Status	Finished
Started	Friday, 31 October 2025, 7:35 PM
Completed	Friday, 31 October 2025, 9:35 PM
Duration	2 hours

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>
int main()
{
    int distance,totalDistance=0;
    float time,totalTime=0.0;
    while(scanf("%d@%f",&distance,&time)==2){
        totalDistance+=distance;
        totalTime+=time;


    }
    if(totalTime>0)
    printf("%.2f kmph",totalDistance/totalTime);
    else
    printf("invalid input");
    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(){
```

```
int main(){
    int x,y,hcf;
    scanf("%d",&x);
    scanf("%d",&y);
    int a=x,b=y;
    while(b!=0){
        int temp=b;
        b=a%b;
        a=temp;
    }
    hcf=a;
    printf("%d",hcf);
    return 0;
}
```

	Input	Expected	Got	
✓	30 40	10	10	✓

Passed all tests! ✓

Question **3**

Incorrect

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>

int main ()
     char S[100];
     scanf("%s",S);
     int num1=0,num2=0;
     char op;
     int i=0;
     while(S[i]>='0'&& S[i]<='9'){
         num1=num1*10+(S[i]-'0');
         i++;
     }

     op=S[i];
     i++;
     while(S[i]!='\0'&& S[i]>='0'&& S[i]<='9'){
         num2=num2*10+(S[i]-'0');
         i++;
     }


     int result=0;
```

```
24      switch(op){
25          case'A':
26          case'a':
27          result=num1+num2;
28          break;
29          case'S':
30          case's':
31          result=num1-num2;
32          break;
33          case'M':
34          case'm':
35          result=num1*num2;
36          break;
37          case'D':
38          case'd':
39          result=num1/num2;
40          break;
41
42      }
43      printf("%d",result);
44      return 0;
45
46  }
```

Syntax Error(s)

```
__tester__.c: In function ‘main’:
__tester__.c:6:6: error: expected declaration specifiers before ‘scanf’
    6 |      scanf("%s",S);
      |      ^~~~
__tester__.c:7:6: error: parameter ‘num1’ is initialized
    7 |      int num1=0,num2=0;
      |      ^~~
__tester__.c:7:6: error: parameter ‘num2’ is initialized
__tester__.c:9:6: error: parameter ‘i’ is initialized
    9 |      int i=0;
      |      ^~~
__tester__.c:10:6: error: expected declaration specifiers before ‘while’
   10 |      while(S[i]>='0'&& S[i]<='9'){
      |      ^~~~~
__tester__.c:15:6: error: expected declaration specifiers before ‘op’
   15 |      op=S[i];
      |      ^~
__tester__.c:16:6: error: expected declaration specifiers before ‘i’
   16 |      i++;
      |      ^
__tester__.c:17:6: error: expected declaration specifiers before ‘while’
   17 |      while(S[i]!='\0'&& S[i]>='0'&& S[i]<='9'){
      |      ^~~~~
__tester__.c:23:6: error: parameter ‘result’ is initialized
   23 |      int result=0;
```

```
      |      ^~~
__tester__.c:24:6: error: expected declaration specifiers before ‘switch’
   24 |      switch(op){
      |      ^~~~~~
__tester__.c:43:6: error: expected declaration specifiers before ‘printf’
   43 |      printf("%d",result);
      |      ^~~~~~
__tester__.c:44:6: error: expected declaration specifiers before ‘return’
   44 |      return 0;
      |      ^~~~~~
__tester__.c:46:1: error: expected declaration specifiers before ‘}’ token
   46 | }
      | ^
__tester__.c:23:10: error: declaration for parameter ‘result’ but no such parameter
   23 |      int result=0;
      |          ^~~~~~
__tester__.c:9:10: error: declaration for parameter ‘i’ but no such parameter
    9 |      int i=0;
      |          ^
__tester__.c:8:11: error: declaration for parameter ‘op’ but no such parameter
    8 |      char op;
      |           ^~
__tester__.c:7:17: error: declaration for parameter ‘num2’ but no such parameter
    7 |      int num1=0,num2=0;
      |                 ^~~~
__tester__.c:7:10: error: declaration for parameter ‘num1’ but no such parameter
    7 |      int num1=0,num2=0;
      |          ^~~~
__tester__.c:5:11: error: declaration for parameter ‘S’ but no such parameter
    5 |      char S[100];
      |           ^
__tester__.c:47: error: expected ‘{’ at end of input
```