

Corrections Encourageantes - doc20260115222309

Nombre de copies: 26

Moyenne ancienne correction: 4.69/20

Moyenne nouvelle correction: 13.58/20

Différence moyenne: +8.88

Copies améliorées: 25/26

Correction Encourageante - Copie N°1



Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
6/20	15/20	+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
 - ✓ 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, y, z ;
    printf ( " Entre N, A, S : " ) ;
    scanf ( " %d, & N, %d A, %d S, %d x " ) ;
    for ( int i = 0 ; i < N ; i ++ ) {
        if ( x < A ) {
            y = y + 1 ;
            printf ( " the student is
            considred a bsent " ) ;
        } Else { z = z + 1 ;
            printf ( " the student is present " ) ;
        }
        if ( y >= S ) {
            printf ( " ( إمتحان ملغى " ) ;
            return 0 ;
        } Else { printf ( " ( إمتحان صالح " ) ;
            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°2

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
7/20	16/20	+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
- ✓ 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>
# include <math.h>
int main ( ) {
    int N, A, S ;
    int x ;
    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 ( " the student is considered
absent " ) ;
        } else {
            printf ( " the student is present " ) ;
        }
    }
    while ( N = ! S )
        S = S + 1
    i ++ ;
    if ( S > عدد الطلبة الحاضرين ) {
        printf ( " الامتحان ملغى " ) ;
    } else {
        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 (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°3



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

- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N

- ✓ Tentative de gestion du seuil d'absence
 - ✓ Comparaison $X < A$ pour déterminer l'absence
 - ✓ Structure if/else pour présence/absence
 - ✓ Affichages multiples
-



Code Soumis

```
# include < stdio.h >
} ( ) int main
int N, A, S ;
int x ;
int i ;
int presents ;
int absents ;
Scanf ( "%d" & N )
Scanf ( "%d" & A )
Scanf ( "%d" & S )
i = 1;
presents = 0 ;
abe sents = 0 ;
} While ( i <= N && absents < S )
Scanf ( "%d" & x )
} if ( x < A )
ab sents = absents + 1 ;
} else {
presents = presents + 1 {
} i = i + 1 {
printf ( " total troutes :
% d \n ", presents +
abesents ) ;
Print f ( " presents :
% d \n ", presents ) ;
Print f ( " A besents :
% d \n ", abesents ) ;
} if ( absents >= S )
printf ( " Sessicem
anulee \n " ) ;
} else {
printf ( " Sessicem
Valide \n " )
{
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 (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

-  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
5/20	16/20	+11

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



Points Forts

- ✓ 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
- ✓ Affichages multiples



Code Soumis

```
# include < stdio .h >
int main ( ) {
char students [ N ] ;
int A, S, N ;
int i = 0, x ;
for ( i = 0 , i <= N , i ++ ) {
if ( x < A ) {
Printf ( " the student [i] is considred absent " ) ;
Scanf ( " %d ", student [i] .
" %d ", & A,
" %d ", & S ) ;
}
else {
Printf ( " the student [i] is presnt " ) ;
}
Printf ( " Number of presnt students is : %d \n ", pesents studets ) ;
Printf ( " Number of abest studens is : %d \n ", absents students ) ;
if ( N == S && N == ) {
Printf ( " can 't continue " ) ; } else { Printf ( " continue " ) ; }
if ( x >= A ) {
Printf ( " Session valid " ) ; }
else {
Prinf ( " 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 (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°5

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 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 a, n, s ;
    int x, i = 0 ;
    int present - cont = 0 ;
    int absent - cont = 0 ;
    printf ( " tatal Number of registeed stusents (N) : " ) ;
    Scanf ( " %d ", & N ) ;
    printf ( " min attendance requerd (A) : " ) ;
    Scanf ( " %d ", & A ) ;
    printf ( " absence theshold (S) : " ) ;
    Scanf ( " %d ", & S ) ;
    while ( i < N && absent - cont < S ) {
        i ++ ;
        printf ( " x : " ) ;
        Scanf ( " %d ", & x ) ;
        if ( x < A ) {
            absent cont ++
            printf ( " absent " ) ;
        } else { present cont ++
            printf ( " presnt " ) ; }
        if ( absent - cont >= S ) {
            printf ( " Final Staut : Session cancelled " ) ;
        } else { pirtf ( " Final Statut : Session valide " ) ;
        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°6

