

# Corrections Encourageantes - doc20260115221134

Nombre de copies: 22

Moyenne ancienne correction: 10.05/20

Moyenne nouvelle correction: 17.50/20

Différence moyenne: +7.45

Copies améliorées: 18/22

## Correction Encourageante - Copie N°1



### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	20/20	+15

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



### Points Forts

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

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



## Code Soumis

```
#include <stdio.h>

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

    for (int i = 0; i <= N || i == S; i++)
    {
        if (int x < A)
        {
            i = S;
            printf("the student is absent");
            cont++;
        }
        else
        {
            printf("the student is present");
            cont++;
        }
    }

    printf("the absent students are: %d", cont);
    printf("the present students are: %d", con);
    printf("the number student is: ");

    if (N > S)
    {
        printf("Session Cancelled");
    }
    else
    {
        printf("Session Valid");
    }
}
```

```
}  
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 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: 1/4

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: 1/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°2



## Comparaison des Notes

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

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N = 0;
    int A = 0;
    int S = 0;
    int absent = 0;
    int present = 0;
    int x = 0;

    printf(N, A, S);
    scanf(N, A, S);

    for (int i = 1; i <= N; i++)
    {
        printf("entre x");
        scanf("%x");
        if (x < A)
        {
            printf("absent");
            absent++;
        }
        else
        {
            printf("present");
            present++;
        }

        printf("number of ... x");
        scanf("%d", &x);

        if (x < A)
        {
            printf("absent");
            absent++;
        }
        else
```

```
        {
            printf("present");
            present++;
        }

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

    printf("total n of student");
    printf("present");
    printf("absent");

    if (absent >= S)
    {
        printf("Valid");
    }
    else
    {
        printf("cancelled");
    }
    return 0;
}
```

## Analyse Détaillée

### Lecture des données (1/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 3/3



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: 2/4*

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

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

*Ancienne note: 2/4*

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

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

*Ancienne note: 3/3*

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

### **Affichages (1/1 pt)**

*Ancienne note: 3/3*

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
6/20	20/20	+14

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



### Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables

- ✓ Utilisation d'une boucle for
  - ✓ Condition de boucle sur N
  - ✓ Tentative de gestion du seuil d'absence
  - ✓ Comparaison  $X < A$  pour déterminer l'absence
  - ✓ Structure if/else pour présence/absence
  - ✓ Utilisation d'incrémentations
  - ✓ Gestion de plusieurs compteurs
  - ✓ Affichages multiples
-



## Code Soumis

```
#include <stdio.h>

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

    for (i = 1; i <= N || B == S; i++)
    {
        scanf("%d", &X);
        if (X < A)
        {
            count++;
        }
        else
        {
            count++;
        }
    }

    printf("%d", i);
    printf("the total of present students is: %d", P);
    printf("the total of absent students is: %d", B);

    if (B < P || B == P)
    {
        printf("Session Valid");
    }
    else
    {
        printf("Session Cancelled");
    }
}
```

```
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 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: 1/4

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

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

Ancienne note: 2/4

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

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

Ancienne note: 0/3

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

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°4



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
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

- ✓ 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
- ✓ Utilisation de structure conditionnelle
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

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

    printf("Enter the number of attended sessions");
    scanf("%d", &X);

    if (X >= A)
    {
        printf("the student is present");
        for (i = 0; i <= N; i++)
        {
            printf("present student");
            present = i;
        }
    }
    else
    {
        printf("the student absent");
        absent = N - i;
        printf("%d", &absent);
    }

    if (i >= S)
    {
        printf("Session cancelled");
    }
    else
    {
        printf("Session Valid");
    }
}
```



```
}  
return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 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: 1/4

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

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

Ancienne note: 0/3

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

## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

1. Comparer X avec A pour déterminer l'absence

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°5



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	17/20	+12

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



## Points Forts

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



## Code Soumis

---

