

Corrections Encourageantes -

doc20260115222057

Nombre de copies: 23

Moyenne ancienne correction: 6.43/20

Moyenne nouvelle correction: 14.48/20

Différence moyenne: +8.04

Copies améliorées: 23/23

Correction Encourageante - Copie N°1



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
9/20	13/20	+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

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



Code Soumis

```
# include < stdio.h>
int main ( ) {
    int A, S, N, x, i ;
    int T = 0, R = 0 ;
    printf ( " enter the minimum attendance required \n " ) ;
    scanf ( " %d ", & A ) ;
    printf ( " enter the absence threshold \n " ) ;
    scanf ( " %d ", & S ) ;
    printf ( " enter total registered students " ) ;
    scanf ( " %d ", & N ) ;
    for ( i = 1 ; i < N ; i ++ ) {
        while ( S > T ) {
            printf ( " how many attendes does student %d have ? ", x ) ;
            scanf ( " %d ", & x ) ;
            if ( x >= A ) {
                R ++ ;
            }
            else {
                T ++ ;
            }
        }
        if ( S > T ) {
            printf ( " present students are %d ", R );
            printf ( " absent students are the total of %d ", T ) ;
            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 (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 (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

- 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
12/20	18/20	+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
- ✓ Affichages multiples



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int A, N, S, g = 0 ; h = 0 ; x ;
    Printf ( " Enter total number of register students : " ) ;
    Scanf ( "%d", &N ) ;
    Printf ( " mimimum attendence requir " ) ;
    Scanf ( "%d", &A ) ;
    Printf ( " absense three should " ) ;
    Scanf ( "%d", &S ) ;
    For ( int i = 1 ; i <= N ; i ++ ) {
        Printf ( " read the number of attended session x " ) ;
        Scanf ( "%d", &x ) ;
        if ( x < A ) {
            Printf ( " الطالب غائب " ) ;
            g = g + 1 ;
        } else {
            Printf ( " الطالب حاضر " ) ;
            h = h + 1 ;
        }
        Printf ( "%d عدد الطلبة الحاضرين " ) ;
        Printf ( "%d عدد الطلبة الغائبين " ) ;
        if ( g >= S ) {
            Printf ( " الامتحان صالح " ) ;
        } else {
            Printf ( " الامتحان ملغى " ) ;
        }
    }
    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°3

Comparaison des Notes

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

- ✓ 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
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
Variables A, S, N, x, i, T = 0, R = 0 int ;
Print f ( " Entre a N, A, S " ) ;
Scan f ( " %d %d %d ", & N, A, S ) ;
Print f ( " Entre a number of sessions x : " ) ;
Scan f ( " %d ", & x ) ;
if ( x < A ) {
Print f ( " Entre a student is absent " ) ;
Els
Print f ( " Entre a student is present " ) ;
for ( i = 1 ; i <= N, i ++ ) {
While ( S > T ) {
Print f ( " how many stende
does student %d have ? ", x ) ;
Scan f ( " %d ", & x ) ;
if ( x >= A ) {
R ++ ;
}
Else {
T ++ ;
}
}
}
if ( S > T ) {
Print f ( " present studet are
%d ", R ) ;
Print f ( " semia Valid " ) ;
else {
Print f ( " ression ancelled " ) ;
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 (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°4

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

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



Code Soumis

```
# incloud < stdio.h >
int main ( ) {
    int N, A, S;
    printf ( " Entrez number of student present or Absent " ) ;
    scanf ( " %d " the number of Studen ) ;
    whielle ( " read the number of Attended sisons x " ) ;
    if ( x < A ) ;
    printf ( is Absent " ) ;
    if ( x > A ) ;
    printf ( " Student is present " ) ;
    for ( " solution Stop \ N are processed, the number of Absent stud reches
    }
    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 (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 (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°5

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

- ✓ Lecture de A avec scanf
 - ✓ Lecture de S avec scanf
 - ✓ Déclaration des variables
 - ✓ Utilisation d'une boucle for
 - ✓ Utilisation de structure conditionnelle
 - ✓ Utilisation d'incrémentation
-



