mayTakeLeave

Given the following structure definition, write the code for the functions getInput(), mayTakeLeave() and printList() with the following function prototypes:

```
typedef struct {
  int id;              /* staff identifier */
  int totalLeave;   /* the total number of days of leave allowed */
  int leaveTaken;   /* the number of days of leave taken so far */
} leaveRecord;
```

(a) void getInput(leaveRecord list[ ], int *n);

Each line of the input has three integers representing one staff identifier, his/her total number of days of leave allowed and his/her number of days of leave taken so far respectively. The function will read the data into the array *list* until end of input and returns the number of records read through *n*.

(b) int mayTakeLeave(leaveRecord list[ ], int id, int leave, int n);

It returns 1 if a leave application for *leave* days is approved. Staff member with identifier *id* is applying for *leave* days of leave. *n* is the number of staff in *list*. Approval will be given if the leave taken so far plus the number of days applied for is less than or equal to his total number of *leave* days allowed. If approval is not given, it returns 0. It will return -1 if no one in *list* has identifier *id*.

(c) void printList(leaveRecord list[ ], int n);

It prints the *list* of leave records of each staff. *n* is the number of staff in *list*.

A sample program template is given below to test the functions:

```
#include <stdio.h>
#define INIT_VALUE 1000
typedef struct {
  int id;              /* staff identifier */
  int totalLeave;   /* the total number of days of leave allowed */
  int leaveTaken;   /* the number of days of leave taken so far */
} leaveRecord;
int mayTakeLeave(leaveRecord list[], int id, int leave, int n);
void getInput(leaveRecord list[], int *n);
void printList(leaveRecord list[], int n);
int main()
{
  leaveRecord listRec[10];
  int len;
  int id, leave, canTake=INIT_VALUE;
  int choice;
```

```
  printf("Select one of the following options: \n");
  printf("1: getInput()\n");
  printf("2: printList()\n");
  printf("3: mayTakeLeave()\n");
  printf("4: exit()\n");
  do {
    printf("Enter your choice: \n");
    scanf("%d", &choice);
    switch (choice) {
      case 1:
        getInput(listRec, &len);
        printList(listRec, len);
        break;
      case 2:
        printList(listRec, len);
        break;
      case 3:
        printf("Please input id, leave to be taken: \n");
        scanf("%d %d", &id, &leave);
        canTake = mayTakeLeave(listRec, id, leave, len);
        if (canTake == 1)
          printf("The staff %d can take leave\n", id);
        else if (canTake == 0)
          printf("The staff %d cannot take leave\n", id);
        else if (canTake == -1)
          printf("The staff %d is not in the list\n", id);
        else
          printf("Error!");
        break;
    }
  } while (choice < 4);
  return 0;
}
void printList(leaveRecord list[], int n)
{
  int p;

  printf("The staff list:\n");
  for (p = 0; p < n; p++)
    printf ("id = %d, totalleave = %d, leave taken = %d\n",
      list[p].id, list[p].totalLeave, list[p].leaveTaken);
}
void getInput(leaveRecord list[], int *n)
{
     /* Write your program code here */
}
int mayTakeLeave(leaveRecord list[], int id, int leave, int n)
```

```
{
    /* Write your program code here */
}
```

Some sample input and output sessions are given below:

(1) Test Case 1:

```
Select one of the following options:
1: getInput()
2: printList()
3: mayTakeLeave()
4: exit()
Enter your choice:
1
Enter the number of staff records:
2
Enter id, totalleave, leavetaken:
11 28 25
Enter id, totalleave, leavetaken:
12 28 6
The staff list:
id = 11, totalleave = 28, leave taken = 25
id = 12, totalleave = 28, leave taken = 6
Enter your choice:
3
Please input id, leave to be taken:
11 6
The staff 11 cannot take leave
Enter your choice:
4
```

(2) Test Case 2:

```
Select one of the following options:
1: getInput()
2: printList()
3: mayTakeLeave()
4: exit()
Enter your choice:
1
Enter the number of staff records:
2
Enter id, totalleave, leavetaken:
11 28 25
Enter id, totalleave, leavetaken:
12 28 6
The staff list:
id = 11, totalleave = 28, leave taken = 25
id = 12, totalleave = 28, leave taken = 6
```

Enter your choice:
3
Please input id, leave to be taken:
12 6
The staff 12 can take leave
Enter your choice:
4

(3) Test Case 3:

Select one of the following options:
1: getInput()
2: printList()
3: mayTakeLeave()
4: exit()
Enter your choice:
1
Enter the number of staff records:
2
Enter id, totalleave, leavetaken:
11 28 25
Enter id, totalleave, leavetaken:
12 28 6
The staff list:
id = 11, totalleave = 28, leave taken = 25
id = 12, totalleave = 28, leave taken = 6
Enter your choice:
3
Please input id, leave to be taken:
13 6
The staff 13 is not in the list
Enter your choice:
4

(4) Test Case 4:

Select one of the following options:
1: getInput()
2: printList()
3: mayTakeLeave()
4: exit()
Enter your choice:
1
Enter the number of staff records:
2
Enter id, totalleave, leavetaken:
11 28 25
Enter id, totalleave, leavetaken:
12 28 6
The staff list:

id = 11, totalleave = 28, leave taken = 25
id = 12, totalleave = 28, leave taken = 6
Enter your choice:
4