```
#include <stdio.h>

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

    printf("Enter the number of attended sessions");
    scanf("%d", &X);

    for (int i = 0; i < X; i++)
    {
        if (X < A)
        {
            printf("the student is absent");
        }
        else
        {
            printf("the student is present");
        }
    }
    return 0;
}
```



## Analyse Détaillée

---



### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

### Gestion des compteurs (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°6

## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
18/20	20/20	+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
  - ✓ Utilisation d'une boucle for
  - ✓ Condition de boucle sur N
  - ✓ Tentative de gestion du seuil d'absence
  - ✓ Comparaison  $X < A$  pour déterminer l'absence
  - ✓ Structure if/else pour présence/absence
  - ✓ Utilisation d'incrémentations
  - ✓ Gestion de plusieurs compteurs
  - ✓ Affichages multiples
-



## Code Soumis

```
#include <stdio.h>

int main()
{
    int n, a, s, x, i;
    int H = 0;
    int P = 0;
    int f = n; // fis a counter

    printf("enter number of student");
    scanf("%d", &n);
    printf("enter number of minimum attendance required");
    scanf("%d", &a);
    printf("enter number of absence threshold");
    scanf("%d", &s);

    do
    {
        for (i = 1; i <= N; i++)
        {
            printf("enter thenumber of the attended sessions for the stude
            scanf("%d", &x);
            if (x < a)
            {
                printf("the student number %d is absence", i);
                H++;
            }
            else
            {
                printf("the student is not absence %d, it's number", i);
                P++;
            }
        }
    } while (f != 0 || H < S);

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



```
    }  
    else  
    {  
        printf("session valid");  
        printf("%d absences", H);  
        printf("%d presences", P);  
    }  
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 3/3

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: 2/4

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

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

Ancienne note: 4/4

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: 3/3

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



## Affichages (1/1 pt)

Ancienne note: 3/3

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°7



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
11/20	18/20	+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 for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, x, j = 0, K = 0;
    printf("enter the number of registered students");
    scanf("%d", &N);
    printf("enter the minimum attendance required");
    scanf("%d", &A);
    printf("enter the absence threshold");
    scanf("%d", &S);

    for (int i = 0; i <= n; i++)
    {
        printf("give me the number of sessions of student number %d", i);
        scanf("%d", &x);
        if (x < A)
        {
            printf("the student number %d is absent", i);
            j = j + 1;
        }
        else
        {
            printf("the student number %d is present", i);
            K = K + 1;
        }
    }

    printf("the number of present student is %d", K);
    printf("the number of absent student is %d", j);

    if (j >= S)
    {
        printf("session cancelled");
    }
    else
    {
        printf("session valid");
    }
}
```

```
}  
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 2/3

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

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

Ancienne note: 1/3

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

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

Ancienne note: 1/4

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

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

Ancienne note: 3/4

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: 3/3

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: 1/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°8



## 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'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

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

    while (i <= N && Z != S)
    {
        printf("What is the number of attended session X");
        scanf("%d", &X);
        if (A > X)
        {
            printf("the student is absent");
            Z++;
        }
        else
        {
            printf("the student is present");
            i++;
        }
    }

    Y = N - Z;
    printf("total processed student = %d", N);
    printf("the number of present students = %d", Y);
    printf("the number of absent students = %d", Z);

    if (Z == S)
    {
        printf("Session cancelled");
    }
}
```



```
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 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: 4/4

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

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

Ancienne note: 2/4

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

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

Ancienne note: 3/3

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

## Affichages (1/1 pt)

Ancienne note: 1/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°9



## 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

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, i;
    printf("total of processed students: ", N);

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

    for (i = 0; i <= N; i++)
    {
        if (N == A || N++)
        {
            printf("Session valid");
        }
        if (not)
        {
            printf("Session cancelled");
        }
    }
    return 0;
}
```



## Analyse Détaillée



### Lecture des données (0/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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°10

### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
20/20	19/20	-1

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



## Points Forts

---

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X, NS = 0, NA = 0;
    printf("total number of registered students ");
    scanf("%d", &N);
    printf("enter minimum attendance required ");
    scanf("%d", &A);
    printf("enter absence threshold ");
    scanf("%d", &S);

    do
    {
        int i = 1;
        printf("enter the number of attendance of student %d", i);
        scanf("%d", &X);
        if (X < A)
        {
            printf("student is absent");
            NS = NS + 1;
        }
        else
        {
            printf("student is present");
            NA = NA + 1;
        }
        i++;
    } while (i <= N || NS < S);

    printf("the number of present student is %d", NA);
    printf("the number of absent student is %d", NS);

    if (NS >= S)
    {
        printf("the session is cancelled");
    }
    else
```



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

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 3/3

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: 4/4

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

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

Ancienne note: 4/4

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: 3/3

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: 3/3*

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

Continuez à pratiquer et à consolider vos acquis !

---

## Message d'Encouragement

---

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

---

## Détails de l'Ancienne Correction

---

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

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°11



## Comparaison des Notes

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

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, Sum1 = 0, Sum2 = 0;
    scanf("%d", &N);
    scanf("%d", &A);
    scanf("%d", &S);

    for (int i = 1; i <= N || S > N; i++)
    {
        int x;
        scanf("%d", &x);
        if (x < A)
        {
            printf("absent student");
            Sum1 = Sum1 + Suma;
        }
        else
        {
            printf("present student");
            Sum2 = Sum2 + Suma;
        }

        printf("the number of absent students = %d", Sum1);
        printf("the number of present student = %d", Sum2);

        if (Sum1 > Sum2)
        {
            printf("session cancelled");
        }
        else
        {
            printf("session valid");
        }
    }
    return 0;
}
```

---

## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

### Gestion des compteurs (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°12

## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
17/20	17/20	+0

**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'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>
#define N 100
#define A 5
#define S 10

int main()
{
    int X, nbre_pr = 0, nbre_ab = 0, i = 0;
    while (i < N || nbre_ab < S)
    {
        printf("enter the number of attended sessions: ");
        scanf("%d", &X);
        if (X < A)
        {
            nbre_ab++;
        }
        else
        {
            nbre_pr++;
        }
        i++;
        printf("present students = %d", nbre_pr);
        printf("absent students = %d", nbre_ab);
        printf("step number %d", i);
    }

    printf("Total processed students = %d", i);
    printf("present students = %d", nbre_pr);
    printf("absent students = %d", nbre_ab);

    if (nbre_ab < 3)
    {
        printf("session valid");
    }
    else
    {
        printf("session cancelled");
    }
}
```



```
    return 0;  
}
```

## Analyse Détaillée

### Lecture des données (1/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 3/3

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: 4/4

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

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

Ancienne note: 4/4

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: 3/3

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

## Affichages (1/1 pt)

Ancienne note: 3/3

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	17/20	+12

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ 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, sum = 0;
    int S, total, A, x;
    printf("enter total number");
    scanf("%d", &n);
    printf("enter minimum attendance");
    scanf("%d", &A);
    printf("enter absence threshold");
    scanf("%d", &S);

    while (n > S || A == 5)
    {
        printf("enter x");
        scanf("%d", &x);
        if (x < A)
        {
            printf("absent");
        }
        else
        {
            printf("present");
        }
        printf("total sum = %d", sum);
    }
    return 0;
}
```



## Analyse Détaillée



### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/3

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

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

## Suggestions d'Amélioration

---

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

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°14

## 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

---

- ✓ Lecture de N avec scanf
  - ✓ Lecture de A avec scanf
  - ✓ Lecture de S avec scanf
  - ✓ 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, s, x, absent, present;
    int n;
    printf("enter the number of registered students: ");
    scanf("%d", &n);
    printf("enter the minimum attendance required: ");
    scanf("%d", &a);
    printf("enter the absence threshold: ");
    scanf("%d", &s);
    printf("enter the number of attended session: ");
    scanf("%d", &x);

