Status	Finished
Started	Monday, 3 November 2025, 6:41 PM
Completed	Monday, 3 November 2025, 8:01 PM
Duration	1 hour 19 mins

Question **1**

Correct

The name and mileage of certain cars is passed as the input. The format is CARNAME@MILEAGE and the input is as a single line, with each car information separated by a space. The program must print the car with the lowest mileage. (Assume no two cars will have the lowest mileage)

Input Format:

The first line contains the CARNAME@MILEAGE separated by a space.

Output Format:

The first line contains the name of the car with the lowest mileage.

Boundary Conditions:

The length of the input string is between 4 to 10000.
The length of the car name is from 1 to 50.

Example Input/Output 1:

Input:

Zantro@16.15 Zity@12.5 Gamry@9.8

Output:

Gamry

For example:

Input	Result
Zantro@16.15 Zity@12.5 Gamry@9.8	Gamry

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main()
{
char input [10001];
```

```
fgets(input,sizeof(input),stdin);
char*token=strtok(input," ");
char carName[51],lowestcar[51];
float mileage,lowestMileage=100000.0;
while(token!=NULL){
char*atPos=strchr(token,'@');
if(atPos !=NULL)
{
*atPos='\0';
strcpy(carName,token);
mileage=atof(atPos+1);
if(mileage<lowestMileage) {
lowestMileage=mileage;
strcpy(lowestcar,carName);
}
}
token=strtok(NULL," ");
}printf("%s\n",lowestcar);
return 0;}

```

	Input	Expected	Got	
✓	Zantro@16.15 Zity@12.5 Gamry@9.8	Gamry	Gamry	✓

Passed all tests! ✓

Question **2**

Correct

A certain number of people attended a meeting which was to begin at 10:00 am on a given day. The arrival time in HH:MM format of those who attended the meeting is passed as the input in a single line, with each arrival time by a space. The program must print the count of people who came late (after 10:00 am) to the meeting.

Input Format:

The first line contains the arrival time separated by a space.

Output Format:

The first line contains the count of late comers.

Boundary Conditions:

The length of the input string is between 4 to 10000.
The time HH:MM will be in 24 hour format (HH is hours and MM is minutes).

Example Input/Output 1:

Input:

10:00 9:55 10:02 9:45 11:00

Output:

2

Explanation:

The 2 people were those who came at 10:02 and 11:00

For example:

Input	Result
`10:00 9:55 10:02 9:45 11:00`	`2`

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include<string.h>
int main() {
    char input[10000];
    fgets(input,sizeof(input),stdin);
    int latecount= 0;
    char*token=strtok(input," ");
    while(token!=NULL) {
        int hour, minute;
        if(sscanf(token,"%d:%d", &hour,&minute)==2){
            if(hour>10||(hour==10 && minute>0))
            latecount++;
        }

        token=strtok(NULL," ");
    }




        printf("%d\n",latecount);
 return 0;
}

```

	Input	Expected	Got	
✓	10:00 9:55 10:02 9:45 11:00	2	2	✓

Passed all tests! ✓

Question **3**

Correct

A single line consisting of a set of integers, each separated by space is passed as input to the program. The program must print the sum of all the integers present.

Input Format:

The first line contains the integer values (Each separated by a space)

Output Format:

The first line contains the sum of all the integers.

Boundary Conditions:

The length of the input string is between 3 to 10000
The value of the integer values will be from -99999 to 99999

Example Input/Output 1:

Input:

100 -99 98 5

Output:

104

Example Input/Output 2:

Input:

100 200 -300 500 -450 -50

Output:

0

For example:

Input	Result
100 -99 98 5	104
100 200 -300 500 -450 -50	0

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<stdlib.h>
#include<ctype.h>
int main()
{
    char str[10001];
    fgets(str,sizeof(str),stdin);
    int sum=0;
    int num=0;
    int sign=1;
    int inNumber=0;
    for(int i=0;str[i]!='\0';i++){
        if(str[i]=='-'){
            sign=-1;

        }else if(isdigit(str[i])){
            num=num*10+(str[i]-'0');
            inNumber=1;
        }else{
            if(inNumber){
                sum+=num*sign;
                num=0;
                sign=1;
                inNumber=0;
            }
        }
    }
    if(inNumber)sum+=num*sign;
    printf("%d\n",sum);
    return 0;
}

```

	Input	Expected	Got	
✓	100 -99 98 5	104	104	✓
✓	100 200 -300 500 -450 -50	0	0	✓

Passed all tests!