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
 - ✓ 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 ( ) {
int N, A, S ;
print ( " Enter N " ) ;
Scanf ( " % d ", & N ) ;
print ( " Enter A " ) ;
Scanf ( " %d ", & A ) ;
print f ( " Enter S " ) ;
Scanf ( " %d ", & S ) ;
for ( i = 1 , x < A , i ++ ) {
if ( x < A ) {
print f ( " the student is absent " ) ;
}
else ( x > A ) {
print f ( " the student is present " ) ;
}
for ( i = 1 ; i <= N , i ++ ) {
Present = N - absent ;
absent = N - Present ;
print f ( " % d ", Present ) ;
wrint ( " % d ", absent " ) ;
print ( " %d , A step number ) ;
}
if ( present < A ) {
print f ( " Session Valid " ) ;
}
else ( Present < A ) {
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 (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°7

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
- ✓ 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, M, K ;
    printf ( " (N) " ; ( ) " ادخل العدد الاجمالي للطلبة المسجلين (N) " ;
    scanf ( " %d " , & N ) ;
    printf ( " (A) " ; ( ) " ادخل الحد الادنى للحضور المطلوب (A) " ;
    scanf ( " %d " , & A ) ;
    printf ( " (S) " ; ( ) " ادخل عتبة الغيابات المسموح بها (S) " ;
    scanf ( " %d " , & S ) ;
    for ( i = 1 ; i <= N ; i ++ ) {
        scanf ( " %d " , & X ) ;
        if ( A > X ) {
            K ++ ;
            printf ( " ( الطالب غائب ) " ) ;
        } else { M ++ ;
            printf ( " ( الطالب حاضر ) " ) ;
        }
    }
    يتوقف عملية المحاكاة // {
    if ( N == 11 ) معالجة الطلبة .
    S == { معالجة الطلبة {
        printf ( " % م الطلبة الحاضرين " ) ;
        printf ( " % الطلبة الغائبين " ) ;
        printf ( " ( ) الامتحان صالح " ) ;
        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 (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°8

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
- ✓ 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 A, N, S, x, P = 0, F = 0 ;
printf ( " enter A " ) ;
scanf ( " %d " , & A ) ;
printf ( " enter N " ) ;
scanf ( " %d " , & N ) ;
printf ( " enter S " ) ;
scanf ( " %d " , & S ) ;
for ( i = 1 , i <= N , i ++ ) {
printf ( " enter x " ) ;
scanf ( " %d " , & x ) ;
if ( x < A ) {
printf ( " absent " ) ; F = F + 1 ;
} else { printf ( " %d present " , i ) }
P = P + 1 ;
}
if ( A > S ) {
printf ( " valid session " ) ;
} else {
printf ( " invalid session " ) ;
}
```

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°9

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
2/20	14/20	+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

- ✓ 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
- ✓ Affichages multiples



Code Soumis

```
# include <stdio.h>
int main ( ) {
    int i, N, A, S, x ; n = 0, m = 0 ;
    Scanf ( " %d ", & N ) ;
    Scanf ( " %d ", & S ) ;
    Scanf ( " %d ", & A ) ;
    while ( N == A || A < N ) {
        if ( x < A ) {
            Scanf ( " %d ", & x ) ;
            n = S - 1 ; n = Si
            Printf ( " عدد الطلاب غائبين " = %d , S ) ;
        } else
            m = N - 1 ; m = N ;
        Printf ( " : عدد الطلبة الحاضرين " = %d , N ) ;
    }
    if ( A > S ) {
        Printf ( " الامتحان صالح " ) ;
    } else
        Printf ( " الامتحان ملغى " ) ;
    i ++ ; }
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 (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°10