    if (x < a)
    {
        printf("the student is absent");
        absent = n + 1;
    }
    else
    {
        printf("the student is present");
        present = n + 1;
    }

    printf("the number of present student is: %d", present);
    printf("the number of absent student is: %d", present);
    return 0;
}
```



## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°15

## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
19/20	19/20	+0

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X;
    int i = 1, present = 0, absent = 0;
    printf("Please input the total number of registered students: ");
    scanf("%d", &N);
    printf("Please input the minimum attendance required: ");
    scanf("%d", &A);
    printf("Please input the absence threshold: ");
    scanf("%d", &S);

    printf("Please input the number of attended sessions for each student:");
    for (i = 1; i <= N; i++)
    {
        if (absent < S)
        {
            printf("Student number: %02d", i);
            printf("X: ");
            scanf("%d", &X);
            printf("Student status: ");
            if (X < A)
            {
                absent++;
                printf("Absent");
            }
            else
            {
                present++;
                printf("Present");
            }
            printf("Present students: %d", present);
            printf("Absent students: %d", absent);
        }
    }

    printf("Total processed students: %d", present + absent);
```

```
printf("Present students: %d", present);  
printf("Absent students: %d", absent);  
  
if (absent < S)  
{  
    printf("Session Valid");  
}  
else  
{  
    printf("Session cancelled");  
}  
return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 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: 3/4

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

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

Ancienne note: 4/4

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: 3/3

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



## Affichages (1/1 pt)

Ancienne note: 3/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

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



## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°16



## Comparaison des Notes

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

**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'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X, i = 1, present = 0, absent = 0;
    printf("entre N");
    scanf("%d", &N);
    printf("entre A");
    scanf("%d", &A);
    printf("entre S");
    scanf("%d", &S);

    while (i <= N && absent < S)
    {
        printf("Student %d - attended session", i);
        scanf("%d", &X);
        if (X < A)
        {
            absent++;
        }
        else
        {
            present++;
        }
        printf("step %d", i);
        printf("present = %d", present);
        printf("absent = %d", absent);
        i++;
    }

    printf("final result");
    printf("processed student %d", i - 1);
    printf("absent student %d", absent);
    printf("present student %d", present);

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



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

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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°17



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	15/20	+1

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



## Points Forts

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



## Code Soumis

---

```
#include <stdio.h>

int main()
{
    // read A, N, S
    int i = 1;
    int present = 0;
    int absent = 0;
    while (i <= N && absent < S)
    {
        // read x
        if (x < A)
        {
            absent++;
        }
        else
        {
            present++;
        }
        // print i, present, absent
        i++;
    }

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

## Analyse Détaillée

---

### Lecture des données (0/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 3/3

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: 4/4

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

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

Ancienne note: 3/4

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: 3/3

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

### Affichages (0/1 pt)

Ancienne note: 1/3

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*

---

# Correction Encourageante - Copie N°18

---

## Comparaison des Notes

---

Ancienne Note	Nouvelle Note	Différence
7/20	18/20	+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 S avec scanf
- ✓ 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'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, X, i;
    int S;
    for (i = 1; i <= N; i++)
    {
        printf("read the number of attended sessions x");
        if (x < A)
        {
            printf("student absent");
            scanf("%d", &student_absent);
        }
        else if (x > A)
        {
            printf("student present");
            scanf("%d", &student_present);
        }

        number_of_present_student = N - student_absent;
        number_of_absent_student = N - student_present;

        scanf("%d %d", &number_of_present_student, &number_of_absent_stude

        if (A == N)
        {
            printf("stop");
        }
        else if (N == S)
        {
            printf("stop");
        }

        if (the student is present)
        {
            printf("session valid");
        }
    }
}
```



```
        else if (the student is absent)
        {
            printf("session cancelled");
        }
    }
    return 0;
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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



## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

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



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !



