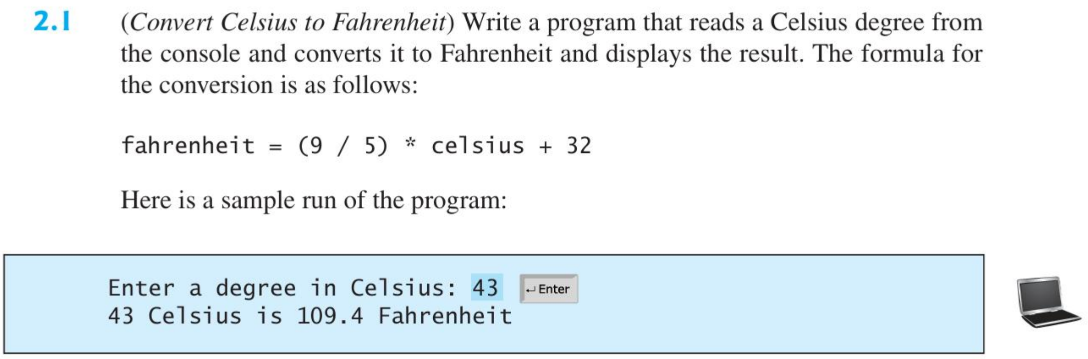
Date: 16/09/2024

Assignment 01

Name: MEHRIN FARZANA
ID: 2101013

Problem no. 1.



2.1 (*Convert Celsius to Fahrenheit*) Write a program that reads a Celsius degree from the console and converts it to Fahrenheit and displays the result. The formula for the conversion is as follows:

```
fahrenheit = (9 / 5) * celsius + 32
```

Here is a sample run of the program:

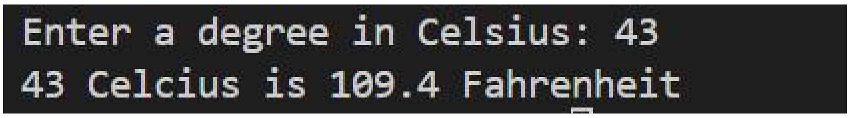
```
Enter a degree in Celsius: 43 ↵Enter
43 Celsius is 109.4 Fahrenheit
```

Solution:

Code:

```
cel = int(input("Enter a degree in Celsius: "))
far = round((9 / 5) * cel + 32, 2)
print(cel, end=" ")
print("Celcius is ", end="")
print(far, end=" ")
print("Fahrenheit")
```

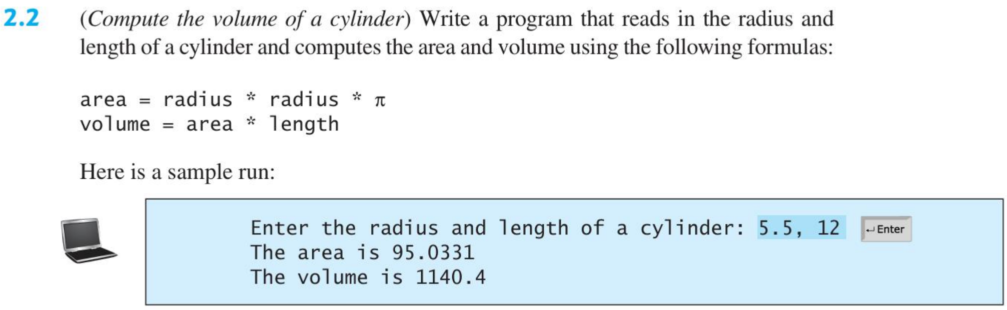
Output:



```
Enter a degree in Celsius: 43
43 Celcius is 109.4 Fahrenheit
```

Figure 1.1: Output

Problem no. 2.



2.2 (*Compute the volume of a cylinder*) Write a program that reads in the radius and length of a cylinder and computes the area and volume using the following formulas:

```
area = radius * radius * π
volume = area * length
```

Here is a sample run:

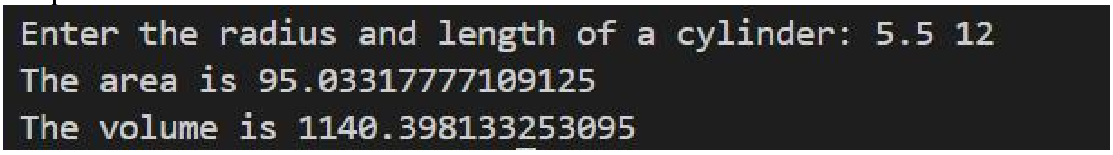
```
Enter the radius and length of a cylinder: 5.5, 12 ↵Enter
The area is 95.0331
The volume is 1140.4
```

Solution:

Code:

```
import math
r, l = map(float, input("Enter the radius and length of a
cylinder: ").split())
a = r * r * math.pi
v = a * l
print("The area is ", end="")
print(a)
print("The volume is ", end="")
print(v)
```

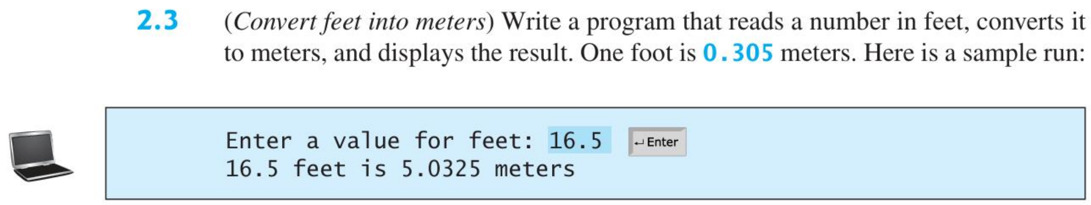
Output:



```
Enter the radius and length of a cylinder: 5.5 12
The area is 95.03317777109125
The volume is 1140.398133253095
```

Figure 2.1: Output

Problem no. 3.

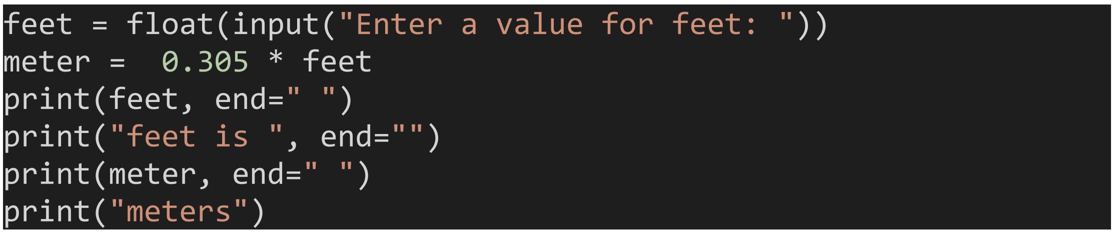


2.3 (*Convert feet into meters*) Write a program that reads a number in feet, converts it to meters, and displays the result. One foot is **0.305** meters. Here is a sample run:

```
Enter a value for feet: 16.5
16.5 feet is 5.0325 meters
```

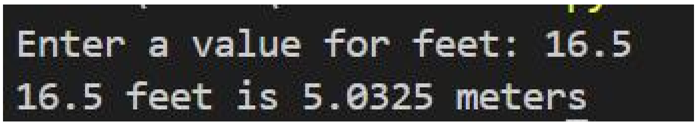
Solution:

Code:



```
feet = float(input("Enter a value for feet: "))
meter =  0.305 * feet
print(feet, end=" ")
print("feet is ", end="")
print(meter, end=" ")
print("meters")
```

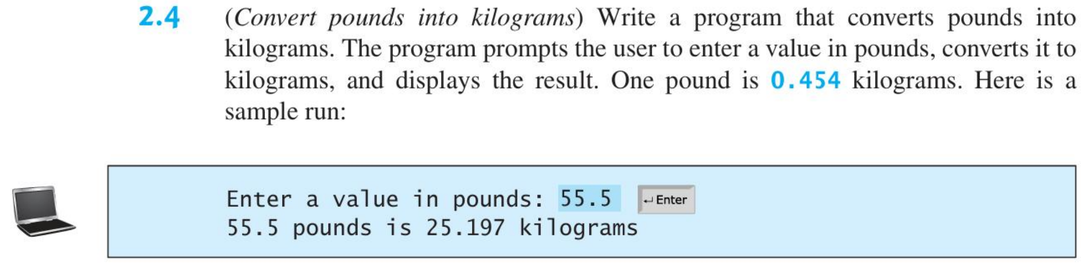
Output:



```
Enter a value for feet: 16.5
16.5 feet is 5.0325 meters
```

Figure 3.1: Output

Problem no. 4.

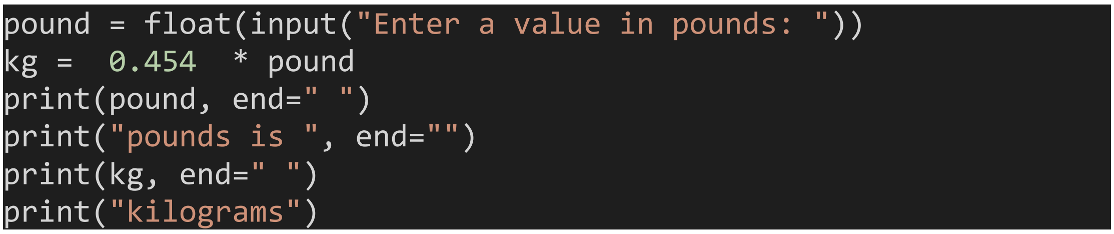


2.4 (*Convert pounds into kilograms*) Write a program that converts pounds into kilograms. The program prompts the user to enter a value in pounds, converts it to kilograms, and displays the result. One pound is **0.454** kilograms. Here is a sample run:

```
Enter a value in pounds: 55.5 ↵Enter
55.5 pounds is 25.197 kilograms
```

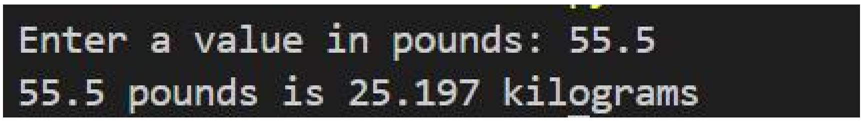
Solution:

Code:



```
pound = float(input("Enter a value in pounds: "))
kg =  0.454  * pound
print(pound, end=" ")
print("pounds is ", end="")
print(kg, end=" ")
print("kilograms")
```

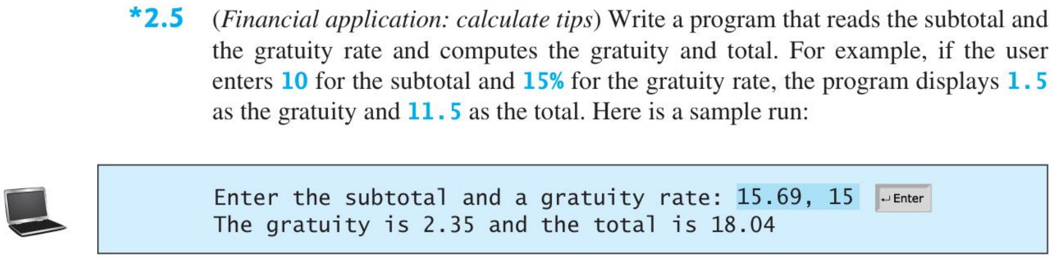
Output:



```
Enter a value in pounds: 55.5
55.5 pounds is 25.197 kilograms
```

Figure 4.1: Output

Problem no. 5.