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int w, A, s, n, i ;
    Print f ( " enter total number of registered students w : " ) ;
    Scanf ( " %d ", &w ) ;
    Print f ( " enter minimum attendance required A : " ) ;
    Scanf ( " %d ", &A ) ;
    Print f ( " enter absence threshold s : " ) ;
    Scanf ( " %d ", &s ) ;
    for ( i = 1 ; i <= A ; i ++ ) {
        if ( n < A ) {
            Print f ( " %d ", absent ) ;
        } else {
            Print f ( " %d ", present ) ;
        }
    }
    if ( w == A ) {
        Print f ( " %d \n", session valid ) ;
    } else {
        Print f ( " %d \n", session cancelled ) ;
    }
    return 0 ;
}
```



Analyse Détailée



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 (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 (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 (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. 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°6



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
7/20	15/20	+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

- ✓ 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
 - ✓ Affichages multiples
-



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int i, N, x, A, as = 0, ps = 0 ;
    printf ( " Enter N : " ) ; scanf ( " %d ", & N ) ;
    printf ( " Enter A : " ) ; scanf ( " %d ", & A ) ;
    printf ( " Enter x : " ) ;
    scanf ( " %d ", & x ) ;
    if ( x < A ) {
        write ( " absent " ) ;
    } else {
        write ( " present " ) ;
    }
    for ( i = x ; i < N || i = S ) {
        ps = ps + x ;
        as = N - PS ;
        i = i + 1 ;
    }
    printf ( " The lumber of absent students is : %d \n ", as ) ;
    printf ( " The number of presnt students is : %d \n ", ps ) ;
    if ( as > ps ) {
        write ( " session valid " ) ;
    } else {
        write ( " 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 (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
13/20	20/20	+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 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 ;
int x ;
int present = 0, absent = 0 ;
int i = 0
printf ( " (N) العدد الاحمالي للطلبة " ;
scanf ( " %d " , &N ) ;
printf ( " (A) العدد الادنى للحضور المطلوب " ;
scanf ( " %d " , &A ) ;
printf ( " (S) عتبة الغيابات المسموحة " ;
scanf ( " %d " , &S ) ;
while ( i < N && absen < S ) {
i ++
printf ( " \n%d - attended session " ,
i )
scanf ( " %d " , &x )
if ( x < A ) {
absent ++ ;
}
else {
present ++ ;
}
printf ( " Step %d : \n " , i ) ;
printf ( " حاضر : %d \n " , present ) ;
printf ( " غائب : %d \n " , absent ) ;
printf ( " \n-- النتيجة " ) ;
printf ( " معالجة حصو الطلبة " : %d \n " , i ) ;
printf ( " حاضر : %d \n " , present ) ;
printf ( " غائب : %d \n " , absent ) ;
if ( absent >= S ) {
printf ( " الامتحان ملغى " ) ;
} else {
printf ( " \n الامتحان صالح " ) ;
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 (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
4/20	16/20	+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
- ✓ Utilisation d'incrémentation



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S [
        scanf ( " %d, %d, %d" &N, &A, &S ) ;
        for ( i = 1, i <= N ) ;
        scanf ( " %d" & x ) ;
        if ( x < A ) ;
        N ++
        print f ( " طالب غائب " ) ;
        esse
        pint f ( " طالب حاضر " ) ;
        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 (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 (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

 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°9

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
2/20	15/20	+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

- ✓ 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, n = 0, m = 0 ; A, S, n
    Print F ( " Enter un mumber of attended sessions x " ) ;
    ( if ( x < A ) the )
    Scanf ( " % d % d % d ", & A, & N, & S ) ;
    for ( i = 1 ; i <= N, i ++ )
        Scanf ( " % d ", & n ) ;
    If ( n < A ) {
        n ++ ;
        Printf ( " a bsent % d ", n ) ;
    else
        m ++
        Printf ( " presht % d ", m ) ; }
    }
    Sum 1 = Sum 1 + 1 ; //
    else {
        Printf ( " Student nem % d " ) ;
        Sum 2 = Sum 2 + 1 ;
    }
    n ++ ;
    if ( Sum 1 > Sum 2 ) {
        Printf ( " Sessions cancelled " ) ;
    If ( m < S )
        Printf ( " الامتحان صالح " )
    else
        Printf ( " الامتحان ملغى " ) ;
    }
    returt 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°10

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
8/20	14/20	+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

- ✓ 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



