Question 1. [10 points]
Assume you executed the following:
width = 17
height = 12.0

What is the value and type of:
1. width / 2
2. width / 2.0
3. height / 3
4. 1 + 2 * 5

Answers:
1. width / 2 = 8 (integer)
2. width / 2.0 = 8.5 (float)
3. height / 3 = 4.0 (float)
4. 1 + 2 * 5 = 11 (integer)

Question 2. [10 points]
Write a program to prompt the user for hours and rate per hour to compute gross pay

Enter Hours: 35
Enter Rate: 10.75
Pay: 376.25

Rewrite your pay computation to give employee 1.5 times the hourly rate for hours worked above 40 hours.

Answers:

a. The python script below is a possible implementation of the desired functionality

```python
#!/usr/bin/python
import os
import sys
import fileinput

# getting inputs
prompt_hours = ' Enter worked hours: '
hours = float(raw_input(prompt_hours))
prompt_rate = ' Enter hourly rate: '
rate = float(raw_input(prompt_rate))
# check values are positive
if hours <= 0.0:
    print "Sorry, worked hours needs to be positive"
    exit()
```
if rate <= 0.0:
    print "Sorry, hourly rate needs to be positive"
    exit()

#calculate gross pay
pay = hours * rate
#print pay in ####.## format
pay = round(pay,2)
paystr = str(pay)
print ('Pay:' + paystr)

b. We need only to change the section on the pay computation as shown below.

    #calculate gross pay
    if hours < 40:
        pay = hours * rate
    if hours >= 40:
        pay = 40 * rate + (hours - 40) * rate * 1.5

Question 3. [10 points]
What is the value of myVar in the python code below? Is the parenthesis really necessary? Why or why not? What is the type of myVar?

    myVar = (4 * 7.5) / 5 ** 1 - 1

Answer:
The order of operations follows PEMDAS, so in the version of this expression without the parenthesis the order of the operation would be as follows:

Exponentiation: 5**1 = 5; Multiplication: 4 * 7.5=30.0; Division: 30.0/5 = 6.0; Subtraction: 6.0 – 1 – 5.0. The result would be 5.00 (float).

In the expression with parenthesis we have the following order:

Exponentiation: 5**1 = 5; Multiplication: 4 * 7.5=30.0; Division: 30.0/5 = 6.0; Subtraction: 6.0 – 1 – 5.0. The result would be 5.00 (float).

For this example, the parentheses are useless as the multiplication would be the first operation after the exponentiation irrespective of the parenthesis.

The type of MyVar is float

Question 4. [30 points]
Write a program which prompts the user for a Celsius temperature, convert the temperature to Fahrenheit, and print out the converted temperature.

Rewrite your pay program using try and except so that your program handles non-numeric input gracefully by printing a message and exiting the program.

Answer:
The python script below is a possible implementation of the desired functionality

```python
#!/usr/bin/python
import os
```

```python
if rate <= 0.0:
    print "Sorry, hourly rate needs to be positive"
    exit()

#calculate gross pay
pay = hours * rate
#print pay in ####.## format
pay = round(pay,2)
paystr = str(pay)
print ('Pay:' + paystr)

b. We need only to change the section on the pay computation as shown below.

    #calculate gross pay
    if hours < 40:
        pay = hours * rate
    if hours >= 40:
        pay = 40 * rate + (hours - 40) * rate * 1.5

Question 3. [10 points]
What is the value of myVar in the python code below? Is the parenthesis really necessary? Why or why not? What is the type of myVar?

    myVar = (4 * 7.5) / 5 ** 1 - 1

Answer:
The order of operations follows PEMDAS, so in the version of this expression without the parenthesis the order of the operation would be as follows:

Exponentiation: 5**1 = 5; Multiplication: 4 * 7.5=30.0; Division: 30.0/5 = 6.0; Subtraction: 6.0 – 1 – 5.0. The result would be 5.00 (float).

In the expression with parenthesis we have the following order:

Exponentiation: 5**1 = 5; Multiplication: 4 * 7.5=30.0; Division: 30.0/5 = 6.0; Subtraction: 6.0 – 1 – 5.0. The result would be 5.00 (float).

For this example, the parentheses are useless as the multiplication would be the first operation after the exponentiation irrespective of the parenthesis.

The type of MyVar is float

Question 4. [30 points]
Write a program which prompts the user for a Celsius temperature, convert the temperature to Fahrenheit, and print out the converted temperature.

Rewrite your pay program using try and except so that your program handles non-numeric input gracefully by printing a message and exiting the program.

Answer:
The python script below is a possible implementation of the desired functionality

```python
#!/usr/bin/python
import os
```
import sys
import fileinput

while True:
    try:
        # convert it into Celsius
        prompt_fah = "Please enter temperature in Fahrenheit degree: ":
        fah = raw_input(prompt_fah)
        fah = float(fah)
        cel = round((fah - 32) / 1.8, 2)

        # print result
        str_fah = str(fah)
        str_cel = str(cel)
        print (str_fah + " Farenheit is " + str_cel + " degree in Celsius")
        break
    except:
        print 'Please enter a valid number. Try again. ' 

**Question 5. [40 points]**
Write a program to prompt for a score between 0.0 and 1.0. If the score is out of range, print an error message. If the score is between 0.0 and 1.0, print a grade using the following table:

**Answer:**
The python script below is a possible implementation of the desired functionality

```python
#!/usr/bin/python
import os
import sys
import fileinput

prompt_score = 'Enter score: '
score = raw_input(prompt_score)
score = float(score)

if score < 0.0 or score > 1.0:
    print "Input a valid score between 0.0 and 1.0."
    exit()

if score >= 0.0 and score < 0.6:
    grade = 'F'
elif score >= 0.6 and score < 0.7:
    grade = 'D'
elif score >= 0.7 and score < 0.8:
    grade = 'C'
elif score >= 0.8 and score < 0.9:
    grade = 'B'
else:
    grade = 'A'
print grade
```