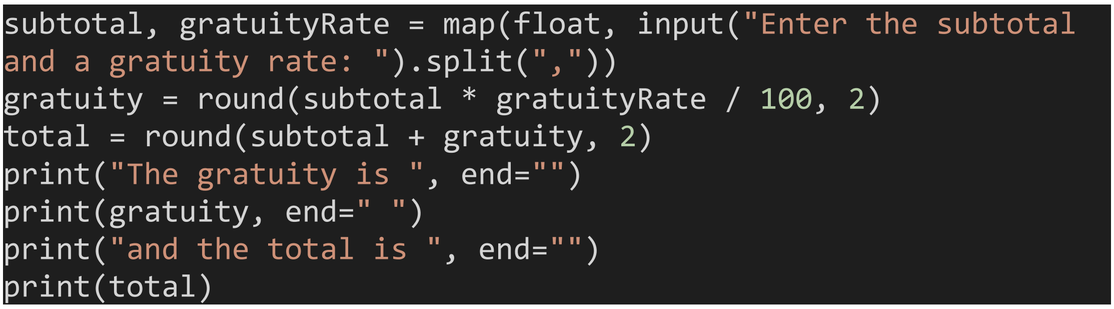


***2.5** (*Financial application: calculate tips*) Write a program that reads the subtotal and the gratuity rate and computes the gratuity and total. For example, if the user enters `10` for the subtotal and `15%` for the gratuity rate, the program displays `1.5` as the gratuity and `11.5` as the total. Here is a sample run:

```
Enter the subtotal and a gratuity rate: 15.69, 15 ↵Enter
The gratuity is 2.35 and the total is 18.04
```

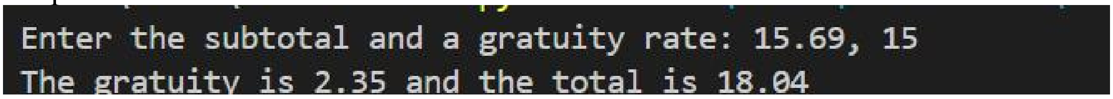
Solution:

Code:



```
subtotal, gratuityRate = map(float, input("Enter the subtotal and a gratuity rate: ").split(","))
gratuity = round(subtotal * gratuityRate / 100, 2)
total = round(subtotal + gratuity, 2)
print("The gratuity is ", end="")
print(gratuity, end=" ")
print("and the total is ", end="")
print(total)
```

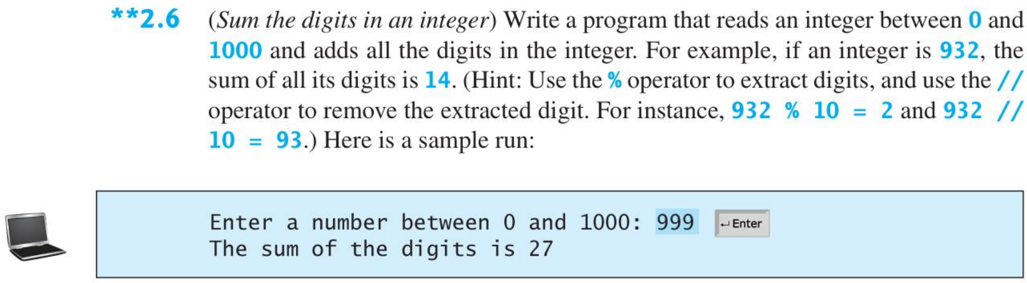
Output:



```
Enter the subtotal and a gratuity rate: 15.69, 15
The gratuity is 2.35 and the total is 18.04
```

Figure 5.1: Output

Problem no. 6.

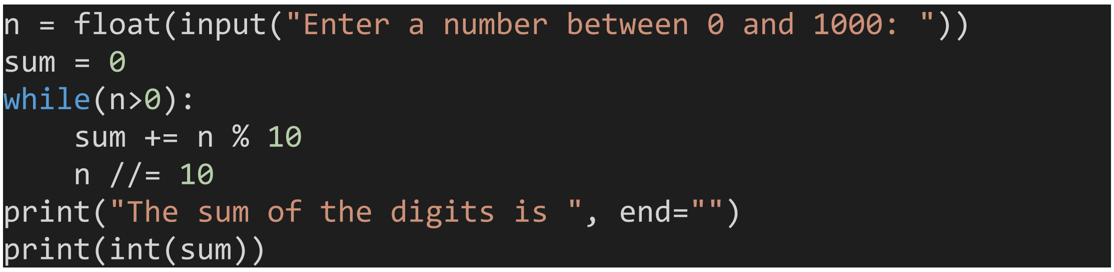


**2.6 (*Sum the digits in an integer*) Write a program that reads an integer between 0 and 1000 and adds all the digits in the integer. For example, if an integer is 932, the sum of all its digits is 14. (Hint: Use the % operator to extract digits, and use the // operator to remove the extracted digit. For instance, 932 % 10 = 2 and 932 // 10 = 93.) Here is a sample run:

```
Enter a number between 0 and 1000: 999 ↵Enter
The sum of the digits is 27
```

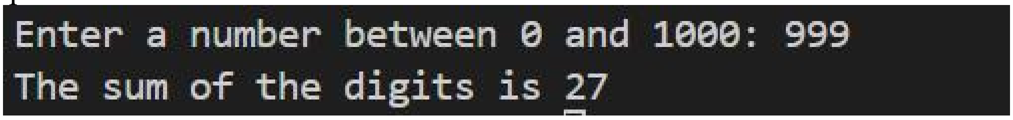
Solution:

Code:



```
n = float(input("Enter a number between 0 and 1000: "))
sum = 0
while(n>0):
    sum += n % 10
    n //= 10
print("The sum of the digits is ", end="")
print(int(sum))
```

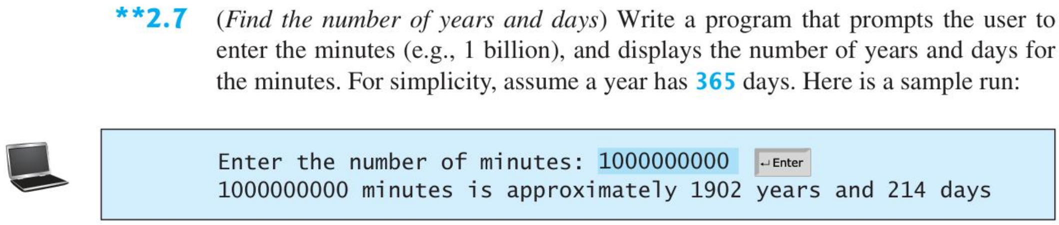
Output:



```
Enter a number between 0 and 1000: 999
The sum of the digits is 27
```

Figure 6.1: Output

Problem no. 7.

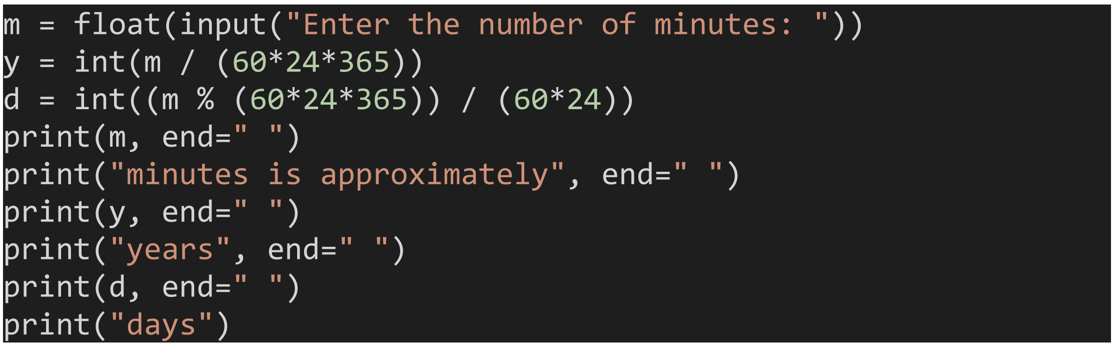


****2.7** (*Find the number of years and days*) Write a program that prompts the user to enter the minutes (e.g., 1 billion), and displays the number of years and days for the minutes. For simplicity, assume a year has **365** days. Here is a sample run:

```
Enter the number of minutes: 1000000000 ↵Enter
1000000000 minutes is approximately 1902 years and 214 days
```

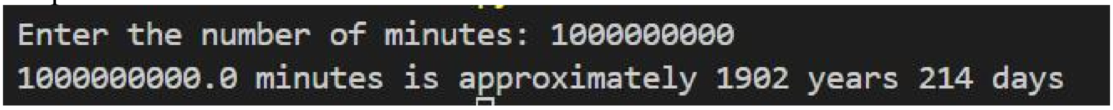
Solution:

Code:



```
m = float(input("Enter the number of minutes: "))
y = int(m / (60*24*365))
d = int((m % (60*24*365)) / (60*24))
print(m, end=" ")
print("minutes is approximately", end=" ")
print(y, end=" ")
print("years", end=" ")
print(d, end=" ")
print("days")
```