```
# include < stdio.h >
int main ( )
int N, A, S ;
int x ;
int presente = 0 ;
int absents = 0 ;
Print F ( " inter N " ) ;
Scan F ( "%d ", & N ) ;
Print F ( " inter A " ) ;
Scan F ( "%d ", & A ) ;
Print F ( " inter S " ) ;
Scan F ( "%d ", & S ) ;
While ( i <= N & & absent < S ) {
Print F ( \
Scan F (
if ( x < A ) {
absent ++ ;
} else {
presents ++ ;
Print F ( " الغياب " %d \n " ;
Print F ( " عدد الحضور " %d \n " ;
Print F ( " عدد الغائبين " %d \n " ;
i ++ ;
Print F ( " Final inpure result " ;
Print F ( " عدد الطلبات الصالحة " %d \n " , i ) ;
Print F ( " عدد الحاضرين " ) ;
Print F ( " عدد الغائبين " ) ;
if ( absents >= S ) {
Print F ( " إمتحان ملغى " ) ;
else {
Print F ( " إمتحان يجرى عادى " ) ;
}
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 (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°11

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

- ✓ Déclaration des variables
- ✓ 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, i, Xi
    Print f ( " Entre a total number of registe students : " ) ;
    Scanf ( " %d ", & N ) ;
    Printf ( " Entre minimum attendance required : " ) ;
    Scanf ( " %d ", & Ali
    Printf ( " Entre absence threshold : " ) ;
    Scanf ( " %d ", & S ) ;
    while ( i < N && absent < S ) {
        Printf ( " Entre the number of attended sessions of student " i ) ;
        Scanf ( " %d ", & x ) i
        if ( x < A ) {
            absent ++ i
        } else {
            Present ++ i }
        Printf ( " present %d, Absent : %d ", i, present, Absent ) i
        Printf ( " final result : \n "
        Printf ( " Total processed students : %d ", i ) i
        Printf ( " present students : %d ", present ) i
        Printf ( " Absent Students %d ", Absents ) i
        return 0 i
    }
```



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 (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°12



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
9/20	17/20	+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

- ✓ 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 S, A, N ;
    int x ;
    int x = 0 ;
    present = 0 ; absent = 0 ;
    printf ( " enter the total number of student " ) ;
    scanf ( " %d " , & N ) ;
    printf ( " enter the minimum attendance requied " ) ;
    scanf ( " %d " , & A ) ;
    printf ( " enter the absence threshold " ) ;
    scanf ( " %d " , & S ) ;
    while ( i < N ; && absent < S ) ;
    printf ( " enter attended sessions for student %d " , i + 1 ) ;
    scanf ( " %d " , & x ) ;
    if ( x < A ) {
        absent ++ ;
    } else {
        present ++ ;
    }
    printf ( " step -> present : %d | absent %d " , present , absent ) ;
    printf ( " final result : \n " ) ;
    printf ( " total prosessed students : %d " , i ) ;
    printf ( " present students : %d " , present ) ;
    printf ( " alesent sutdents %d " , Absent ) ;
}
```

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°13

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

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Utilisation de structure conditionnelle



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, M, B, P, i ;
    print f ( " enter A, N, C : " ) ;
    Scanf ( " %d, %d, %d ", &A, &N, &C ) ;
    print f ( " enter x : " ) ;
    Scanf ( " %d ", &x ) ;
    print f ( " %d \ n ", x ) ;
    i = 0 ;
    while ( n < A ) {
        1) i = i - 1 ;
        12) p = -i ;
        13) B = N + i ;
        14) printf ( " present number student is : %d \ n ", B ) ;
        15) print f ( " absent student is : %d \ n ", P ) ;
        16) }
        17) if ( P < S ) {
        18) print f ( " the session valid " ) ;
        19) }
        20) else if {
        21) print f ( " session cancelled " ) ;
        22) }
        23) 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 (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 (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 (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. Comparer X avec A pour déterminer l'absence
 3. Utiliser des compteurs pour présents/absents
 4. Ajouter plus d'affichages pour suivre l'évolution
-

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°14

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
7/20	10/20	+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

- ✓ Déclaration des variables
 - ✓ Initialisation de variables à 0
 - ✓ Utilisation d'une boucle while
 - ✓ Utilisation de structure conditionnelle
 - ✓ Utilisation d'incrémentation
 - ✓ Affichages multiples
-