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 N, A, S ;
int x ;
int present = 0, absent = 0 ;
int i = 1 ;
printf ( " Enter total number of students : N " ) ;
scanf ( " %d ", & N ) ;
Printf ( " A : " ) ; // إدخال الحد الأدنى للحضور المطلوب
scanf ( " %d ", & A ) ;
Printf ( " S : " ) ; // إدخال عتبة الغيابات المسموح
scanf ( " %d ", & S ) ;
while ( i <= N && absent < S ) {
Printf ( " Enter attended sessions for students %d = i " ) ;
Scanf ( " %d ", & x ) ;
if ( x < A ) {
absent ++ i
} else {
present ++ i
Printf ( " step % d : \n ", i ) ;
```



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
0/20	3/20	+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

- ✓ Utilisation de structure conditionnelle
- ✓ Utilisation d'incrémentation



Code Soumis

```
Prinf ( " Present students = %d \n ", Present ) ;
Prinf ( " Absent students : %d \n ", absent ) ;
i ++ ;
Prinf ( " Final Results : \n " ) ;
Prinf ( " Total processed students : %d \n ", i-- 1 ) ;
Prinf ( " Present students : %d \n ", Present ) ;
Prinf ( " Absent students : %d \n ", absent ) ;
if ( absent < F ) {
Prinf ( " sessions valid \n " ) ;
} else {
prinf ( " sessions cancelled \n " ) ;
}
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 (0/3 pts)

Ancienne note: 0/3

N'oubliez pas de déclarer vos variables avec `int` et de les initialiser à 0.



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 (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).

1
2
3
4

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



Message d'Encouragement

 C'est un début ! La programmation demande de la pratique. Revoyez les exemples de cours sur les boucles et les conditions. N'hésitez pas à demander de l'aide à votre enseignant.



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
2/20	16/20	+14

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



Points Forts

- ✓ 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
 - ✓ Affichages multiples
-



Code Soumis

```
# include <stdio.h>
int main {
    int N, A, S, X, i ;
    Sum of prusent students = 0 ;
    Sum of absent students = 0 ;
    printf ( " total number of registed Student : %d ", N ) ;
    Scanf ( " %d \n ", N ) ;
    printf ( " minimum attendence required : %d ", A ) ;
    Scanf ( " %d \n ", A ) ;
    printf ( " absence threshold : %d ", S ) ;
    Scanf ( " %d \n ", S ) ;
    for ( i = 1 ; i <= N || Sum of absence Student = S , i ++ ) ;
    { printf ( " Student Number : %d \n ", i ) }
    Scanf ( " %d ", X ) ;
    if ( X < A )
    { Sum of absent student = Sum of absent studen + 1 ;
    printf ( " Sum of absent studen = %d \n ", Sum of absent student ) ;
    print f ( " session cansselod " )
    else
    {
        Sum of present student = Sum of present student + 1 ;
        printf ( " Sum of present student : %d \n ", Sum of present student ) ;
        printf ( " Session Valid " )
        if printf ( " present students is : %d \n ", Sum of present student ) ,
        printf ( " absent students is : %d \n ", Sum of absent student ) ,
        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°13

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 ( ) {
long N, A, S, X, Ap, Pr ;
prints ( " total number of registered student : " ) ;
Scanf ( " %ld ", & N ) ;
prints ( " minimum attendance required : " ) ;
Scanf ( " %ld ", & A ) ;
prints ( " absence Threshold " ) ;
Scanf ( " %ld ", & S ) ;
prints ( " read The number of attended sessions : " ) ;
Scanf ( " %ld ", & X ) ;
for ( int i = 0 ; i < N ; i ++ ) {
prints ( " i " ) ;
if ( X < A )
prints ( " session canceled " ) ;
Sum = Ap ++ ;
else prints ( " Session valid " ) ;
Sum = Pr ++ ;
prints ( " present student %ld \n ", Sum ) ;
prints ( " absent student %ld ", Sum ) ;
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°14



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

- ✓ 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 ;
    printf ( " Enter N " ) ;
    scanf ( " %d ", & N ) ;
    printf ( " Enter A " ) ;
    scanf ( " %d ", & A ) ;
    printf ( " Enter S " ) ;
    scanf ( " %d ", & S ) ;
    for ( int i = 1 ; i <= N ; i ++ ) {
        scanf ( " %d ", & x ) ;
        if ( x < A )
            {
                G = G + 1 ;
            }
        else
            { H = H + 1 ;
            }
        printf ( " %d - present = %d - absent = %d ", i , H, G )
        if ( G > S )
            {
                i = i + N ;
                printf ( " present = %d /n absent = %d ", H, G ) ;
                if ( G < 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 (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°15

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
- ✓ Condition de boucle sur N
- ✓ Comparaison $X < A$ pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémantation
- ✓ Affichages multiples



Code Soumis