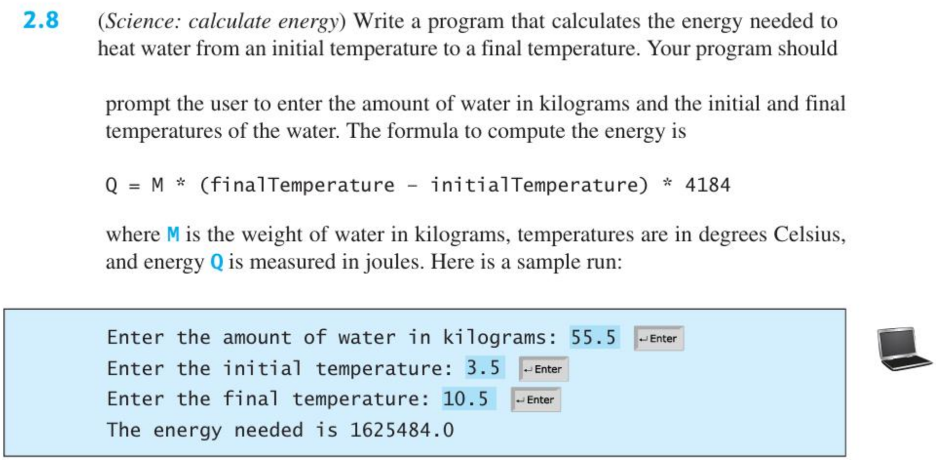
Output:



```
Enter the number of minutes: 1000000000
1000000000.0 minutes is approximately 1902 years 214 days
```

Figure 7.1: Output

Problem no. 8.



2.8 (*Science: calculate energy*) Write a program that calculates the energy needed to heat water from an initial temperature to a final temperature. Your program should prompt the user to enter the amount of water in kilograms and the initial and final temperatures of the water. The formula to compute the energy is

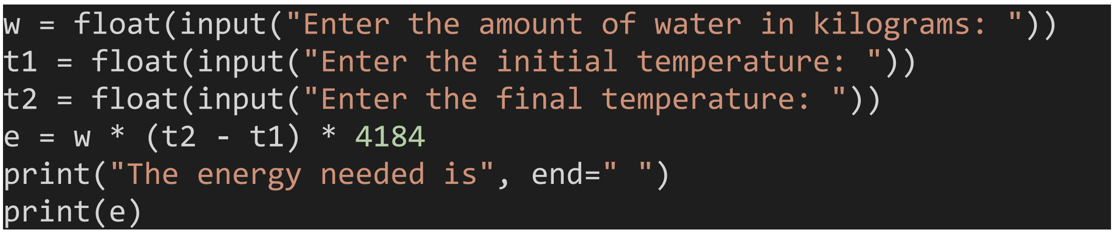
```
Q = M * (finalTemperature – initialTemperature) * 4184
```

where `M` is the weight of water in kilograms, temperatures are in degrees Celsius, and energy `Q` is measured in joules. Here is a sample run:

```
Enter the amount of water in kilograms: 55.5 ↵Enter
Enter the initial temperature: 3.5 ↵Enter
Enter the final temperature: 10.5 ↵Enter
The energy needed is 1625484.0
```

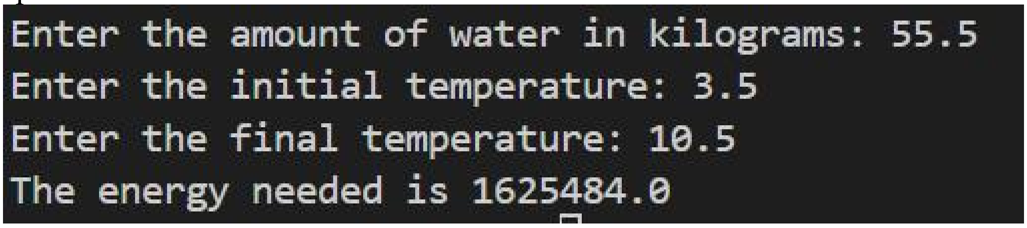
Solution:

Code:



```
w = float(input("Enter the amount of water in kilograms: "))
t1 = float(input("Enter the initial temperature: "))
t2 = float(input("Enter the final temperature: "))
e = w * (t2 - t1) * 4184
print("The energy needed is", end=" ")
print(e)
```

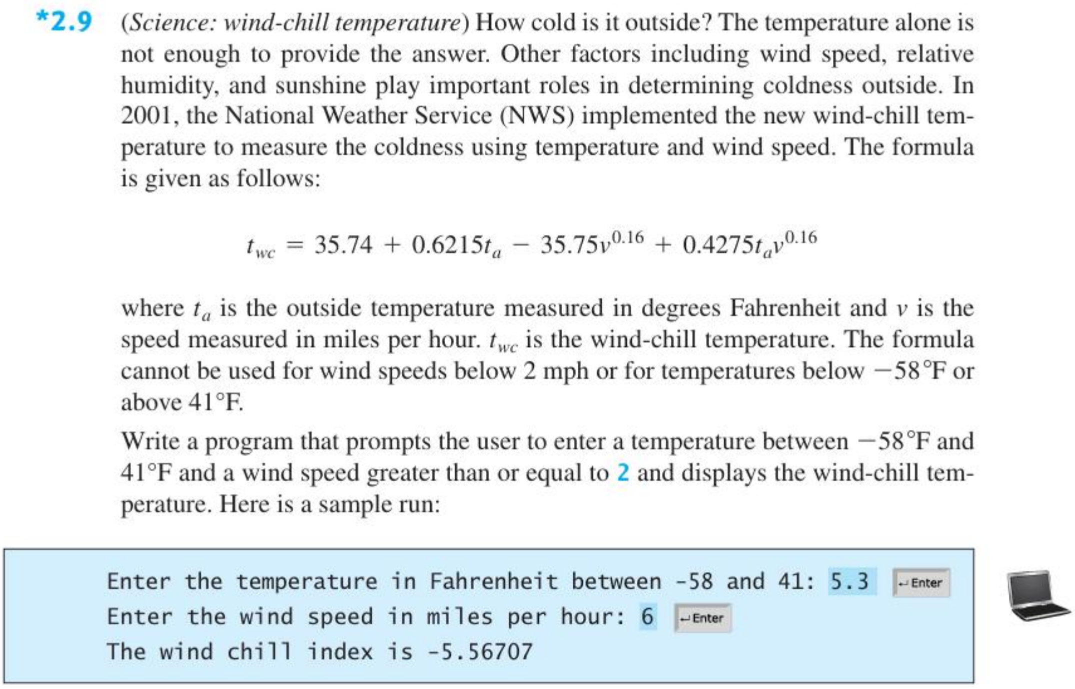
Output:



```
Enter the amount of water in kilograms: 55.5
Enter the initial temperature: 3.5
Enter the final temperature: 10.5
The energy needed is 1625484.0
```

Figure 8.1: Output

Problem no. 9.



***2.9** (*Science: wind-chill temperature*) How cold is it outside? The temperature alone is not enough to provide the answer. Other factors including wind speed, relative humidity, and sunshine play important roles in determining coldness outside. In 2001, the National Weather Service (NWS) implemented the new wind-chill temperature to measure the coldness using temperature and wind speed. The formula is given as follows:

$$t_{wc} = 35.74 + 0.6215t_a - 35.75v^{0.16} + 0.4275t_a v^{0.16}$$

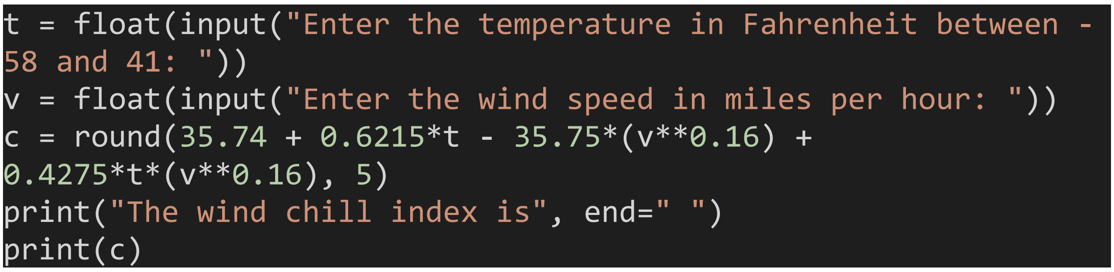
where t_a is the outside temperature measured in degrees Fahrenheit and v is the speed measured in miles per hour. t_{wc} is the wind-chill temperature. The formula cannot be used for wind speeds below 2 mph or for temperatures below −58°F or above 41°F.

