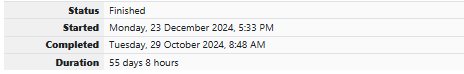
**ASSESSMENT 02**

**PROFIT CALCULATOR**

****

Each Sunday, a newspaper agency sells X copies of a certain newspaper for Rs.A per copy. The cost to the agency of each newspaper is Rs.B . The agency pays a fixed cost for storage, delivery and so on of Rs.100 per Sunday. [The newspaper agency](http://www.rajalakshmicolleges.org/moodle/mod/quiz/view.php?id=54) wants to calculate the profit obtained on Sundays. Can you please help them out by writing a C program to compute the profit given X, A and B.

**Input Format:**

Input consists of 3 integers: X, A and B. X is the number of copies sold, A is the cost per copy and B is the cost the agency spends per copy.

**Output Format:**

Refer Sample Input and Output for exact formatting specifications.

**Sample Input and Output:**

Input

1000

2

1

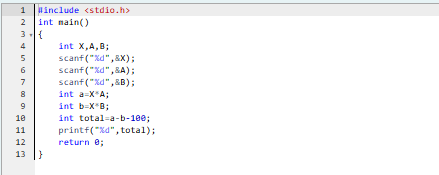
Output

900

**For example:**

| **Input** | **Result** |
| --- | --- |
| 1000  2  1 | 900 |

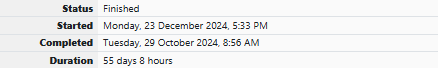
**SOURCE CODE**

****

**OUTPUT**

****

**MONEY WITH BABA**

****

Baba is very kind to beggars and every day Baba donates half of the amount he has when ever a beggar requests him. The money M left in Baba's hand is passed as the input and the number of beggars B who received the alms are passed as the input. The program must print the money Baba had in the beginning of the day.

**Input Format:**

The first line denotes the value of M.  
The second line denotes the value of B.

**Output Format:**

The first line denotes the value of money with Baba in the beginning of the day.

**Example Input/Output:**

Input:

100  
2

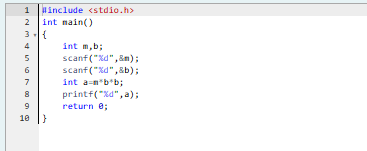
Output:

400

Explanation:

Baba donated to two beggars. So when he encountered second beggar he had 100\*2 = Rs.200 and when he encountered 1st he had 200\*2 = Rs.400.

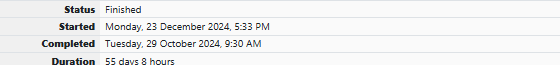
**SOURCE CODE**

****

**OUTPUT**

****

**PUNCTUALITY INCENTIVE**

****

The CEO of company ABC Inc wanted to encourage the employees coming on time to the office. So he announced that for every consecutive day an employee comes on time in a week (starting from Monday to Saturday), he will be awarded Rs.200 more than the previous day as "Punctuality Incentive". The incentive I for the starting day (ie on Monday) is passed as the input to the program. The number of days N an employee came on time consecutively starting from Monday is also passed as the input. The program must calculate and print the "Punctuality Incentive" P of the employee.

**Input Format:**

The first line denotes the value of I.  
The second line denotes the value of N.

**Output Format:**

The first line denotes the value of P.

**Example Input/Output:**

Input:

500  
3

Output:

2100

Explanation:

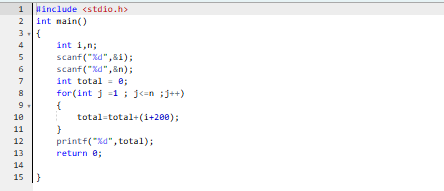
On Monday the employee receives Rs.500, on Tuesday Rs.700, on Wednesday Rs.900

So total = Rs.2100

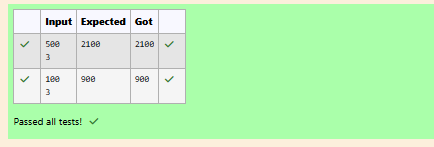
**For example:**

| **Input** | **Result** |
| --- | --- |
| 500  3 | 2100 |
| 100  3 | 900 |

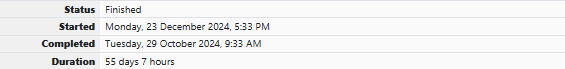
**SOURCE CODE**

****

**OUTPUT**

****

**CHOCOLATES REMAINING**

****

Bajan Lal distributes C chocolates to school N students every Friday. The C chocolates are distributed among N students equally and the remaining chocolates R are given back to Bajan Lal.

