| | |
|---|---|
| **Status** | Finished |
| **Started** | Monday, 3 November 2025, 9:02 PM |
| **Completed** | Monday, 3 November 2025, 10:05 PM |
| **Duration** | 1 hour 2 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 %)

```c
#include<stdio.h>
#include<string.h>
int main()
{
char input[1000];
fgets(input,sizeof(input),stdin);
double totalDistance=0,totalTime=0;
char*token=strtok(input," ");
while(token != NULL)
{
double distance,time;
sscanf(token,"%lf@%lf",&distance,&time);
totalDistance +=distance;
totalTime +=time;
token=strtok(NULL,"");
}
double avgSpeed=totalDistance/totalTime;
printf("%.2f kmph\n",avgSpeed);
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<br>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 (x !=y)
    {
        if(x>y)
        x=x-y;
        else
        y=y-x;
    }
    printf("%d\n",x);
    return 0;
}
```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 30<br>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[101],op;
    long long n1=0,n2=0;
    scanf("%s",s);
    int i=0;
    while(isdigit(s[i]))
    {
        n1=n1*10+(s[i++]-'0');


    }
    op=toupper(s[i]);
    i++;
    while(isdigit(s[i]))
    {
        n2=n2*10+(s[i++]-'0');
    }
    switch(op)
    {
```

```
24          case 'A':n1+=n2;break;
25          case 'S':n1-=n2;break;
26          case 'M':n1*=n2;break;
27          case 'D':n1/=n2;break;
28      }
29      printf("%lld\n",n1);
30      return 0;
31  }
```

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

Passed all tests! ✓