

# Corrections Encourageantes -

## doc20260115220910

---

**Nombre de copies:** 18

**Moyenne ancienne correction:** 10.61/20

**Moyenne nouvelle correction:** 17.22/20

**Différence moyenne:** +6.61

**Copies améliorées:** 17/18

---

## Correction Encourageante - Copie N°1

---



### Comparaison des Notes

---

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

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



### Points Forts

---

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

- ✓ Déclaration des variables
  - ✓ Initialisation de variables à 0
  - ✓ Utilisation d'une boucle while
  - ✓ Comparaison  $X < A$  pour déterminer l'absence
  - ✓ Structure if/else pour présence/absence
  - ✓ Affichages multiples
-



## Code Soumis

```
#include <stdio.h>

int main()
{
    int A, N, S, a;
    int x, i = 0, p = 0;
    printf("Enter the integer positive N: ");
    scanf("%d", &N);
    printf("Enter the minimum attendance required A: ");
    scanf("%d", &A);
    printf("Enter absence threshold S: ");
    scanf("%d", &S);

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

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

    while (x >= A)
    {
        p = p + 1;
        printf("%d", p);
        a = N - p;
    }
    printf("%d", A);

    if (p > A)
    {
        printf("Exam valid");
    }
    else
```

```
{  
    printf("Exam cancelled");  
}  
return 0;  
}
```

## Analyse Détailée

### 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 (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: 2/4

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: 1/3

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

## Affichages (1/1 pt)

Ancienne note: 2/3

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°2



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
8/20	14/20	+6

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X;
    printf("Enter Number of attended sessions: ");
    int i = 0, j = 0;
    while (i <= N || j <= S)
    {
        j++;
        i++;
        if (scanf("%d", &X) <= A)
        {
            printf("Number of Student Absent is %d", j);
        }
        else
        {
            printf("Number of Student Present is %d", i);
        }
    }

    b = N - i;
    c = N - j;

    printf("Number of All Student Absent is: %d", c);
    printf("Number of All Student Present is: %d", b);

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

## Analyse Détailée

### Lecture des données (1/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 2/3

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 (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 (2/2 pts)

Ancienne note: 2/3

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

### Affichages (1/1 pt)

Ancienne note: 3/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

1. Comparer X avec A pour déterminer l'absence

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

Correction réalisée avec une approche encourageante pour étudiants débutants

## Correction Encourageante - Copie N°3

### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
5/20	15/20	+10

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



## Points Forts

- ✓ Messages utilisateur avant saisie
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle while
- ✓ 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, Np, Na;
    int i = 0;
    while (i <= N, i < S, i++)
    {
        i = i + 1;
        printf("the student number is: ", student_number);
        scanf("%d", &x);

        if (x < A)
        {
            printf("the student is absent");
            Na = N - A;
            printf("number of absent student is: ", Na);
        }
        else
        {
            printf("the student is present");
            Np = N - S;
            printf("number of present student is: ", Np);
        }
    }

    if (present_students > absent_students)
    {
        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: 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: 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: 2/3

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°4

### Comparaison des Notes

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

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ 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>
#include <stdlib.h>

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

    for (int i = 1; i <= N; i++)
    {
        id = i + 1;
        printf("%d", id);
        scanf("%d", &x);
        if (x < A)
        {
            printf("student is absent");
        }
        else
        {
            printf("student is present");
        }
    }

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

```
    return 0;  
}
```



## Analyse Détailée



### Lecture des données (4/4 pts)

Ancienne note: 3/3

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



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

Ancienne note: 1/3

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



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

Ancienne note: 2/4

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: 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: 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°5



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	19/20	+5

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int A;
    printf("entre A\n");
    scanf("%d", &A);
    int N;
    printf("entre N\n");
    scanf("%d", &N);
    int S;
    printf("entre S\n");
    scanf("%d", &S);

    int x, Absent, Present;
    int i, j;

    for (i = 0; i <= N; i++)
    {
        scanf("%d", &x);
        printf("x: %d", i);
        if (x < A)
        {
            printf("x is absent", Absent);
            Absent++;
        }
        else
        {
            printf("x is present", Present);
            Present++;
        }
    }

    printf("the Absent is: %d", Absent);
    printf("the Present is: %d", Present);

    if (Absent < S)
    {
```

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



## Analyse Détailée



### 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: 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.

12  
34

## 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°6



## Comparaison des Notes

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

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

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

    for (i = 1; i <= N; i++)
    {
        printf("entre x : ");
        scanf("%d", &x);
        if (x < A)
        {
            printf("student absent");
        }
        else
        {
            printf("student present");
        }

        i = i + 1;
        if (absent_count >= S)
        {
            printf("Exam valid");
        }
        else
        {
            printf("Exam cancelled");
        }
    }
    return 0;
}
```

## Analyse Détailée

---

### 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: 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 (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°7

### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
4/20	17/20	+13

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X, absents, presents;
    printf("total students: ");
    scanf("%d", &N);
    printf("minimum attendance required: ");
    scanf("%d", &A);
    printf("absence threshold: ");
    scanf("%d", &S);

    while (i <= N)
    {
        scanf("%d", &x);
        printf("student = %d", i);
        if (x < A)
        {
            scanf("%d", &absents);
        }
        else
        {
            scanf("%d", &presents);
        }
    }

    printf("absents = %d", absents);
    printf("presents = %d", presents);

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

```
    return 0;  
}
```

## Analyse Détailée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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. 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°8



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
19/20	14/20	-5

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



## Points Forts

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



```
#include <stdio.h>

int main()
{
    int a, s, n, tot = 0, pr = 0, ab = 0;
    printf("Enter N, A and S\n");
    scanf("%d %d %d", &n, &a, &s);

    int x = 1;
    for (int x = 1; x <= n; x++)
    {
        if (x > a)
        {
            printf("present\n");
            pr = pr + 1;
        }
        else
        {
            printf("absent\n");
            ab = ab + 1;
        }

        if (ab >= s)
        {
            printf("cancelled\n");
            break;
        }
        tot = tot + 1;
    }

    printf("present students = %d\n", pr);
    printf("absent students = %d\n", ab);
    printf("total processed students = %d", tot);

    if (x > a)
    {
        printf("valid session\n");
    }
}
```

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

## Analyse Détailée

### 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 (3/5 pts)

Ancienne note: 3/4

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: 4/4

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: 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

---

1. Revoir la condition d'arrêt de la boucle
  2. Comparer X avec A pour déterminer l'absence
- 



## Message d'Encouragement

---

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

---



## Détails de l'Ancienne Correction

---

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

---

Correction réalisée avec une approche encourageante pour étudiants débutants

---

# Correction Encourageante - Copie N°9



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
16/20	18/20	+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
- ✓ 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'incrémantation
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, X;
    int i, na = 0, np;
    printf("Enter the total number of registered students: ");
    scanf("%d", &N);

    for (i = 1; i <= N; i++)
    {
        printf("student %d", i);
        scanf("%d", &X);
        if (X < A)
        {
            na = na + 1;
        }
        else
        {
            np = np + 1;
        }

        if (i == N || na == S)
        {
            printf("stop the simulation");
            break;
        }
    }

    printf("present students is %d", np);
    printf("absent students is %d", na);

    if (na <= np)
    {
        printf("valid");
    }
    else
    {
```

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



## Analyse Détailée



### Lecture des données (3/4 pts)

Ancienne note: 0/3

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



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

Ancienne note: 0/3

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



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

Ancienne note: 0/4

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



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

Ancienne note: 0/4

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



### Gestion des compteurs (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°10



## Comparaison des Notes

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

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

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

    for (i = 1; i <= N; i++)
    {
        printf("enter the number of attended session: ");
        scanf("%d", &X);
        if (X < A)
        {
            printf("student absent");
            B = B + 1;
            printf("absent students = %d", B);
        }
        else
        {
            printf("student present");
            P = P + 1;
            printf("present students = %d", P);
        }

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

```
}
```

## Analyse Détailée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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°11



## 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

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle while
- ✓ Condition de boucle sur N
- ✓ Utilisation de structure conditionnelle
- ✓ Utilisation d'incrémentation
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

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

    while (i <= N)
    {
        i++;
        printf("%d", N);
        if (i > A)
        {
            printf("total present", i);
        }
        else
        {
            printf("S of", i);
        }
    }
    return 0;
}
```



## Analyse Détailée



### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/3

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

 **Affichages (1/1 pt)**

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---

 **Suggestions d'Amélioration**

---

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

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

## Correction Encourageante - Copie N°12

### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	19/20	+5

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



## Points Forts

---

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



## Code Soumis

```
#include <stdio.h>

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

    while (student <= N || count == 5)
    {
        printf("Enter number of attended sessions: ");
        scanf("%d", &x);
        if (x < A)
        {
            printf("Considered absent");
            count++;
        }
        else
        {
            printf("Considered Present");
            student++;
        }
    }

    absent = count;
    present = N - count;

    printf("number of present students: %d", present);
    printf("number of absent student: %d", absent);

    if (present < absent)
    {
        printf("Exam cancelled");
    }
}
```

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

## Analyse Détailée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

### Structure de boucle (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

---

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°13



## Comparaison des Notes

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

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de 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'incrémantation
- ✓ Gestion de plusieurs compteurs
- ✓ Affichages multiples



## Code Soumis

```
#include <stdio.h>

int main()
{
    int A, N, S, n = 0, m = 0;
    printf("entre the number A, N, S: ");
    scanf("%d %d %d", &A, &N, &S);

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

        if (m < S)
        {
            printf("Exam valid");
        }
        else
        {
            printf("Exam cancelled");
        }
    }
    return 0;
}
```

## Analyse Détailée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/3

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

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

► [Cliquez pour voir l'analyse précédente \(correction stricte\)](#)

*Correction réalisée avec une approche encourageante pour étudiants débutants*

## Correction Encourageante - Copie N°14

### Comparaison des Notes

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

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



## Points Forts

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, sc, m = 0, n = 0, S;
    printf("entre the total number of registered student: ");
    scanf("%d", &N);
    printf("entre the minimum attendance required: ");
    scanf("%d", &A);

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

    printf("entre the absence threshold: ");
    scanf("%d", &S);

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

## Analyse Détailée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/3

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

### Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

## Suggestions d'Amélioration

Continuez à pratiquer et à consolider vos acquis !