Write a program that prompts the user to enter a temperature between −58°F and 41°F and a wind speed greater than or equal to 2 and displays the wind-chill temperature. Here is a sample run:

```
Enter the temperature in Fahrenheit between -58 and 41: 5.3 ↵Enter
Enter the wind speed in miles per hour: 6 ↵Enter
The wind chill index is -5.56707
```

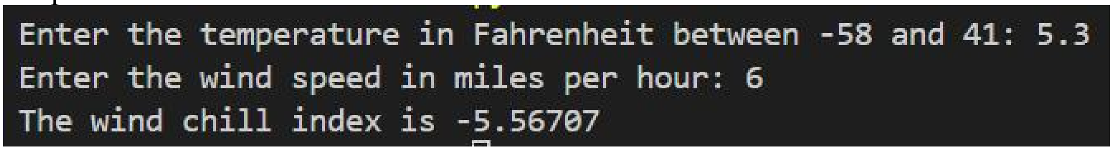
Solution:

Code:



```
t = float(input("Enter the temperature in Fahrenheit between -58 and 41: "))
v = float(input("Enter the wind speed in miles per hour: "))
c = round(35.74 + 0.6215*t - 35.75*(v**0.16) + 0.4275*t*(v**0.16), 5)
print("The wind chill index is", end=" ")
print(c)
```

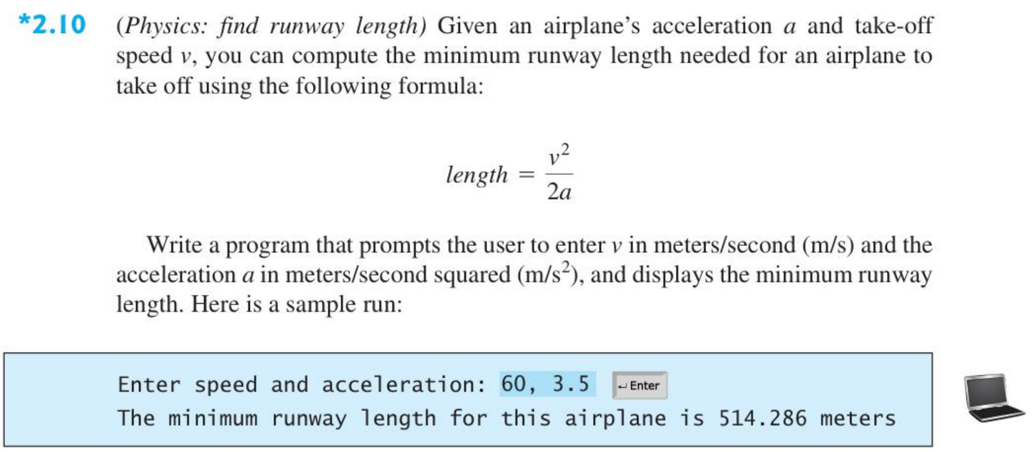
Output:



```
Enter the temperature in Fahrenheit between -58 and 41: 5.3
Enter the wind speed in miles per hour: 6
The wind chill index is -5.56707
```

Figure 9.1: Output

Problem no. 10.



***2.10** (*Physics: find runway length*) Given an airplane's acceleration a and take-off speed v, you can compute the minimum runway length needed for an airplane to take off using the following formula:

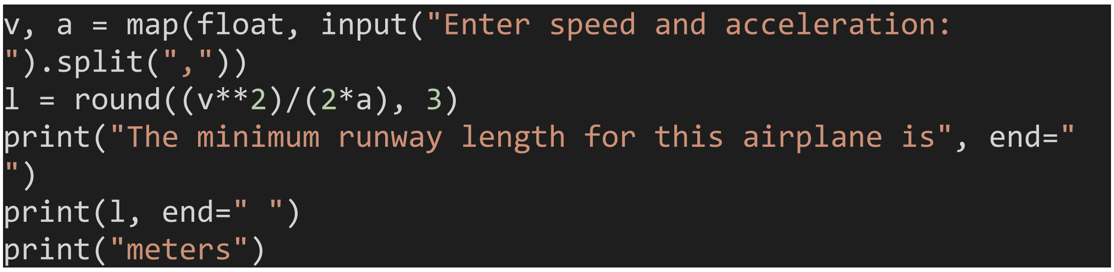
$$length = \frac{v^2}{2a}$$

Write a program that prompts the user to enter v in meters/second (m/s) and the acceleration a in meters/second squared (m/s^2), and displays the minimum runway length. Here is a sample run:

```
Enter speed and acceleration: 60, 3.5 ↵Enter
The minimum runway length for this airplane is 514.286 meters
```

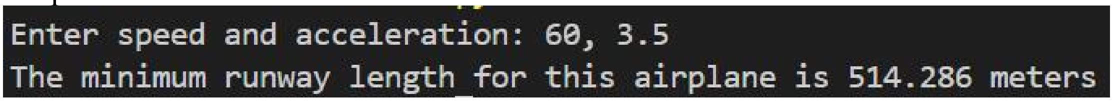
Solution:

Code:



```python
v, a = map(float, input("Enter speed and acceleration: ").split(","))
l = round((v**2)/(2*a), 3)
print("The minimum runway length for this airplane is", end=" ")
print(l, end=" ")
print("meters")
```

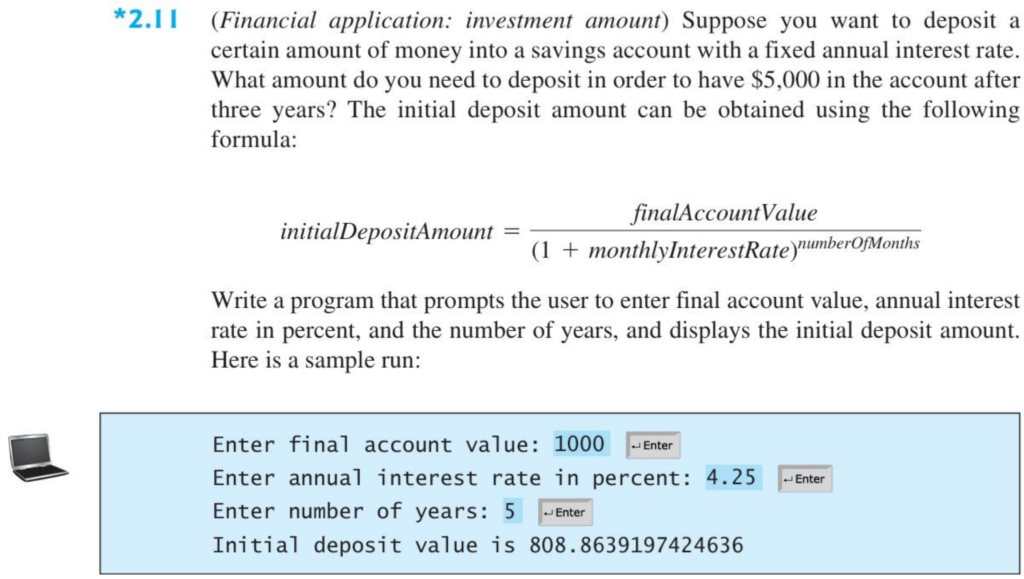
Output:



```
Enter speed and acceleration: 60, 3.5
The minimum runway length for this airplane is 514.286 meters
```

Figure 10.1: Output

Problem no. 11.



***2.11** (*Financial application: investment amount*) Suppose you want to deposit a certain amount of money into a savings account with a fixed annual interest rate. What amount do you need to deposit in order to have $5,000 in the account after three years? The initial deposit amount can be obtained using the following formula:

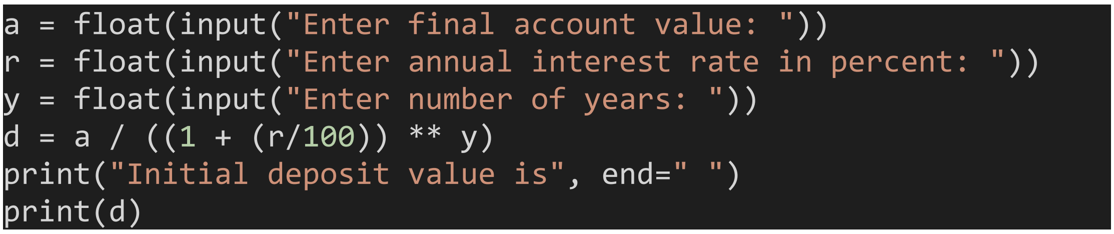
$$initialDepositAmount = \frac{finalAccountValue}{(1 + monthlyInterestRate)^{numberOfMonths}}$$

Write a program that prompts the user to enter final account value, annual interest rate in percent, and the number of years, and displays the initial deposit amount. Here is a sample run:

```
Enter final account value: 1000 ↵Enter
Enter annual interest rate in percent: 4.25 ↵Enter
Enter number of years: 5 ↵Enter
Initial deposit value is 808.8639197424636
```

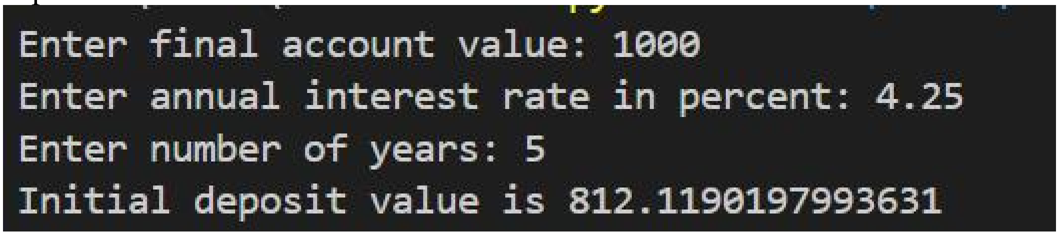
Solution:

Code:



```python
a = float(input("Enter final account value: "))
r = float(input("Enter annual interest rate in percent: "))
y = float(input("Enter number of years: "))
d = a / ((1 + (r/100)) ** y)
print("Initial deposit value is", end=" ")
print(d)
```

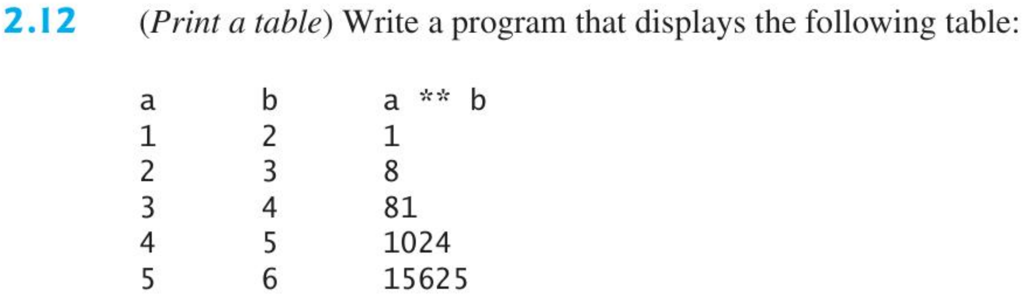
Output:



```
Enter final account value: 1000
Enter annual interest rate in percent: 4.25
Enter number of years: 5
Initial deposit value is 812.1190197993631
```

Figure 11.1: Output

Problem no. 12.



