Status	Finished
Started	Sunday, 2 November 2025, 12:07 AM
Completed	Sunday, 2 November 2025, 12:43 AM
Duration	35 mins 55 secs

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 %)

```c
#include<stdio.h>
#include<string.h>
int main(){
    char input[200];
    fgets(input, sizeof(input),stdin);
    double total_distance=0,total_time=0;
    char*token=strtok(input," ");
    while(token!=NULL){
        double distance,time;
        sscanf(token,"%lf@%lf",&distance,&time);
        total_distance+=distance;
        total_time+=time;
        token=strtok(NULL," ");

    }
        double average_speed=total_distance/total_time;
        printf("%.2f 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 %)

```
1 #include<stdio.h>
2 int main()
```

```c
int main()
{
    int x,y;
    scanf("%d",&x);
    scanf("%d",&y);
    while(y!=0){
        int temp=y;
        y=x%y;
        x=temp;
    }
    printf("%d\n",x);
    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 %)

```
#include<stdio.h>
#include<string.h>
#include<ctype.h>
int main(){
    char s[100];
    scanf("%s",s);

    int num1 = 0, num2 = 0, i = 0;
    char op;
    while(s[i] != '\0'){
        if(isalpha(s[i])){
            op = s[i];
            break;
        }
        i++;
    }
    sscanf(s,"%d%*c%d", &num1, &num2);

    int result;

    switch(toupper(op)){
        case 'A':
        result = num1 + num2;
```

```
        break;
        case'S':
        result = num1 - num2;
        break;
        case'M':
        result = num1 * num2;
        break;
        case'D':
        result = num1 / num2;
        break;
        default:
        printf("Invalid operator");
        return 0;
    }

    printf("%d",result);
    return 0;
}
```

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

Passed all tests! ✓