```
# include <stdio.h>
int main {
    int N, A, S, X, P, a, n, i ;
    printf ( " ادخل عدد الطلبة الإجمالي " ) ;
    scanf ( " %d " , & N ) ;
    printf ( " ادخل الحد الأدنى للحضور المطلوب " ) ;
    scanf ( " %d " , & A ) ;
    printf ( " ادخل عتبة الغياب المسموحة " ) ;
    scanf ( " %d " , & S ) ;
    for ( i = 0 ; i <= N ; i ++ ) {
        printf ( " أدخل عدد الحصص التي حضرها الطالب " ) ;
        scanf ( " %d " , & x ) ;
        if ( X < A ) {
            printf ( " الطالب حاضر " ) ;
        }
        else
            printf ( " الطالب غائب " ) ;
        printf ( " ادخل عدد الطلبة الحاضرين " ) ;
        scanf ( " %d " , & a ) ;
        printf ( " ادخل عدد الطلبة الغائبين " ) ;
        scanf ( " %d " , & P ) ;
        if ( P > A ) {
            printf ( " الامتحان ملغى " ) ;
        }
        else
            printf ( " الامتحان صالح " ) ;
    }
    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°16

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
0/20	13/20	+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
- ✓ 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, sum 1
    printf ( " enter the Total number of the Students " ) ;
    scanf ( N ) ;
    printf ( " enter the minium attendunce required " ) ;
    Scanf ( A ) ;
    printf ( " enter the absence thereshold " ) ;
    Scanf ( S ) ;
    while ( 1 ) {
        printf ( " enter the Student number " ) ;
        Scanf ( N ) ;
        printf ( " enter the number of the attended sessions " ) ;
        Scanf ( X ) ;
        if ( X < A ) { printf ( " the student is consider absent " ) ; }
        else { printf ( " the student is present " ) ; }
    }
    Sum 1 = X + 1
    if ( Sum 1 > A ) { printf ( " the session valid " ) ; }
    else { prinf ( " the session concelled " ) ; }
```

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°17

Comparaison des Notes

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

- ✓ 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

```
# include < stdio.h >
int main ( ) { int, A, N, X, S ;
for ( int i = 0 ; i >= N ; i ++ ) {
if ( X < A ) { printf ( " غائب " ) ;
else
printf ( " حاضر " ) ;
X = N
N = S
printf ( "
}
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 (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°18

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
12/20	14/20	+2

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
- ✓ Affichages multiples



Code Soumis

```
# include < stdio.h >
int main ( )
{ int N, A, S, gh = 0, ha = 0, X ;
print f ( " enter N total Number of registered " ) ;
scanf ( " %d ", & N ) ;
print f ( " enter minimum attendanc1 A " ) ;
scanf ( " %d ", & A ) ;
print f ( " enter absence there shold " ) ;
scanf ( " %d ", & S ) ;
for ( int i = 1 ; i <= N ; i ++ )
{
scanf ( " %d ", & X ) ;
if ( X < A )
{
gh = gh + 1 ;
}
else {
ha = ha + 1 ;
}
printf ( " %d - present = %d - absent = %d ", i, ha, gh
if ( gh > S )
{
i = i + N ;
}
}
printf ( " present = %d \n absent = %d \n ", ha, gh
if ( gh <= S )
printf ( " Session Valide " ) ;
else {
printf ( " Sessia cancelled " ) ;
```

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 (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