2.12 (*Print a table*) Write a program that displays the following table:

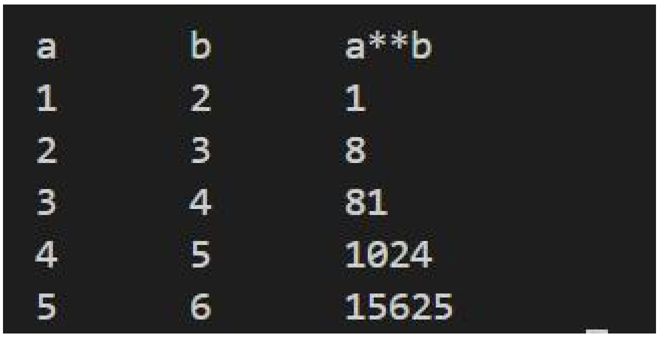
```
a      b      a ** b
1      2      1
2      3      8
3      4      81
4      5      1024
5      6      15625
```

Solution:

Code:

```python
print('a' , end="      ")
print("b" , end="      ")
print("a**b")
for i in range(1,6):
    print(i , end="      ")
    print(i+1 , end="      ")
    print(i**(i+1))
```

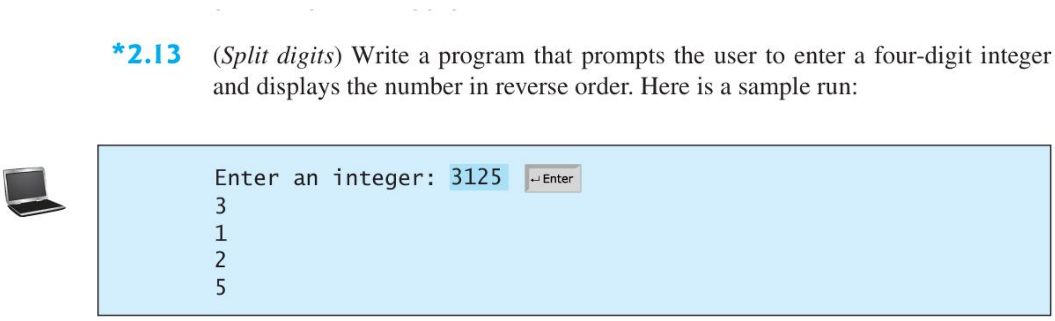
Output:



```
a      b      a**b
1      2      1
2      3      8
3      4      81
4      5      1024
5      6      15625
```

Figure 12.1: Output

Problem no. 13.

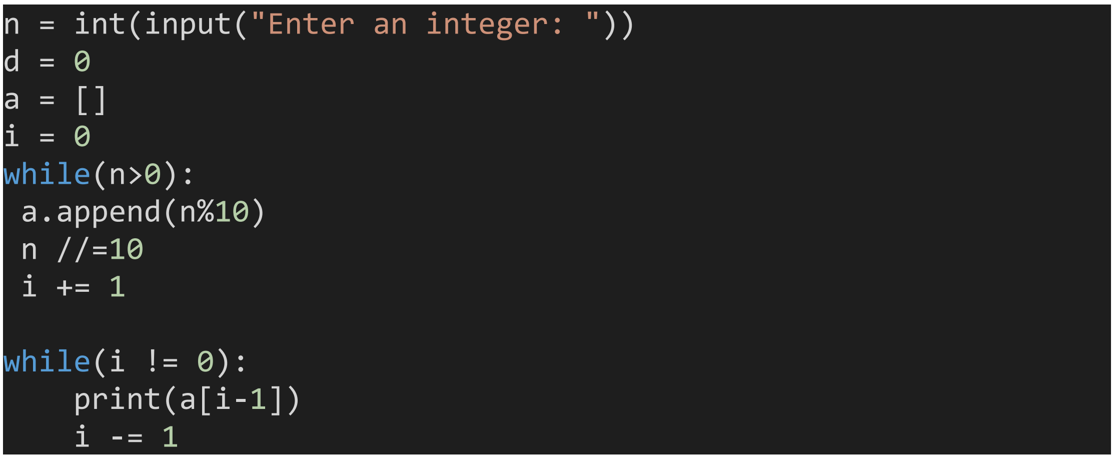


***2.13** (*Split digits*) Write a program that prompts the user to enter a four-digit integer and displays the number in reverse order. Here is a sample run:

```
Enter an integer: 3125 ↵Enter
3
1
2
5
```

Solution:

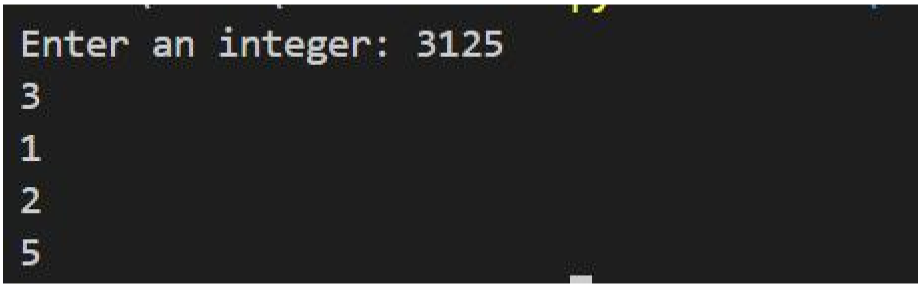
Code:



```
n = int(input("Enter an integer: "))
d = 0
a = []
i = 0
while(n>0):
 a.append(n%10)
 n //=10
 i += 1

while(i != 0):
    print(a[i-1])
    i -= 1
```

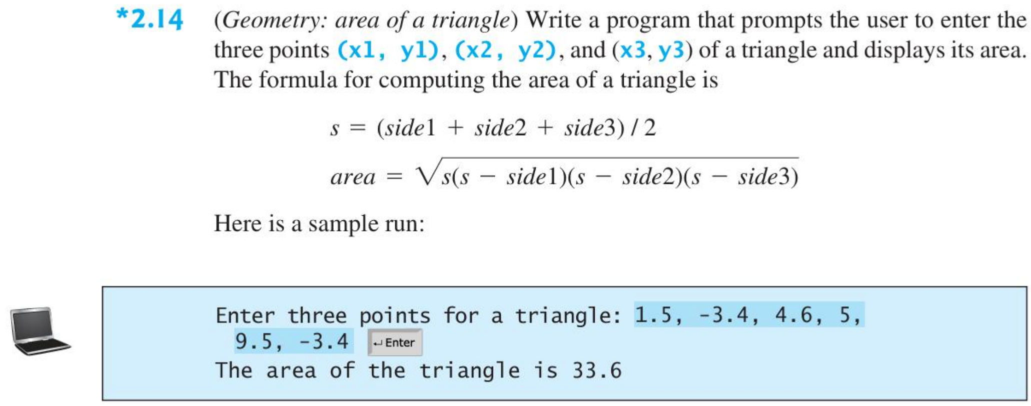
Output:



```
Enter an integer: 3125
3
1
2
5
```

Figure 13.1: Output

Problem no. 14.



***2.14** (*Geometry: area of a triangle*) Write a program that prompts the user to enter the three points **(x1, y1)**, **(x2, y2)**, and (**x3**, **y3**) of a triangle and displays its area. The formula for computing the area of a triangle is

$$s = (side1 + side2 + side3) / 2$$

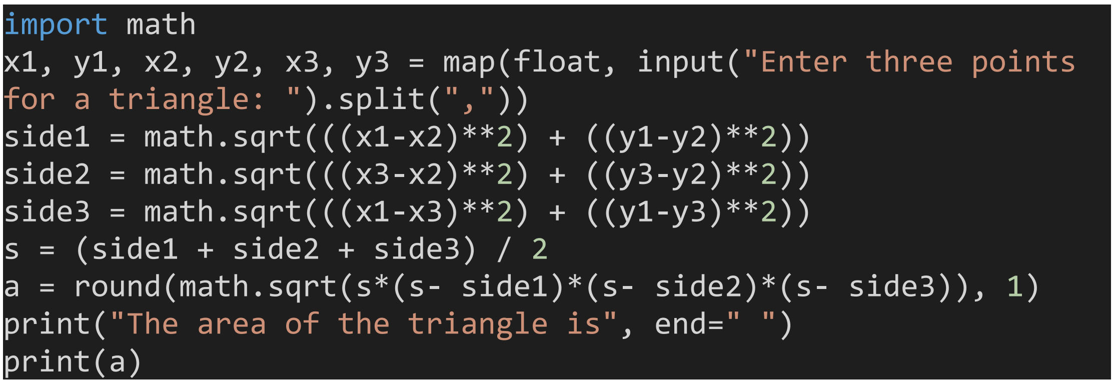
$$area = \sqrt{s(s - side1)(s - side2)(s - side3)}$$

Here is a sample run:

```
Enter three points for a triangle: 1.5, -3.4, 4.6, 5,
  9.5, -3.4 ↵Enter
The area of the triangle is 33.6
```

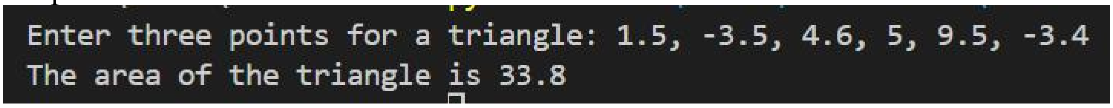
Solution:

Code:



```
import math
x1, y1, x2, y2, x3, y3 = map(float, input("Enter three points for a triangle: ").split(","))
side1 = math.sqrt(((x1-x2)**2) + ((y1-y2)**2))
side2 = math.sqrt(((x3-x2)**2) + ((y3-y2)**2))
side3 = math.sqrt(((x1-x3)**2) + ((y1-y3)**2))
s = (side1 + side2 + side3) / 2
a = round(math.sqrt(s*(s- side1)*(s- side2)*(s- side3)), 1)
print("The area of the triangle is", end=" ")
print(a)
```

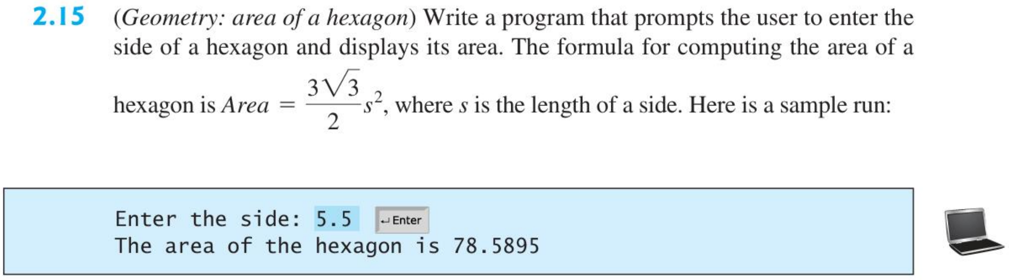
Output:



```
Enter three points for a triangle: 1.5, -3.5, 4.6, 5, 9.5, -3.4
The area of the triangle is 33.8
```

Figure 14.1: Output

Problem no. 15.