As an example if C=100 and N=40, each student receives 2 chocolates and the balance 100-40\*2 = 20 is given back.

If C=205 and N=20, then each student receives 10 chocolates and the balance 205-20\*10 = 5 is given back.

Help the school to calculate the chocolates to be given back when C and N are passed as input.

**Input Format:**

The first line denotes C  
The second line denotes N

**Output Format:**

The first line denotes R - the number of chocolates to be given back.

**Example Input/Output:**

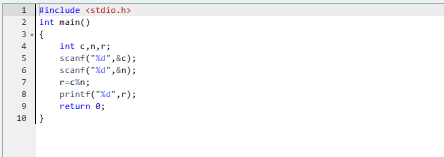
Input:

300  
45

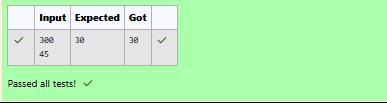
Output:

30

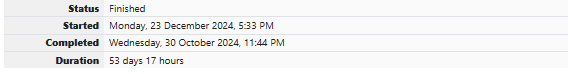
**SOURCE CODE**

****

**OUTPUT**

****

**if CONSTRUCT**

****

The general format of if statement is

if (condition) {

statement-1;

statement-2;

....

statement-n;

}

The if construct is a **selective statement**, the statements within the block are executed only once when the **condition evaluates to true**, otherwise the control goes to the first statement after the if construct.  
  
If only one statement is presented in the if construct then there is no need to specify the braces {, } i.e., if braces are not specified for the if construct, by default the next immediate statement is the only statement considered for the if construct.  
  
Below code prints the number only when it is **divisible by 3**:

#include <**stdio.h**>

**int** main()   
{

**int** num;

printf("Enter a number : ");

scanf("%d", &num);

if (**num % 3 == 0**)

{

printf("Given number %d is divisible by 3", num);

}  
 return 0;

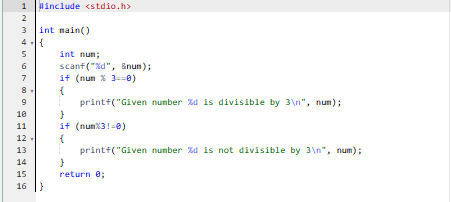
}

In the above code, num % 3 == 0 is the **condition**, which verifies whether the **number is divisible by 3**. Only if the condition returns 1 (true) then the control enters in to the **if-block** and executes the statement.  
  
Fill in the missing code in the below program to check whether the given number is divisible by **3** or not.

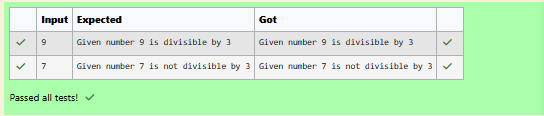
**For example:**

| **Input** | **Result** |
| --- | --- |
| 9 | Given number 9 is divisible by 3 |
| 7 | Given number 7 is not divisible by 3 |

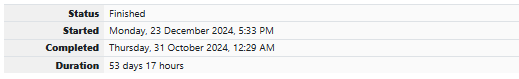
**SOURCE CODE**

****

**OUTPUT**

****

**if – else CONSTRUCT**

****

**QUESTION 1**

The if statement tells a program to execute a certain section of code only if a particular test evaluates to true. if (*expression*) {*statement*}.  
  
Below is a sample code which uses a if statement:

**int** distinction\_marks = 75;

if (marks > distinction\_marks)   
{

printf("User secured distinction.\n");

}

An if statement will execute its block only when condition evaluates to 1 (**true**).  
  