- 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°19

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
2/20	12/20	+10

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



Points Forts

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



```
# include < stdio.h >
int main ( ) {
    int N, S = 3, A, absent, present ;
    int n, step, total, sun 1 = 0, sun 2 = 0 ;
    Print f ( " enter the total number of registered students : " ) ;
    scanf ( " %d ", &N ) ;
    Print f ( " enter the number of student : " ) ;
    scanf ( " %d ", &step ) ;
    while ( n != S ) {
        Print f ( " enter the number of student : " ) ;
        scanf ( " %d ", &n ) ;
        if ( n < A ) {
            sun 1 = sun 1 + Present ;
            Print f ( " the student is present " ) ;
            scanf ( " %d ", &present ) ;
        }
        else {
            sun 2 = sun 2 + absent ;
            Print f ( " the student is absent : " ) ;
            scanf ( " %d ", &absent ) ;
        }
        if ( N <= sun 2 ) {
            Print f ( " session cancelled " ) ;
        } else {
            Print f ( " session valid " ) ;
        }
        return 0 ;
    }
    Print f ( " Present students is : %d ", sun 1 ) ;
    Print f ( " absent students is : %d ", sun 2 ) ;
    total = sun 1 + sun 2 ;
    Print f ( " total Processed student is %d ", total ) ;
```

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 (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

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°20

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	12/20	-2

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
- ✓ Structure if/else pour présence/absence



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, a, P, X, i = 1 ;
    Print f ( " Enter number of registered Student " ) ;
    Scanf ( "%d", &N ) ;
    Print f ( " Enter minimum number of attended required " ) ;
    Scanf ( "%d", &A ) ;
    Print f ( " Enter absence threshold " ) ;
    Scanf ( "%d", &S ) ;
    whil ( N != 0 && a < S ) {
        Print f ( " Enter number of attended Session of the Student number %d ", i )
        Scanf ( "%d", &X ) ;
        if ( X < A ) { a = a + 1 }
        else { P = P + 1 } .
        i = i + 1 ;
        N = N - 1 ;
        Print f ( " Student number : %d /, Present Students : %d /, absent Students :
i, P, a ) ;
        Print f ( " \n " ) ; }
        Print f ( " number of total processed Studies : %d \n ", i ) ;
        Print f ( " Present Students : %d \n ", P ) ;
        Print f ( " absent students : %d \n ", a ) ;
        if ( a >= S) { Print f ( " Session Valid " ) }
        else { Print f ( " 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 (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 (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. Ajouter une boucle pour traiter N étudiants
 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°21

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
- ✓ Utilisation d'une boucle while
- ✓ 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 ;
    int i ;
    while ( A = 1, A > = S, A ++ ) {
        printf ( " Enter number : " ) ;
        scanf ( " %d " , & X ) ;
        if ( X < A ) do
            printf ( " the studen absant " ) ;
        else
            printf ( " the student prsent " ) ;
        } at }
        if ( A >= S ) {
            printf ( " the session Valid " ) ;
        else {
            printf ( " 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 (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°22

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 for
- ✓ 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 ;
    Print ( " enter the Nuber S " ) ;
    Print f ( " entor the Nuber A " ) ;
    Print f ( " enter the Number S " ) ;
    for ( i = N ; A < i ; i ++ ) {
        if X < A
            Scanf ( " The student is absent " ) ;
        else
            ( " Student is Present " ) .
        Scanf ( Sum of Present ) ;
        Scanf ( " Sum of absent " ) ;
        Print f ( Sesseen Valide ) ;
        Print f ( sesseen comselled ) ; }
    }
    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 (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

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°23



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

- ✓ 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
- ✓ Utilisation d'incrémentation