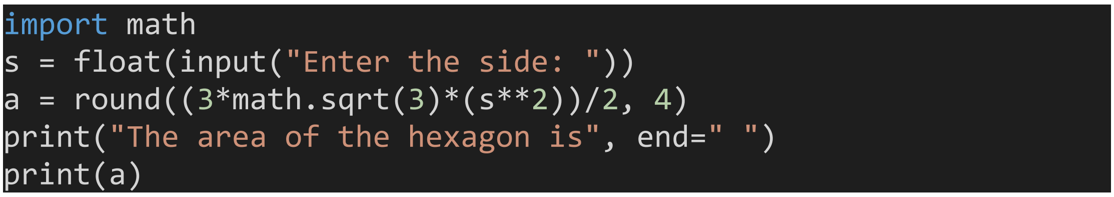


2.15 (*Geometry: area of a hexagon*) Write a program that prompts the user to enter the side of a hexagon and displays its area. The formula for computing the area of a hexagon is $Area = \frac{3\sqrt{3}}{2}s^2$, where s is the length of a side. Here is a sample run:

```
Enter the side: 5.5 ↵Enter
The area of the hexagon is 78.5895
```

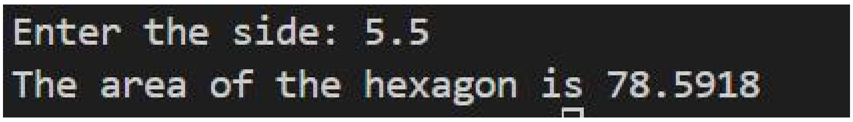
Solution:

Code:



```
import math
s = float(input("Enter the side: "))
a = round((3*math.sqrt(3)*(s**2))/2, 4)
print("The area of the hexagon is", end=" ")
print(a)
```

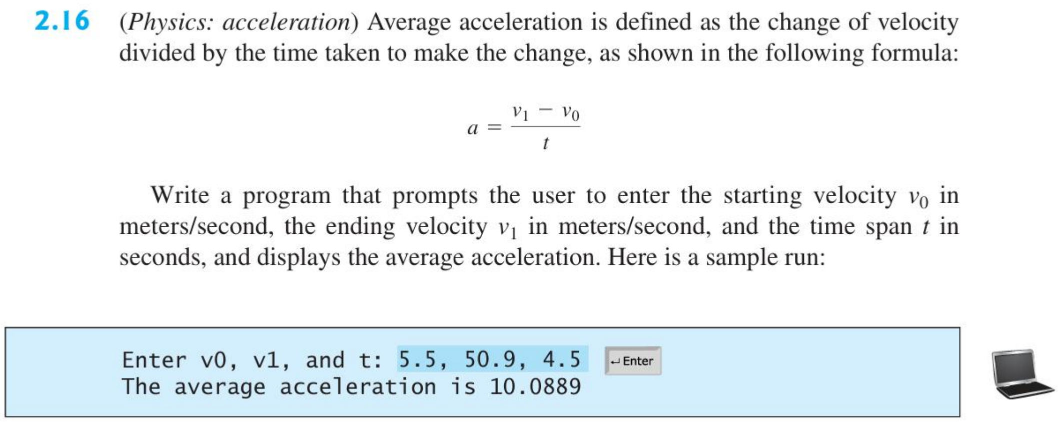
Output:



```
Enter the side: 5.5
The area of the hexagon is 78.5918
```

Figure 15.1: Output

Problem no. 16.



2.16 (*Physics: acceleration*) Average acceleration is defined as the change of velocity divided by the time taken to make the change, as shown in the following formula:

$$a = \frac{v_1 - v_0}{t}$$

Write a program that prompts the user to enter the starting velocity v_0 in meters/second, the ending velocity v_1 in meters/second, and the time span t in seconds, and displays the average acceleration. Here is a sample run:

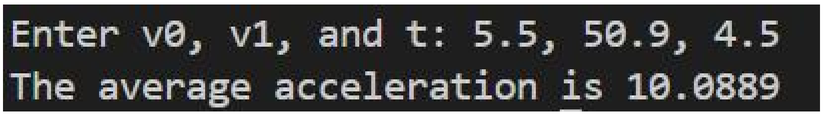
```
Enter v0, v1, and t: 5.5, 50.9, 4.5 ↵Enter
The average acceleration is 10.0889
```

Solution:

Code:

```
v0, v1, t = map(float, input("Enter v0, v1, and t: ").
split(","))
a = round((v1-v0)/t, 4)
print("The average acceleration is", end=" ")
print(a)
```

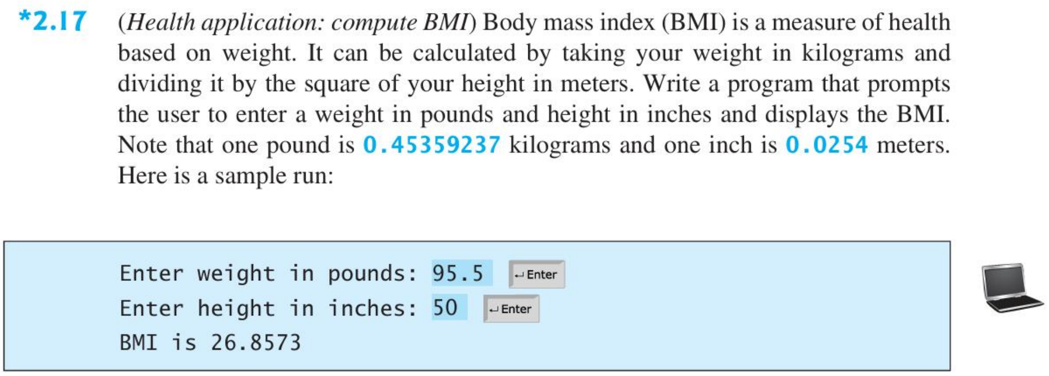
Output:



```
Enter v0, v1, and t: 5.5, 50.9, 4.5
The average acceleration is 10.0889
```

Figure 16.1: Output

Problem no. 17.

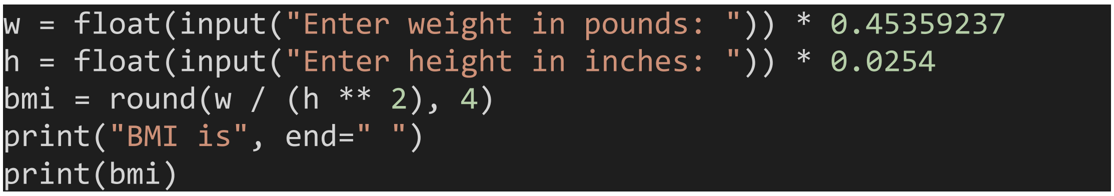


***2.17** (*Health application: compute BMI*) Body mass index (BMI) is a measure of health based on weight. It can be calculated by taking your weight in kilograms and dividing it by the square of your height in meters. Write a program that prompts the user to enter a weight in pounds and height in inches and displays the BMI. Note that one pound is `0.45359237` kilograms and one inch is `0.0254` meters. Here is a sample run:

```
Enter weight in pounds: 95.5 ↵Enter
Enter height in inches: 50 ↵Enter
BMI is 26.8573
```

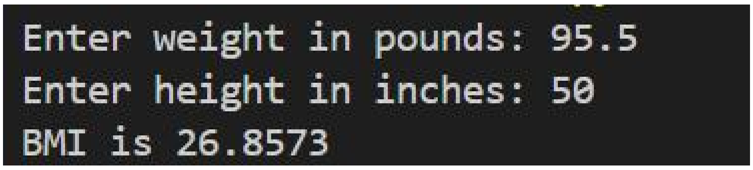
Solution:

Code:



```
w = float(input("Enter weight in pounds: ")) * 0.45359237
h = float(input("Enter height in inches: ")) * 0.0254
bmi = round(w / (h ** 2), 4)
print("BMI is", end=" ")
print(bmi)
```

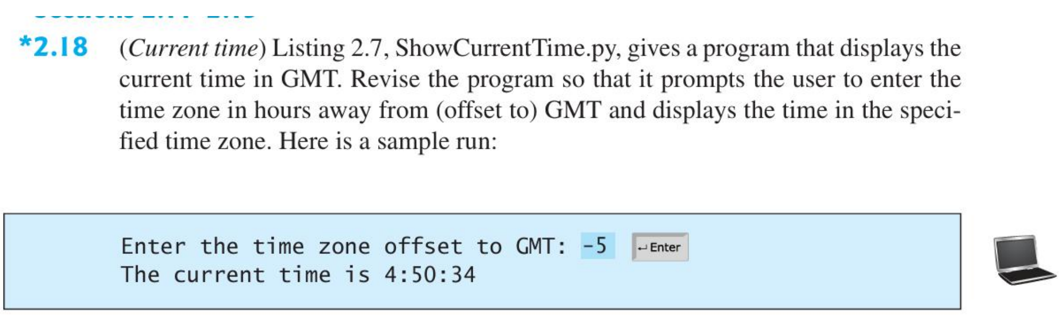
Output:



```
Enter weight in pounds: 95.5
Enter height in inches: 50
BMI is 26.8573
```

Figure 17.1: Output

Problem no. 18.

