 0 ;
}
```



Analyse Détailée



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 (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

 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°23

Comparaison des Notes

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

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Comparaison X < A pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentation
- ✓ Affichages multiples



Code Soumis

```
# include < studio.h >
int main ( )
int X ; N ; S ; P ; a ; s ; i = 1
for ( i = 1 ; N = i + 1 ; i ++ ) :
printf ( " أدخل رقم الطالب " ) ;
Scanf ( " % d % f S " ) ;
printf ( " أدخل عدد الحصص التي حضرها الطالب " ) :
Scanf ( % d : & x ) :
if ( x < A )
printf ( " الطالب غائب " ) :
a = i + 1
else printf ( " الطالب حاضر " ) ;
P = i + 1
:( P / " عدد الطلبة الحاضرين " printf
( S / " عدد الطلبة الغائبين " printf
if ( S >= S )
printf ( " الامتحان صالح " ) :
else printf ( " الامتحان ملغى " ) ;
return 0 ;
```

Analyse Détailée

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 (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 (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. Revoir la condition d'arrêt de la boucle

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°24

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
2/20	13/20	+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
- ✓ 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 ) ;
    Printf ( " read the number of attended sessions " ) ;
    Scanf ( X ) ;
    while ( X < A, the student is considered absent " ) ;
    else ( the student is present ) ;
    Printf ( " cont: present student, absent student " ) ;
    if ( all N student are processed or the numbre of absent stu dent
        reaches S = Stop ) ;
    Printf ( " Stop " numbre of present student & absent satudent " ) ;
    Scanf ( total processed student, Present student & absent student ) ;
    if ( X > A, the session is valid ) ;
    if ( X < A, the session cancelled )
        return 0 ;
}
```

Analyse Détailée

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 (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

 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°25

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
1/20	12/20	+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

- ✓ Déclaration des variables
- ✓ 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



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, x ;
    print f ( " enter the total number of registered students : " ) ;
    scan f ( " %d ", & N ) ;
    print f ( " enter the minimum attendance required : " ) ;
    scan f ( " %d ", & A ) ;
    print f ( " enter the absence threshold : " ) ;
    scan f ( " %d ", & S ) ;
    x =
    if ( x < A )
        print f ( " the students is considered absent " ) ;
    else {
        print f ( " the students is present : " ) ;
    }
    N - A = absent student ;
    N - S = Session canelled .
    while ( x == n || x == S ) ;
    print f ( " stop the programme
    return 0 ;
}
```



Analyse Détailée



Lecture des données (0/4 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 (0/1 pt)

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

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°26

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 for
 - ✓ 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, x, i ;
int comptem 1 = 0, comptem 2 = 0 ;
printf ( " ادخل عدد الاجمالي للطلبة المسجلين " ) ;
scanf ( " %d " , & N ) ;
printf ( " ادخل الحد الادنى للحضور المطلوب " ) ;
scanf ( " %d " , & A ) ;
printf ( " ادخل عتبة الغيابات المسموح بها " ) ;
Scnf ( " %d " , & S ) ;
for ( i = 1 ; i <= x ; i ++ ) {
Printf ( " عدد الحصص الذى حضرها الطالب " ) ;
Scanf ( " %d " , & x ) ;
if ( x < A ) {
Printf ( " الطالب غائب " ) ;
else { Printf ( " الطالب حاضر " ) }
compteur 1 = - comteu 1 ;
cmpteur 2 = N - comteur 2 ;
Printf ( " الطلبة الحاضرين " , comteur 1 ) ;
Printf ( " الطلبة الغائبين " , comteur 2 ) ;
if ( comteur 2 = S && N = N )
Printf ( " لا يوجد شيء " ) ;
if ( comteur 2 > conter 1 ) { Printf ( " الامتحان غير صالح " ) }
else if ( comteur 1 > conter 2 ) { Printf ( " الامتحان صالح " ) }
}
```



Analyse Détailée



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 (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. Revoir la condition d'arrêt de la boucle
-

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°27

Comparaison des Notes

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

- ✓ Messages utilisateur avant saisie
 - ✓ 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'incrémentation
 - ✓ Affichages multiples
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, x, c, bj
    printf ( " enter the number of Student : " );
    scanf ( " %d, & N );
    printf ( " enter minimum attendance required : " );
    scanf ( " %d ", & A );
    printf ( " enter absence threshold : " );
    scanf ( " %d ", & S );
    for ( i = 1 ; i <= N ; i ++ ) {
        x = x + 1 ;
        if ( x < A ) {
            printf ( " Student is Absent " );
        } else {
            printf ( " student is present " );
        }
        c = N - bj
        if ( c == S ) {
            print f ( Simulation stop );
        } else {
            printf ( Simulation continue );
        }
        printf ( " the number of present students is : %d \n " );
        printf ( " the number of absent studens is : %d \n " );
        if ( b > c )
            printf ( session valid );
        else
            printf ( session cancelled );
    }
    return 0 ;
}
```



Analyse Détailée



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 (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

 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°28

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
0/20	7/20	+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

- ✓ Déclaration des variables
- ✓ Comparaison $X < A$ pour déterminer l'absence
- ✓ Utilisation d'incrémentation



Code Soumis

```
include < stdio.h >
int main ( ) {
int N, A, S;
Print f ( " Entre les naumbre " );
Scanf ( " %d %d %d ; N, A, S \n " );
wahl {
Print f ( " x < A " );
Scanf ( " %d ; X < A, S > X, i ++ n \ " );
rutorn 0
}
}
```



Analyse Détaillée



Lecture des données (0/4 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 (0/5 pts)**

Ancienne note: 0/4

Il faut utiliser une boucle `for` ou `while` pour traiter chaque étudiant. Exemple:

```
for(i=1; i<=N; i++)
```

 **Logique présence/absence (3/5 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 (0/1 pt)**

Ancienne note: 0/3

N'oubliez pas d'afficher les résultats avec `printf`.

 **Suggestions d'Amélioration**

1. Ajouter une boucle pour traiter N étudiants
-

 **Message d'Encouragement**

 Vous avez fait des efforts et montré que vous comprenez certains concepts de base. Concentrez-vous sur la structure des boucles et la lecture des données. Vous pouvez y arriver !



Détails de l'Ancienne Correction

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

Correction réalisée avec une approche encourageante pour étudiants débutants