Status	Finished
Started	Monday, 3 November 2025, 12:57 PM
Completed	Monday, 3 November 2025, 2:02 PM
Duration	1 hour 5 mins

Question **1**

Correct

A single line L with a set of space separated values indicating distance travelled and time taken is passed as the input. The program must calculate the average speed S (with precision upto 2 decimal places) and print S as the output.

Note: The distance and time taken will follow the format DISTANCE@TIMETAKEN. DISTANCE will be in kilometers and TIMETAKEN will be in hours.

Input Format:

The first line contains L.

Output Format:

The first line contains the average speed S.

Boundary Conditions:

Length of L will be from 3 to 100.

Example Input/Output 1:

Input:

60@2 120@3

Output:

36.00 kmph

Explanation:

Total distance = 60+120 = 180 km.
Total time taken = 2+3 = 5 hours.
Hence average speed = 180/5 = 36.00 kmph

For example:

Input	Result
60@2 120@3	36.00 kmph

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main ()
{
    char L[101];
    fgets(L,sizeof(L),stdin);
    L[strcspn(L,"\n")]=0;
    char *token;
    double total_distance=0.0;
    double total_time=0.0;
    token=strtok(L," ");
    while (token !=NULL) {
        char *at_pos=strchr(token,'@');
        if (at_pos !=NULL){
            *at_pos = '\0';
```

```
            char *distance_str=token;
            char *time_str=at_pos+1;
            double distance =atof(distance_str);
            total_distance+=distance;
            total_time+=atof(time_str);
        }
        token=strtok(NULL," ");
    } double average_speed=0.0;

    if (total_time>0){
        average_speed=total_distance/total_time;
    }
    printf ("%2.2lf kmph\n",average_speed);
    return 0;
}
```

	Input	Expected	Got	
✓	60@2 120@3	36.00 kmph	36.00 kmph	✓

Passed all tests! ✓

Question **2**

Correct

The program must accept two numbers X and Y and then print their HCF/GCD.

Input Format:

The first line denotes the value of X.
The second line denotes the value of Y.

Output Format:

The first line contains the HCF of X and Y.

Boundary Conditions:

1 <= X <= 999999
1 <= Y <= 999999

Example Input/Output 1:

Input:

30
40

Output:

10

Example Input/Output 2:

Input:

15
10

Output:

5

For example:

Input	Result
30 40	10

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int gcd(int a,int b){
    if (b==0)
    return a;
    return gcd(b, a%b);
    }
    int main (){
        int x,y;
        scanf ("%d",&x);
        scanf ("%d",&y);
        int result = gcd(x,y);
        printf ("%d\n", result);
        return 0;
    }
```

	Input	Expected	Got	
✓	30 40	10	10	✓

Passed all tests! ✓

Question **3**

Correct

A string S is passed as input. S will contain two integer values separated by one of these alphabets - A, S, M, D where

- A or a is for addition
- S or s is for subtraction
- M or m is for multiplication
- D or d is for division

The program must perform the necessary operation and print the result as the output. (Ignore any floating point values just print the integer result.)

Input Format:

The first line contains S.

Output Format:

The first line contains the resulting integer value.

Boundary Conditions:

Length of S is from 3 to 100.

Example Input/Output 1:

Input:

5A11

Output:

16

Explanation:

As the alphabet is A, 5 and 11 are added giving 16.

Example Input/Output 2:

Input:

120D6

Output:

20

Example Input/Output 3:

Input:

1405d10

Output:

140

For example:

Input	Result
5A11	16
120D6	20
1405d10	140

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main (){
    char S[101];

    fgets(S,sizeof(S),stdin);
    S[strcspn(S,"\n")]=0;
    int num1,num2;
    char operator=' ';
    int len=strlen(S);
    int op_pos=-1;
    for (int i=0; i<len;i++){
        if (S[i]=='A'||S[i]=='s'||S[i]=='s'||
        S[i]=='M'||S[i]=='m'||S[i]=='D'||S[i]=='d'){
            operator=S[i];
            op_pos=i;
            break;
        }
    }
if (op_pos !=-1){

    char num1_str[50]={0};
    strncpy(num1_str,S,op_pos);
    num1=atoi(num1_str);
    char num2_str[50]={0};
    strcpy (num2_str,S+op_pos+1);
    num2=atol(num2_str);
}
int result;
switch (operator){
    case 'A':
    case 'a':
    result=num1+num2;
    break;
    case 'S':
    case 's':
    result=num1*num2;
    break;
    case 'M':
    case 'm':
    result =num1-num2;
    break;
    case 'D':
    case 'd':
    if (num2!=0){
        result=num1/num2;
    }else {
        result=0;
    }
    break;
}
```

	Input	Expected	Got	
✓	5A11	16	16	✓
✓	120D6	20	20	✓
✓	1405d10	140	140	✓

Passed all tests! ✓