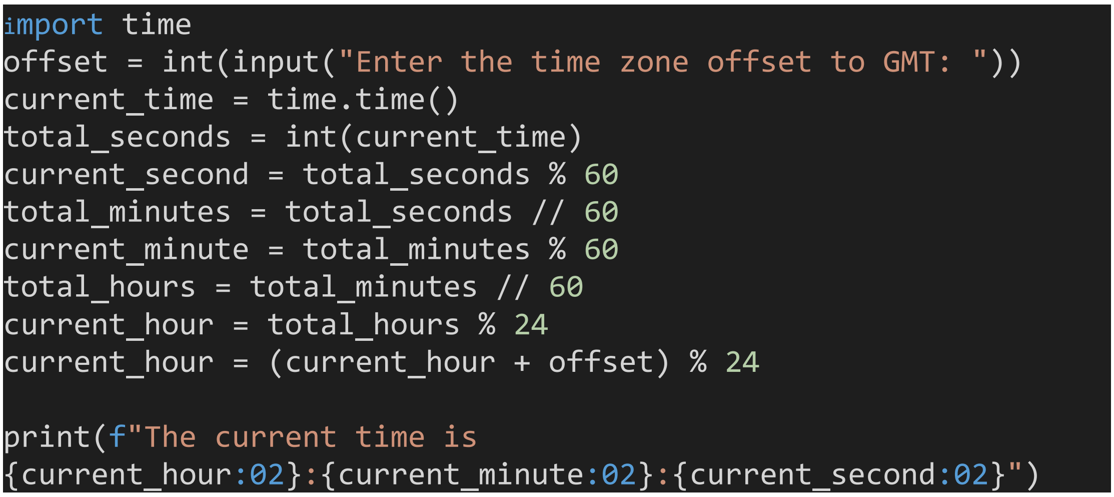


***2.18** (*Current time*) Listing 2.7, ShowCurrentTime.py, gives a program that displays the current time in GMT. Revise the program so that it prompts the user to enter the time zone in hours away from (offset to) GMT and displays the time in the specified time zone. Here is a sample run:

```
Enter the time zone offset to GMT: -5 ↵Enter
The current time is 4:50:34
```

Solution:

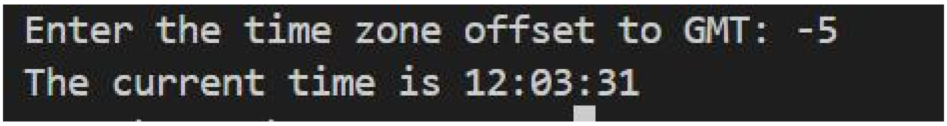
Code:



```
import time
offset = int(input("Enter the time zone offset to GMT: "))
current_time = time.time()
total_seconds = int(current_time)
current_second = total_seconds % 60
total_minutes = total_seconds // 60
current_minute = total_minutes % 60
total_hours = total_minutes // 60
current_hour = total_hours % 24
current_hour = (current_hour + offset) % 24

print(f"The current time is
{current_hour:02}:{current_minute:02}:{current_second:02}")
```

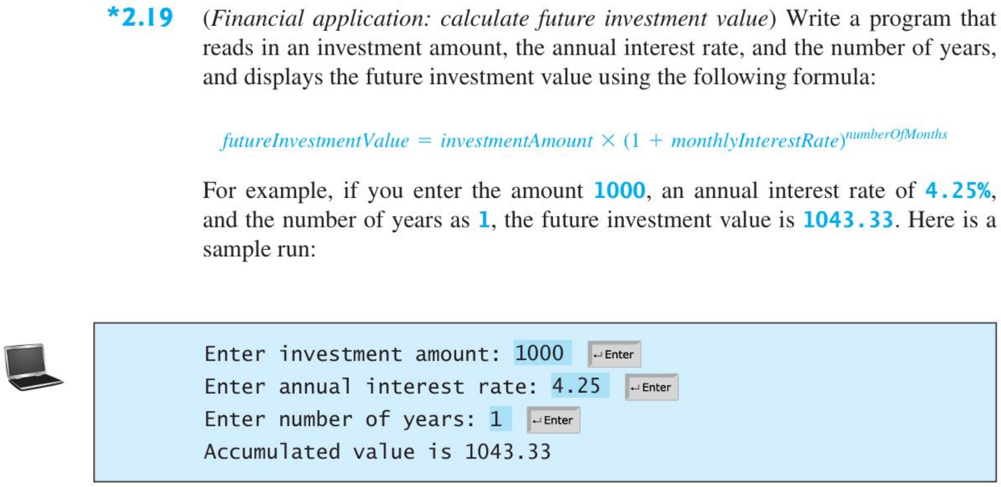
Output:



```
Enter the time zone offset to GMT: -5
The current time is 12:03:31
```

Figure 18.1: Output

Problem no. 19.



***2.19** (*Financial application: calculate future investment value*) Write a program that reads in an investment amount, the annual interest rate, and the number of years, and displays the future investment value using the following formula:

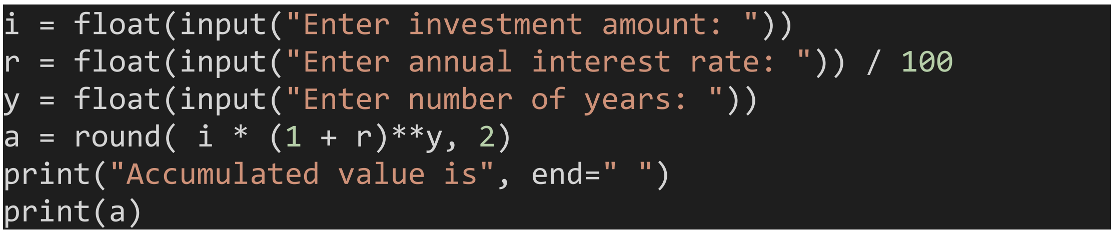
$$futureInvestmentValue = investmentAmount \times (1 + monthlyInterestRate)^{numberOfMonths}$$

For example, if you enter the amount **1000**, an annual interest rate of **4.25%**, and the number of years as **1**, the future investment value is **1043.33**. Here is a sample run:

```
Enter investment amount: 1000 ↵Enter
Enter annual interest rate: 4.25 ↵Enter
Enter number of years: 1 ↵Enter
Accumulated value is 1043.33
```

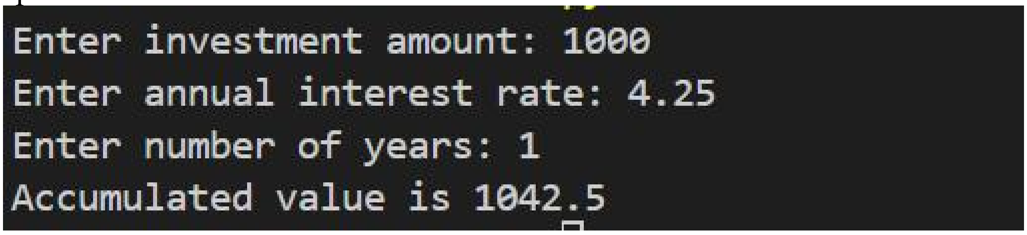
Solution:

Code:



```
i = float(input("Enter investment amount: "))
r = float(input("Enter annual interest rate: ")) / 100
y = float(input("Enter number of years: "))
a = round( i * (1 + r)**y, 2)
print("Accumulated value is", end=" ")
print(a)
```

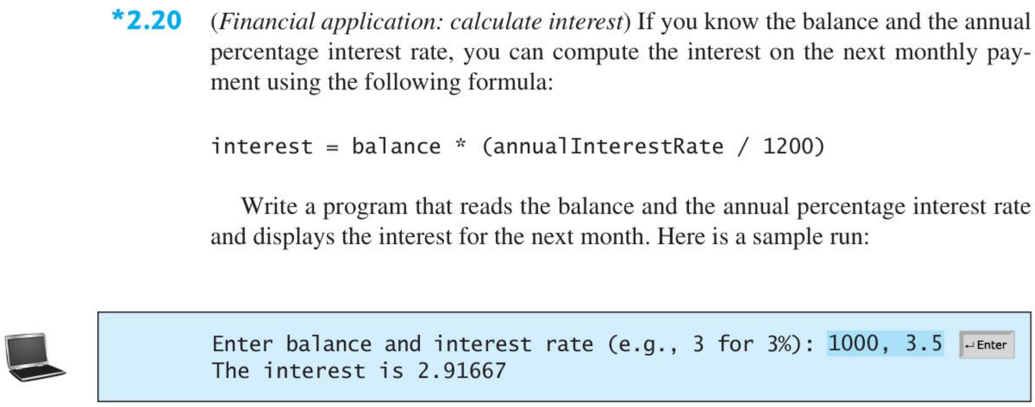
Output:



```
Enter investment amount: 1000
Enter annual interest rate: 4.25
Enter number of years: 1
Accumulated value is 1042.5
```

Figure 19.1: Output

Problem no. 20.



***2.20** (*Financial application: calculate interest*) If you know the balance and the annual percentage interest rate, you can compute the interest on the next monthly payment using the following formula:

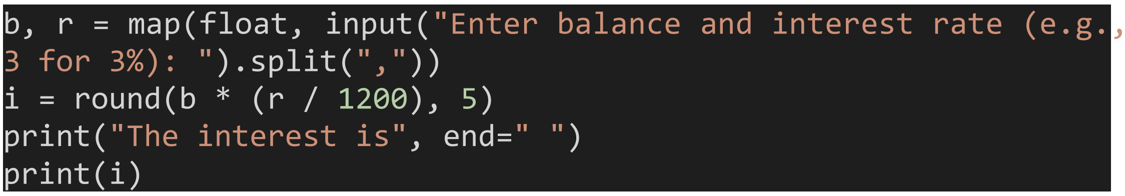
```
interest = balance * (annualInterestRate / 1200)
```

Write a program that reads the balance and the annual percentage interest rate and displays the interest for the next month. Here is a sample run:

```
Enter balance and interest rate (e.g., 3 for 3%): 1000, 3.5
The interest is 2.91667
```

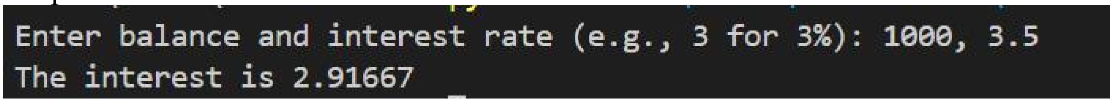
Solution:

Code:



```
b, r = map(float, input("Enter balance and interest rate (e.g.,
3 for 3%): ").split(","))
i = round(b * (r / 1200), 5)
print("The interest is", end=" ")
print(i)
```