## Message d'Encouragement

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

## Détails de l'Ancienne Correction

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

*Correction réalisée avec une approche encourageante pour étudiants débutants*

## Correction Encourageante - Copie N°15

### Comparaison des Notes

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

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, i, X;
    int Sum1 = 0;
    int Sum2 = 0;
    printf("enter N: ");
    scanf("%d", &N);
    i = 1;
    while (i <= N || Sum1 == S)
    {
        printf("enter X: ");
        scanf("%d", &X);
        if (X < A)
        {
            printf("the student num %d: absent", i);
            Sum1 = Sum1 + 1;
        }
        else
        {
            printf("the student num %d: present", i);
            Sum2 = Sum2 + 1;
        }
        i++;
    }

    printf("%d %d", Sum1, Sum2);

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

```
    return 0;  
}
```

## Analyse Détailée

### Lecture des données (3/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/3

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

 **Affichages (1/1 pt)**

*Ancienne note: 0/3*

Bien ! Vous avez pensé à afficher des résultats.

---

 **Suggestions d'Amélioration**

---

Continuez à pratiquer et à consolider vos acquis !

---

 **Message d'Encouragement**

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

---

 **Détails de l'Ancienne Correction**

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

---

*Correction réalisée avec une approche encourageante pour étudiants débutants*

---

# Correction Encourageante - Copie N°16



## Comparaison des Notes

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

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Utilisation d'une boucle for
- ✓ 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 %d %d", &N, &A, &S);

    for (A = 1; A <= N; A++)
    {
        scanf("%d", &x);
        if (x < A)
        {
            printf("the student is absent");
        }
        else
        {
            printf("the student is present");
        }
        count++;
    }

    // simulation stops
    if (S <= A)
    {
        printf("Session cancelled");
    }
    else
    {
        printf("session valid");
    }
    return 0;
}
```

## Analyse Détailée

---

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/4

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

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

Ancienne note: 0/3

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

### Affichages (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°17

### Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
14/20	19/20	+5

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



## Points Forts

- ✓ Lecture de N avec scanf
- ✓ Lecture de A avec scanf
- ✓ Lecture de S avec scanf
- ✓ Déclaration des variables
- ✓ Initialisation de variables à 0
- ✓ Utilisation d'une boucle for
- ✓ Condition de boucle sur N
- ✓ 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, n = 0;
    int OS = 0;
    int PS = 0;
    printf("enter number de students\n");
    scanf("%d", &N);
    printf("enter minimum attendance required\n");
    scanf("%d", &A);
    printf("enter absence threshold\n");
    scanf("%d", &S);

    for (i = 1; i <= N; i++)
    {
        if (i <= N && OS < S)
        {
            printf("enter number of attended sessions\n");
            scanf("%d", &x);
            if (x < A)
            {
                OS = OS + 1;
            }
            else
            {
                PS = PS + 1;
            }
            printf("%d", OS);
            printf("%d", PS);
            printf("%d", i);
        }
        n = n + i;
    }

    if (n < S)
    {
        printf("Exam valid");
    }
}
```

```
    }
else
{
    printf("Exam 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 (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.

12  
34

## Gestion des compteurs (1/2 pts)

Ancienne note: 0/3

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



## Affichages (1/1 pt)

Ancienne note: 0/3

Bien ! Vous avez pensé à afficher des résultats.

---



## Suggestions d'Amélioration

---

Continuez à pratiquer et à consolider vos acquis !

---



## Message d'Encouragement

---

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

---



## Détails de l'Ancienne Correction

---

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

---

Correction réalisée avec une approche encourageante pour étudiants débutants

---

# Correction Encourageante - Copie N°18



## Comparaison des Notes

Ancienne Note	Nouvelle Note	Différence
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

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



## Code Soumis

```
#include <stdio.h>

int main()
{
    int N, A, S, OS, PS;
    printf("Enter N and A and S: ");
    scanf("%d %d %d", &N, &S, &A);

    for (int i = 1; i <= N; i++)
    {
        int n;
        scanf("%d", &n);
        if (n < A)
        {
            printf("The student absent");
        }
        else
        {
            printf("The student present");
        }

        if (n < A)
        {
            PS = PS + 1;
            printf("Number student present");
        }
        else
        {
            PS = PS + 1;
            printf("Number student absent");
        }

        if (N student are processed || absent student reaches)
        {
            printf("simulation stops");
        }

        if (0 S = s)
```

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

## Analyse Détailée

### Lecture des données (4/4 pts)

Ancienne note: 0/3

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

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

Ancienne note: 0/3

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

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

Ancienne note: 0/4

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

### Logique présence/absence (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).

12  
34

## 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