

Status	Finished
Started	Monday, 3 November 2025, 9:20 PM
Completed	Monday, 3 November 2025, 10:00 PM
Duration	39 mins 34 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 %)

```

1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<string.h>
4 int main()
5 {
6     char i[101];
7     fgets(i,sizeof(i),stdin);
8     i[strcspn(i, "\n")]=0;
9     char*token;
10    double fd=0.0;
11    double ft=0.0;
12    token=strtok(i, " ");
13    while(token!=NULL){
14        char*atsign=strchr(token, '@');
15        if(atsign!=NULL){
16            *atsign='\n';
17            double d=atof(token);
18            double t=atof(atsign+1);
19            fd+=d;
20            ft+=t;
21        }
22        token=strtok(NULL, " ");
23    }
24    double s=0.0;
25    if(ft>0)
26        s=fd/ft;
27    printf("%.2lf kmph\n",s);
28    return 0;
29
30 }
31

```



	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:

$1 \leq X \leq 999999$

$1 \leq Y \leq 999999$

Example Input/Output 1:

Input:

30

40

Output:

10

Example Input/Output 2:

Input:

15

10

Output:

5

For example:

Input	Result
30	10
40	

Answer: (penalty regime: 0 %)

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

```
3 {  
4     int x,y,hcf;  
5     scanf("%d %d",&x,&y);  
6     while(x!=y){  
7         if(x>y)  
8             x=x-y;  
9         else  
10            y=y-x;  
11     }  
12     hcf=x;  
13     printf("%d\n",hcf);  
14     return 0;  
15 }
```

	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 %)

```

1 #include<stdio.h>
2 #include<string.h>
3 #include<stdlib.h>
4 #include<ctype.h>
5 int main()
6 {
7     char s[101];
8     scanf("%s",s);
9     int n1=0,n2=0;
10    char op=' ';
11    char*op_ptr=NULL;
12    for(int i=0;s[i]!='\0';i++){
13        if(isalpha(s[i])){
14            op=s[i];
15            op_ptr=&s[i];
16            break;
17        }
18    }
19    if(op_ptr!=NULL){
20        *op_ptr='\0';
21        n1=strtol(s,NULL,10);
22        n2=strtol(op_ptr+1,NULL,10);
23    }

```

```
24     r=0;
25     switch(op){
26         case 'A':
27         case 'a':
28             r=n1+n2;
29             break;
30         case 'S':
31         case 's':
32             r=n1-n2;
33             break;
34         case 'M':
35         case 'm':
36             r=n1*n2;
37             break;
38         case 'D':
39         case 'd':
40             if(n2!=0)
41             {
42                 r=n1/n2;
43             }
44             break;
45     }
46     printf("%d\n",r);
47     return 0;
48 }
```

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

Passed all tests! ✓