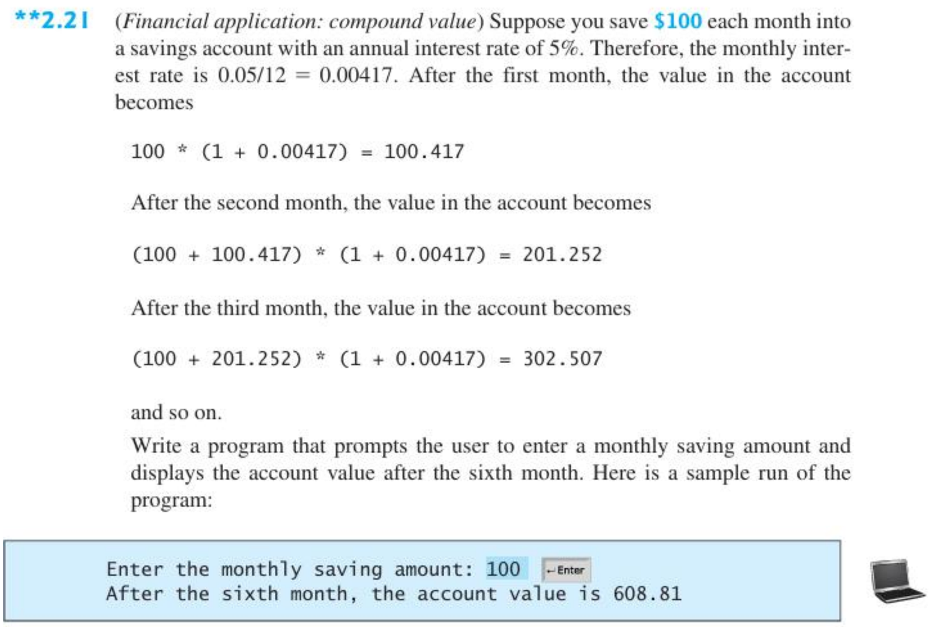
Output:



```
Enter balance and interest rate (e.g., 3 for 3%): 1000, 3.5
The interest is 2.91667
```

Figure 20.1: Output

Problem no. 21.



**2.21 (*Financial application: compound value*) Suppose you save $100 each month into a savings account with an annual interest rate of 5%. Therefore, the monthly interest rate is 0.05/12 = 0.00417. After the first month, the value in the account becomes

```
100 * (1 + 0.00417) = 100.417
```

After the second month, the value in the account becomes

```
(100 + 100.417) * (1 + 0.00417) = 201.252
```

After the third month, the value in the account becomes

```
(100 + 201.252) * (1 + 0.00417) = 302.507
```

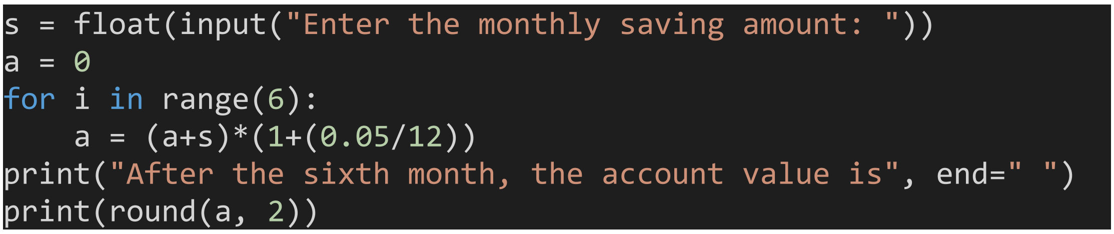
and so on.

Write a program that prompts the user to enter a monthly saving amount and displays the account value after the sixth month. Here is a sample run of the program:

```
Enter the monthly saving amount: 100 ←Enter
After the sixth month, the account value is 608.81
```

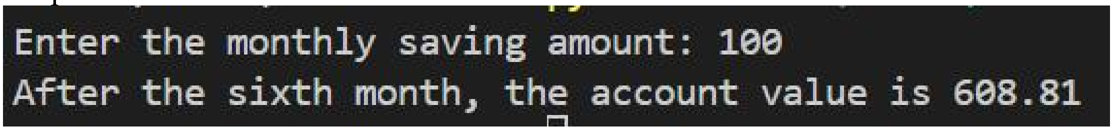
Solution:

Code:



```
s = float(input("Enter the monthly saving amount: "))
a = 0
for i in range(6):
    a = (a+s)*(1+(0.05/12))
print("After the sixth month, the account value is", end=" ")
print(round(a, 2))
```

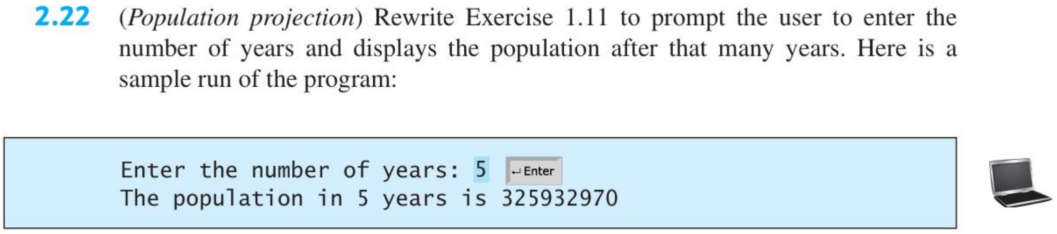
Output:



```
Enter the monthly saving amount: 100
After the sixth month, the account value is 608.81
```

Figure 21.1: Output

Problem no. 22.

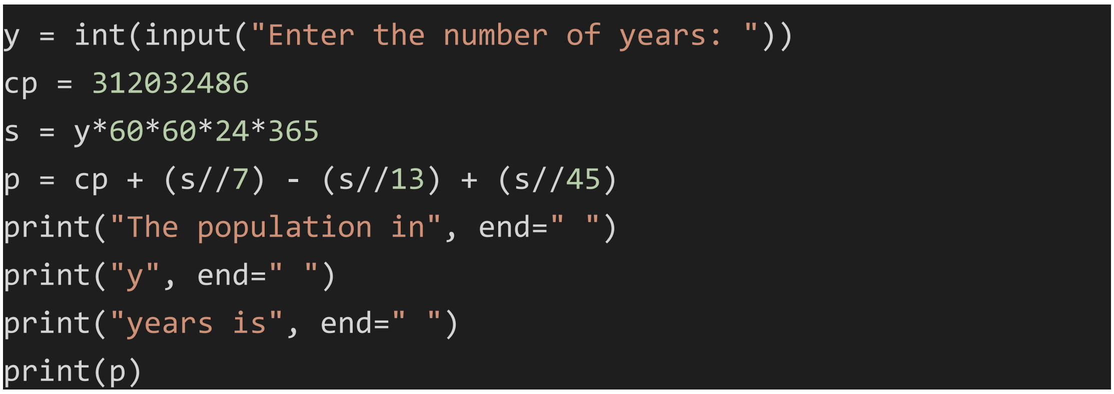


2.22 (*Population projection*) Rewrite Exercise 1.11 to prompt the user to enter the number of years and displays the population after that many years. Here is a sample run of the program:

```
Enter the number of years: 5 ↵Enter
The population in 5 years is 325932970
```

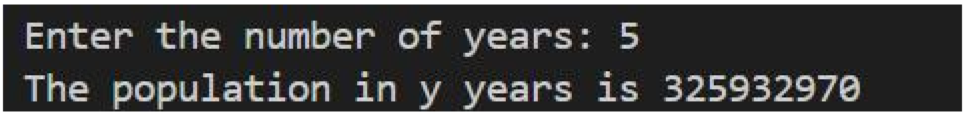
Solution:

Code:



```
y = int(input("Enter the number of years: "))
cp = 312032486
s = y*60*60*24*365
p = cp + (s//7) - (s//13) + (s//45)
print("The population in", end=" ")
print("y", end=" ")
print("years is", end=" ")
print(p)
```

Output:

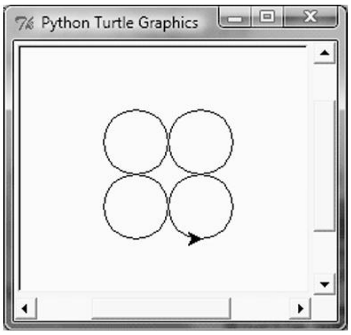


```
Enter the number of years: 5
The population in y years is 325932970
```

Figure 22.1: Output

Problem no. 23.

2.23 (*Turtle: draw four circles*) Write a program that prompts the user to enter the radius and draws four circles in the center of the screen, as shown in Figure 2.4a.



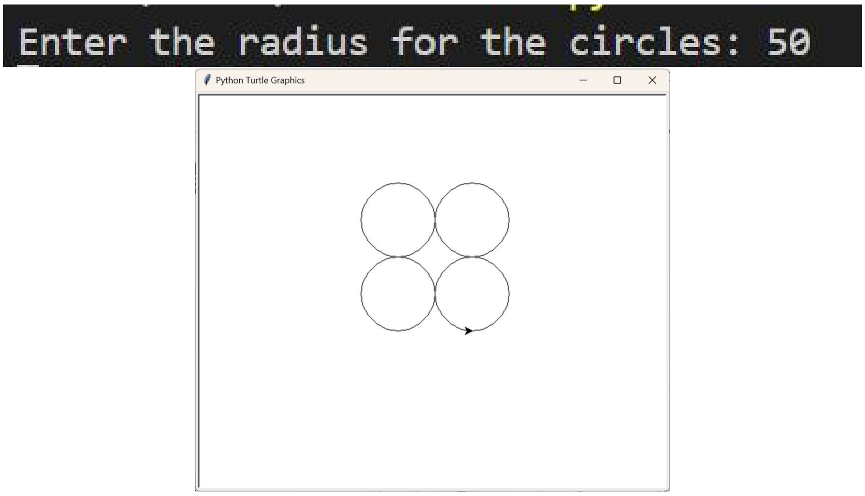
(a)

Solution:

Code:

```
import turtle
radius = float(input("Enter the radius for the circles: "))
turtle.penup()
positions = [(-radius, radius), (radius, radius), (-radius, -radius), (radius, -radius)]
for pos in positions:
    turtle.goto(pos)
    turtle.pendown()
    turtle.circle(radius)
    turtle.penup()
turtle.done()
```

Output:



```
Enter the radius for the circles: 50
```

Figure 23.1: Output