We can also conditionally execute another block when the condition evaluates to 0 (**false**) using the else construct. The else construct must be attached to an if, hence together they are referred to as if-else construct.  
  
The if-else statement provides two different paths of execution depending on the result of the condition.  
  
Below is the general syntax for the if-else statement :

if (expression)   
{

statement-1;

}   
else   
{

statement-2;

}

Below is an example with code:

**int** distinction\_marks = 75;

if (marks > distinction\_marks)   
{

printf("User secured distinction.\n");

}   
else   
{

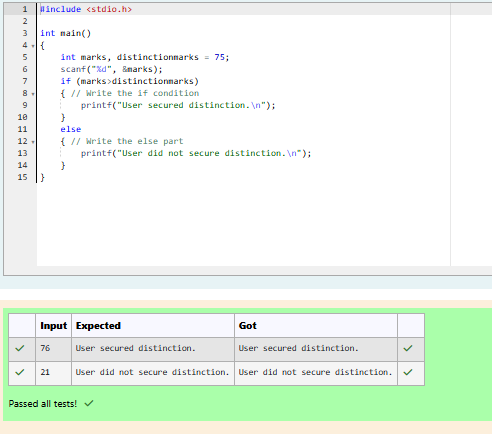
printf("User did not secure distinction.\n");

}

Fill in the missing code in the below program to check whether the user secured distinction or not.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 76 | User secured distinction. |
| 21 | User did not secure distinction. |

****

**QUESTION 2**

Write code which uses an if-else statement to check whether a given account balance is greater or lesser than the minimum balance.  
  
Use the if-else statement and print "Balance is low" if the balance is less than **1000**, otherwise print "Sufficient balance".  
  
For example, if the user gives the **input** as 1500:

1500

then the program should **print** the result as:

Sufficient balance

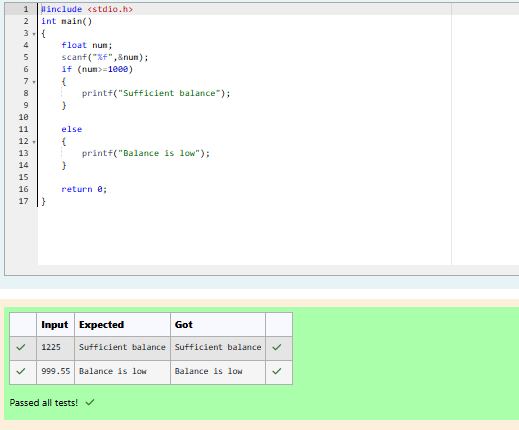
Similarly, if the input is given as 700 then print

Balance is low

[**Hint:** Make sure to read the input as a float value.]

**For example:**

| **Input** | **Result** |
| --- | --- |
| 1225 | Sufficient balance |
| 999.55 | Balance is low |

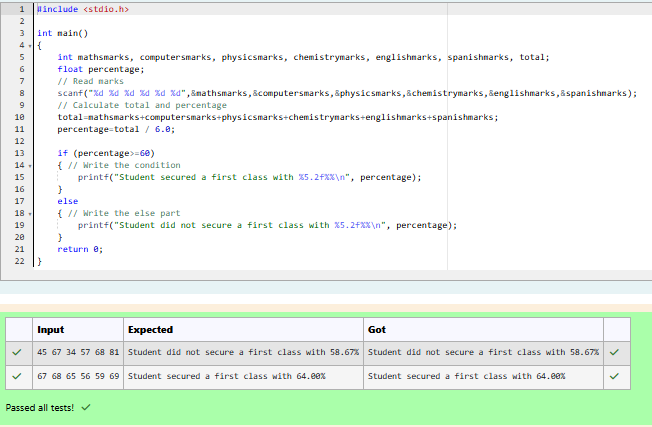
****

**QUESTION 3**

Fill in the missing code in the below program to check whether the student secured first class or not.  
  
