

Input reading: 3 pts | Initialization: 3 pts | Loop condition: 4 pts | Counters logic: 4 pts | Stop conditions: 3 | Final output: 3

```

#include <stdio.h>
int main() {
    int N, A, S, NA = 0, NP = 0, X;
    printf("enter the number of N and A and S");
    scanf("%d %d %d", &N, &A, &S);
    for (i = 1; i <= N; i++) {
        printf("enter the number of attended Session %d", i);
        scanf("%d", &X);
        if ("X < A") {
            NA = NA + 1;
        }
        else {
            NP = NP + 1;
        }
        printf("The number of present Students: %d", NP);
        printf("The number of Absent Students: %d", NA);
        if (NA > S) {
            printf("Session cancelled");
        }
    }
}

```

```
printf ("the number of present Students %.d", NP);  
printf ("the number of Absent Students %.d", NA);  
  
if (NP < 5) {  
    printf ("Session valid");  
}  
else {  
    printf ("Session cancelled");  
}  
  
set  
return 0;  
}
```

Copy 8

```
#include <stdio.h>
int main()
{
    int N, A, S, NA = 0, NP = 0, x, i;
    printf("entre the number de N and A and S");
    scanf("%d %d %d", &N, &A, &S);
    for (i = 1; i <= N; i++)
    {
        printf("entre the number of attended Sessin x : %d", i);
        scanf("%d", &x);
        if (x < A)
        {
            NA = NA + 1;
        }

        else
        {
            NP = NP + 1;
        }

        printf("the number of present students : %d", NP);
        printf("the number of Absent students : %d", NA);
        if (NA >= S)
        {
            printf("Session cancellad");
        }

    }

    printf("the number of present students %d", NP);
    printf("the number of Absent students %d", NA);
    if (NP < S)
    {
        printf("Session valid");
    }

    else
    {
        printf("Session cancellad");
    }

    return 0;
}
```

Analyse :

Algorithmique :

- Lecture OK.
- Boucle `for`.
- Logique interne correcte (Compteurs `NA`, `NP`).
- Condition finale `if (NP < S)`. Erreur: c'est le seuil d'ABSENCE (`NA`) qui compte, pas la présence. Donc `valid` si `NA < S` (et non `NP`). Là il dit `valid` si `NP < S` (peu de présents -> valide ? Contresens).

NOTE FINALE : 14 / 20

Feedback :

- **Appréciation globale : Moyen.** Confusion sur la condition de validité finale.
-