

```

#include <stdio.h>
int main() {
    int N, A, S;
    printf("N = ");
    scanf("%d", &N);
    printf("A = ");
    scanf("%d", &A);
    printf("S = ");
    scanf("%d", &S);

    int i, X absent, p = 0, a = 0; // p: present students, a = absent students
    for (i i = 1; i <= N || a > S; i++) {
        printf("%d", i);
        scanf("%d", &X absent);
        if (X absent < A) {
            printf("%d absent", i);
            a++;
        } else {
            printf("%d present", i);
            p++;
        }
        printf("a = %d", a);
        printf("p = %d", p);
    }
    printf if (a < S) {
        printf("Session Valid");
    } else {
        printf("Session Cancelled");
        return 0;
    }
}

```

```
#include <stdio.h>
int main()
{
    int N, A, S;
    printf("N = ");
    scanf("%d", &N);
    printf("A = ");
    scanf("%d", &A);
    printf("S = ");
    scanf("%d", &S);
    int i, X, P = 0, a = 0;
    // p: present students, a: absent for (i = 1; i <= N || a > S; i++)
    {
        printf("%d", i);
        scanf("%d", &X);
        if (X < A)
        {
            printf("%d absent", i);
            a++;
        }

        else
        {
            printf("%d present", i);
            P++;
        }

        printf("a = %d", a);
        printf("p = %d", P);
    }

    if (a < S)
    {
        printf("Session Valid");
    }

    else
    {
        printf("session cancelled");
    }

    return 0;
}
```

Analyse :**Algorithmique :**

- Initialisation $P=0$, $a=0$.
- Boucle `for (i = 1; i <= N || a > S; i++)` : Condition `||` (OU) signifie que la boucle continue SI on dépasse le seuil !! C'est l'inverse (`&& a < S`).
- Lectures et tests corrects.
- Incrémentation correcte.
- Affichage final correct.

Notation :

Critère	Points	Commentaire
Lecture N, A, S	3 / 3	Correct.
Initialisation	3 / 3	Correct.
Condition boucle	2 / 4	Erreur de logique booléenne (<code> </code> prolonge la boucle au lieu d'arrêter).
Logique prés./abs.	4 / 4	Correct.
Compteurs	3 / 3	Correct.
Affichages inter.	2 / 2	Correct.
Affichage final	1 / 1	Correct.

NOTE FINALE : 18 / 20**Feedback :**

- **Points forts** : Très bon code.
 - **Points faibles** : Petite erreur de logique sur la condition d'arrêt (`||` vs `&&`).
 - **Appréciation globale** : Très Bon.
-