**Note-1:** Read **6** subjects marks, find total and percentage, then print the student secured first class or not.  
  
**Note-2:** If percentage is greater than or equal to **60** then print student secured first class and the percentage.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 45 67 34 57 68 81 | Student did not secure a first class with 58.67% |
| 67 68 65 56 59 69 | Student secured a first class with 64.00% |

****

**QUESTION 4**

Write a program which uses an if-else statement to verify and print if the given number is an odd or an even.  
  
For example, if the user gives the **input** as 10:

10

then the program should **print** the result as:

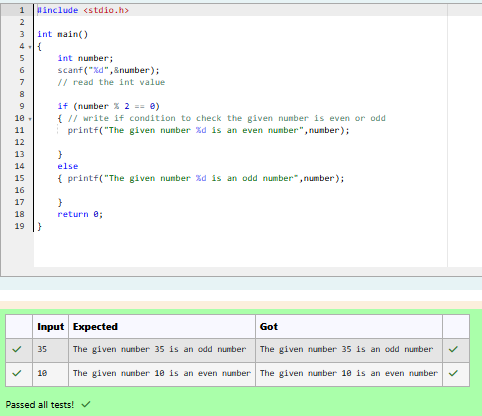
The given number 10 is an even number

If the input is given as 35, then the program should print the result as :

The given number 35 is an odd number

**For example:**

| **Input** | **Result** |
| --- | --- |
| 35 | The given number 35 is an odd number |
| 10 | The given number 10 is an even number |

****

**QUESTION 5**

Write a program which uses an if-else statement to verify if the given character is an alphabet or not.  
  
For example, if the user gives the **input** as W:

W

then the program should **print** the result as:

Given character W is an alphabet

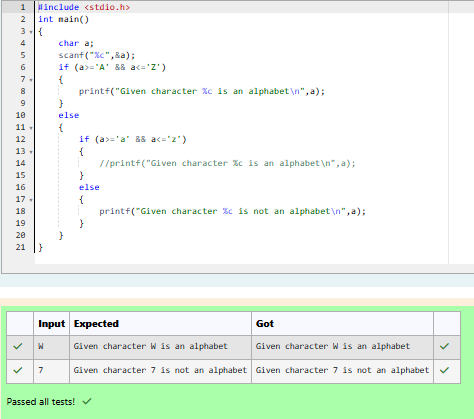
If the input us given as 7, then print the result as:

Given character 7 is not an alphabet

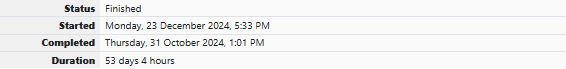
[**Hint:** The ASCII values of alphabets '**A**' to '**Z**' are 65 to 90 and '**a**' to '**z**' are 97 to 122.]

**For example:**

| **Input** | **Result** |
| --- | --- |
| W | Given character W is an alphabet |
| 7 | Given character 7 is not an alphabet |



**NESTED if – else CONSTRUCT**

****

When an if-else construct appear as a statement within another if-block or a else-block, it is referred to as nesting of if-else construct.

Below is an example of a **nested if-else** construct:

if (expression\_1)   
{

if (expression\_2)   
 {

if (expression\_3)   
 {

statement\_1;

}   
 else   
 {

statement\_2;

}

}   
 else   
 {

statement\_3;

}

}

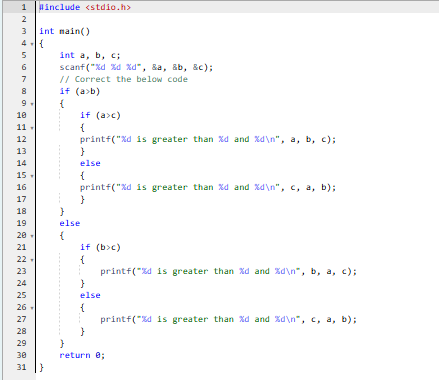
In the above syntax, the **statement\_2** will be executed only when the conditions in expression\_1, expression\_2 and expression\_3 evaluates to 1 (true).

Fill in the missing code in the below program to find the **largest** of three numbers using nested if-else.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 23 56 77 | 77 is greater than 23 and 56 |

**SOURCE CODE**

****

**OUTPUT**

****