## Message d'Encouragement

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



## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°19



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
17/20	19/20	+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
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ Comparaison  $X < A$  pour déterminer l'absence
- ✓ Structure if/else pour présence/absence
- ✓ Utilisation d'incrémentations
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, S, A;
    printf("enter num of students");
    scanf("%d", &N);
    printf("min attendance red");
    scanf("%d", &A);
    printf("absence threshold");
    scanf("%d", &S);

    int i, x, P = 0, D = 0;
    for (int i = 1; i <= N; i++)
    {
        printf("enter num of attendance for student %d", i);
        scanf("%d", &x);
        printf("Num of students are %d", i);
        if (x < A)
        {
            P++;
            printf("students are absents");
        }
        else
        {
            D++;
            printf("%d students are present", D);
        }
    }

    int T = 0;
    T = D + P;
    printf("total students is %d", T);
    printf("present students are %d", D);
    printf("absent students are %d", P);

    if (T >= S)
    {
```

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

## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 3/3

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: 2/4

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

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

Ancienne note: 4/4

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: 3/3

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



## Affichages (1/1 pt)

Ancienne note: 2/3

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	9/20	+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

---

- ✓ Lecture de N avec scanf
  - ✓ Lecture de A avec scanf
  - ✓ Messages utilisateur avant saisie
  - ✓ Déclaration des variables
  - ✓ Utilisation de structure conditionnelle
-



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S;
    int total_number_of_registered;
    printf("enter total number of registered students");
    scanf("%d", &N);

    if (number_of_attended_sessions < A)
    {
        write("the student is absent");
        write("number of attended session");
    }
    else if (number_of_attended_sessions > A)
    {
        write("the student is present");
    }

    printf("enter the number of attended sessions x student by student");
    scanf("%d %d", &present_students, &absent_students);
    write("simulations stop");

    if (x == N)
    {
        // total processed student
        if (absent < S)
        {
            write("session valid");
        }
    }

    if (absent > S)
    {
        write("session cancelled");
    }
}
```



```
return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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 (0/2 pts)

Ancienne note: 0/3

Utilisez des compteurs (ex: `resents++` 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. 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°21



## Comparaison des Notes

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

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ 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'incrémentations
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X, aps, ress;
    printf("entre a total number of registered student");
    scanf("%d", &N);
    printf("entre minimum attendance required");
    scanf("%d", &A);
    printf("entre absence threshold");
    scanf("%d", &S);

    printf("entre number of attended session");
    scanf("%d", &X);
    printf("entre aps");
    scanf("%d", &aps);
    printf("entre ress");
    scanf("%d", &ress);

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

    while (A == N && A == S)
    {
        if (X < A)
        {
            S = N - 1;
        }
        else
        {
            X++;
        }
    }
}
```

```
    }  
  
    if (aps < S && ress >= A)  
    {  
        printf("session valide");  
    }  
    return 0;  
}
```

## Analyse Détaillée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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



## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

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



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.



## Suggestions d'Amélioration

1. Revoir la condition d'arrêt de la boucle



## Message d'Encouragement

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



## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

# Correction Encourageante - Copie N°22



## Comparaison des Notes

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

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ 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;
    printf("Enter Total number of registered students: ");
    scanf("%d", &N);
    printf("Enter number of minimum attendance required: ");
    scanf("%d", &A);
    printf("Enter number of absence threshold: ");
    scanf("%d", &S);

    while (i <= N || B == S)
    {
        int B = 0;
        int P = 0;
        printf("Enter the number of absence of student n %d", i);
        scanf("%d", &X);

        if (i == N || B == S)
        {
            printf("Total processed students is %d", i);
            printf("Present students is %d", P);
            printf("Absent students is %d", B);

            if (B == S)
            {
                printf("Session status is cancelled");
            }
            else
            {
                printf("Session status is valid");
            }
        }

        if (X < A)
        {
            B++;
        }
    }
}
```



```
    }  
    else  
    {  
        P++;  
    }  
}  
return 0;  
}
```

## Analyse Détaillée

### Lecture des données (4/4 pts)

Ancienne note: 3/3

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

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

Ancienne note: 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: 4/4

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*