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int N, A, S, X, i = 1, Tous, Absent, Ti, Fi ;
    int Ti = 0 ; عدد الطلبة الغائبين F = 0 ، عدد الطلبة الحاضرين
    print f ( " enter the total number of registered students " ) ;
    scanf ( " %d ", & N ) ;
    while ( i < N || i < S ) {
        i ++ ;
        scanf ( " %d ", & X ) ;
        X = Tous - Absent ;
        if ( X < A ) {
            print f ( " the student is considered absent " ) ;
            print f ( " %d ", Ti ) ;
        }
        if ( X > A ) {
            print f ( " the student is present " ) ;
            print f ( " %d ", Fi ) ;
        }
        Ti + Fi = عدد الطلبة الحاضرين
        Fi + F = عدد الطلبة الغائبين
    }
    print f ( " عدد الطلبة الذين تمت معالجتهم " ) ;
    print f ( " T / % d " ) ;
    print f ( " F / % d " ) ;
    if ( S > F ) {
        print f ( " الإمتحان صالح " ) ;
    }
    if ( S < F ) {
        print f ( " الإمتحان ملغى " ) ;
    }
}
```

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 (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

- 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	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

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ 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

```
# includ < stdio.h >
int main ( ) {
    int n ;
    int a ;
    int s ;
    printf ( " enter the total number of registeres students " ) ;
    scanf ( & n ) ;
    printf ( " enter the minimum attenslance required " ) ;
    scanf ( & a ) ;
    printf ( " enter the absence thresholds " ) ;
    scanf ( & s ) ;
    int n = 1, absent, present ;
    for ( n = 1, ( N = 1 && ; absent = s ) , i ++ ) {
        printf ( " enter the number of attended sessions " ) ;
        scanf ( & x ) ;
        if ( A > x ) then
            printf ( " student is absent " ) ;
        else
            printf ( " student is present " ) ;
        and if }
        printf ( " present students is %d \n ", present ) ;
        printf ( " absent students is %d \n ", absent ) ;
        if ( ) then
            sessen valis
        else
            sessen conselles
        and if
    retuen 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°25

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
1/20	10/20	+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
 - ✓ Comparaison $X < A$ pour déterminer l'absence
 - ✓ Structure if/else pour présence/absence
 - ✓ Affichages multiples
-



Code Soumis

```
# include < stdio.h >
int main ( ) {
    int A, N, S, x, B, C
    printf ( " أدخل N " ) ;
    scanf ( % d, & N ) ;
    printf ( " أدخل A " ) ;
    scanf ( % d, & A ) ;
    printf ( " أدخل S " ) ;
    scanf ( % d, & S ) ;
    printf ( " أدخل x " ) ;
    scanf ( % d, & x ) ;
    if ( x < A )
        printf ( " الطالب حاضر " ) ;
    else
        printf ( " الطالب غائب " ) ;
    Sum ( x < A ) = B ;
    B = عدد طلبة غائبين
    C = N - B ;
    C = عدد طلبة حاضرين
    printf ( " B " ) ;
    printf ( " c " ) ;
    if ( B > S )
        printf ( " امتحان صالح " ) .
    else
        printf ( " امتحان 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 (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 (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. Ajouter une boucle pour traiter N étudiants
 2. Utiliser des compteurs pour présents/absents
-

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°26

Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
1/20	8/20	+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 de structure conditionnelle
 - ✓ Utilisation d'incrémentation
 - ✓ Gestion de plusieurs compteurs
 - ✓ Affichages multiples
-



Code Soumis

```
# include < stdio.h >
int main ( )
int N, S ;
int present = 0 ;
int absenet = 0 ;
int i = 1 ; status ;
Scanf ( % d, & ) printf ( " entre total number n " ) ;
Will ( i <= 88 absente )
printf ( " student % d ( 1 = prstent ; 0 = absent " ) .
scanf ( % d ; & status ) ;
if ( status == 1
pustent == j
ense
absent ++ i
prints ( " step % d > prusent % d / absent : % d \n " i ; prsent, absent )
i ++ )
printf ( " \n final out put : \n " ) ;
printf ( " total prolessted studnt % d \n ; preset + abesent ) .
printf ( " present stednt : % d \n ", present ) ;
if ( absent == 28 ) printf ( absent studnt : % d \n " , absent ) ;
printf ( " session : can clud / n " ) ;
else
printf ( " session : valid ( n " ) ;
retem 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 (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. Ajouter une boucle pour traiter N étudiants
 2. Comparer X avec A pour déterminer l'absence
-

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