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int A, S, N;
    int DC = 1, B = 0, P = 0, V ;
    Scanf ( " %d ", & N ) ;
    Scanf ( " %d ", & S ) ;
    Scanf ( " %d ", & A ) ;
    while ( DC <= N )
        if ( B < S )
            if ( DC < A )
                P = P + 1 ;
            else
                B = B + 1 ;
        DC ++
        printf ( " %d ", DC ) ;
        printf ( " %d ", P ) ;
        printf ( " %d ", B ) ;
        else :
            printf ( " stop " ) ;
            V = P + B ;
            printf ( " %d ", P ) ;
            printf ( " %d ", B ) ;
            printf ( " %d ", V ) ;
            if ( B > S )
                printf ( " session cancelled " )
            else
                printf ( " session valid " )
    }
```



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

2. 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°15

Comparaison des Notes

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

- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ Comparaison X < A pour déterminer l'absence
- ✓ Utilisation d'incrémantation

Code Soumis

```
# include < Studio.h >
int main ( ) {
    int N, A, S, X, Sum, Sum'
    Scan f ( " %d " &, " %d " &, " %d " &, " %d " &, N, X, S, A ) ;
    int i = 1 ;
    For ( N > i ; Sum + = i ; i ++ ) {
        if ( x < A ) ;
        Sum = Sun + ;
    else :
        reeturn 0 ;
    } ;
    Sum' = N - Sum ;
    Print f ( " عدد الطلبة الغائبين " Sum ) ;
    Print f ( " عدد الطلبة الحاضرين " Sum' ) ;
    if ( Sum > S ) ;
        print f ( " اختبار ملغى " ) ;
    else
        print f ( " اختبار صالح " ) ;
    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 (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 (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 (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

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°16

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

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Comparaison X < A pour déterminer l'absence
- ✓ Utilisation d'incrémentation
- ✓ Affichages multiples



Code Soumis

```
include < stdio.h > ;
int main ( ) {
    int A, S, N, b = 0, P = 0, i = 0, intigre ;
    scanf ( " %d " ; &A ) ;
    printf ( " %d " ; A ) ;
    scanf ( " %d " ; &N ) ;
    printf ( " %d " ; N ) ;
    scanf ( " %d " ; &S ) ;
    printf ( " %d " ; S ) ;
    whail ( i = N ; b = S ; i ++ ) {
        pritf ( " d% " ; & i ) ;
        scanf ( " d% " ; & x ) ;
        pritf ( " d% " ; x ) ;
        if ( X < A ) {
            printf ( " الطالب غائب " )
            b <- b + 1 ;
            printf ( " d% " ; b ) ;
        } else
            printf ( " الطالب حاضر " )
            P <- P + 1
            printf ( " d% " ; P )
    } end if
} end whail
printf ( " d% " ; P )
printf ( " d% " ; b )
if ( b >= S ) {
    printf ( " الامتحان ملغى " )
} else
    printf ( " الامتحان صالح " )
} end if
return 0 ;
} end
```

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 (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 (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

1. Ajouter une boucle pour traiter N étudiants

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
5/20	13/20	+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

- ✓ 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, A, S, x = 1 ;
    float i = 0, j = 0 ;
    printf ( " Enter number of students : \n " ) ;
    printf ( " Enter the minimun attendance required : \n " ) ;
    printf ( " Enter the alesence thershold : " ) ;
    scanf ( " %d ", & N ) ;
    scanf ( " %d ", & A ) ;
    scanf ( " %d ", & S ) ;
    scanf ( " %d ", & x ) ;
    while ( x < A ) {
        if ( x < A ) {
            i = i + 1 ;
            printf ( " the number of alesent students is : %d ", i ) ;
        } else
            printf ( " the sessien cancelled " ) ; return 0 ;
        j = j + 1
        printf ( " the number of present students is : %d ", j ) ;
        printf ( " the session valid " ) ;
        return 0 ;
    }
    x = x + 1 ;
    printf ( " the student number : %d " i+j ) ;
    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°18

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
6/20	17/20	+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
 - ✓ Initialisation de variables à 0
 - ✓ Utilisation d'une boucle while
 - ✓ 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 ;
    int A, S ;
    int X, i ;
    Print f ( " entre N : Number Students " ) ;
    scanf ( " %d ", &N ) ;
    Print f ( " %d ", A minmum-attendance-required ) ;
    scan f ( " %d ", &A ) ;
    Print f ( " %d ", S absence-Students ) ;
    scan f ( " %d ", &S ) ;
    Pint f ( " %d ", X ) ;
    scan f ( " %d ", &X ) ;
    while ( X < A ) {
        if ( X < A ) {
            Print f ( " The students is absent \n " ) ;
            i ++ ;
        }
        else
            Print f ( " The students is Present \n " ) ;
        i ++ ;
    }
    if ( N = 0 )
        if ( S = Number Students absent ) {
            if ( Students absent = S ) {
                Print f ( " sesion cancelled ) ;
            }
            else
                Print f ( " session valid " ) ;
        }
    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 (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 (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
-

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
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'incrémentation
 - ✓ Affichages multiples
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N ;
    int A ;
    int S ;
    int i ;
    Printf ( " enter N ; " ) ;
    Scanf ( " %d ", &N ) ;
    Printf ( " enter A : " ) ;
    Scanf ( " %d ", &A ) ;
    Printf ( " enter S : " ) ;
    Scanf ( " %d ", &S ) ;
    for ( i = 1 ; i <= N ; i ++ ) {
        x = x + i ;
        if ( x < A ) { N = A % 10 .
            Print x = x % 10 ;
            x = x + i ;
        } if ( x < A ) {
            Printf ( " The student is considered absent \n " ) ;
        } else {
            Printf ( " The student is present \n " ) ;
        }
    } 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°20

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
 - ✓ 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, K, W ;
    Printf ( " Enter X : " ) ;
    Scanf ( "%d" , & X ) ;

    if ( X < A )
    {
        Printf ( " absent students " ) ;
    }
    else {
        Printf ( " present students " ) ;
    }
    for ( i = 0 ; i == N && i >= S ; i ++ ) .
    { scanf ( " %d " , i ) ;
        Printf ( " Enter i : " \n ) ;
        K == i
        Scanf ( " %d " , K ) ;
        Printf ( " Enter k = " \n ) ;
        W == N - K
        Scanf ( " %d " , W ) ;
        Printf ( " Enter w = " \n ) ;
    }
    if ( K >= A ; ) {
        Printf ( " Session Valid " ) ;
    }
    else {
        Printf ( " Session celled " ) ;
    }
    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

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°21

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
12/20	13/20	+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 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, P = 0, ab ;
    Print f ( " Total number of registered students N = " ) ;
    Scan f ( "%d ", &N ) ;
    Print f ( " minimum attendance required A = " ) ;
    Scan f ( "%d ", &A ) ;
    Print f ( " absence threshold S = " ) ;
    Scan f ( "%d ", &S ) ;
    for ( int i = 1 ; i <= N ; i ++ ) {
        Print f ( " enter the number of attended session for student %d x = ", i ) ;
        Scan f ( "%d ", &X ) ;
        if ( X < A ) {
            Print f ( " The student %d is absent ", i ) ;
            ab = N - P ;
            Print f ( " absent students %d ", ab ) ;
        } else {
            Print f ( " The student %d is present ", i ) ;
            P = P + 1 ;
            Print f ( " Present students %d ", P ) ;
        }
    }
    Print f ( " Present students %d ", P ) ;
    Print f ( " Absent students %d ", ab ) ;
    if ( ab >= 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°22

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
10/20	15/20	+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

- ✓ 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'incrémantation
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



Code Soumis

```
int main ( ) {  
    int N, A, S, X, D = 0, B = 0, i = 0  
    scanf ( " %d %d %d ", & N, & A, & S ) ;  
    do {  
        scanf ( " %d ", & X ) ;  
        if ( X < A ) {  
            B ++ ;  
        } else { D ++ ; }  
        i ++ ; printf ( " %d %d %d ", i, B, D ) ;  
    } while ( B != S || i != N )  
    printf ( " %d %d %d ", i, B, D ) ;  
    if ( B => S ) {  
        printf ( " session cancelled " ) ; }  
    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.

1
2
3
4

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

 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	16/20	+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
- ✓ 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
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, X, i = 0 ;
    printf ( " Enter a number of students " ) ;
    scanf ( " %d " , & N ) ;
    printf ( " Enter a number of minimum attendance required " ) ;
    scanf ( " %d " , & A ) ;
    printf ( " Enter the absence threshold " ) ;
    scanf ( " %d " , & S ) ;
    while ( i < N && i < S )
        printf ( " Enter the number of attended sessions " ) ;
        scanf ( " %d " , & X ) ;
        i <- i + 1
        if ( X < A )
            printf ( " the student is absent " , A ++ ) ;
        else
            printf ( " the student is Present " , P ++ ) ;
        if (
            printf "
            end while .
        if ( Error not exict )
            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 (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 (0/1 pt)

Ancienne note: 0/3

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

Suggestions d'Amélioration

1. Ajouter plus d'affichages pour suivre l'évolution
-

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