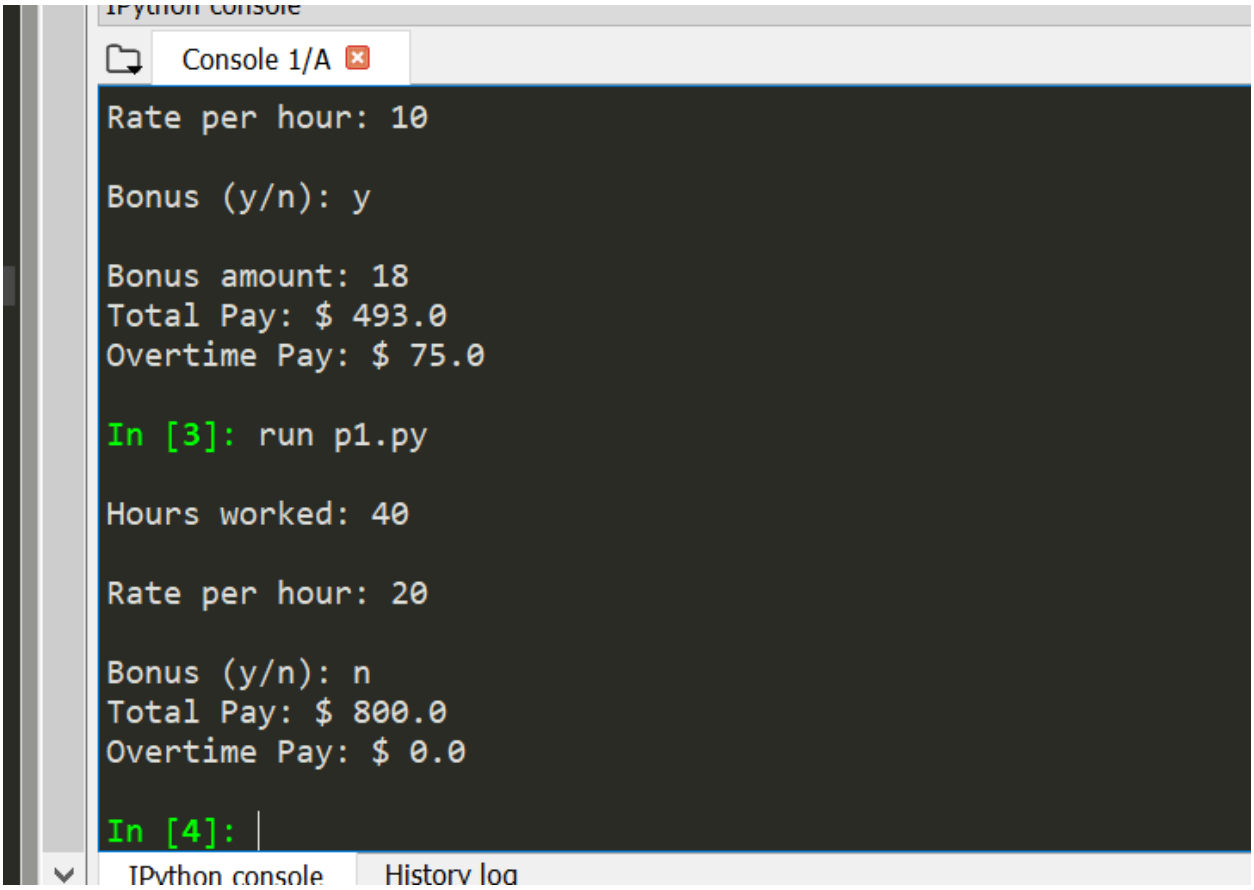
Homework 1

P1

```python
# Problem 1
""" Input: hours, wage, bonus: (y or n). Output: totalpay, overtimepay"""
# String Constants
hours_prompt = 'Hours worked: '
wage_prompt  = 'Rate per hour: '
has_bonus    = 'Bonus (y/n): '
bonus_prompt = 'Bonus amount: '

def get_float(message):
    """Returns keyboard input as float"""
    return float(input(message))

def overtime(h, r):
    """Returns the amount of overtime pay earned"""
    return (h - 40) * r * 1.5

def reg_pay(h,r):
    """Returns pay amount before overtime"""
    if h <= 40:
        return h * r
    else:
        return 40 * r

def pay(h, r):
    """Calculates total pay without regard to bonus"""
    if h <= 40:
        return reg_pay(h,r)
    else:
        return reg_pay(h,r) + overtime(h, r)

    return print("Total pay: $", pay + bonus, "overtime pay: $", ovr)

def main():
    hours_worked  = get_float(hours_prompt)
    rate_per_hour = get_float(wage_prompt)
    yes_no = input(has_bonus)
    if yes_no.lower() == 'y':
        bonus = get_float(bonus_prompt)
        print("Total Pay: $", pay(hours_worked, rate_per_hour) + bonus)
        print("Overtime Pay: $", overtime(hours_worked, rate_per_hour))
```

```
    else:
        print("Total Pay: $", pay(hours_worked, rate_per_hour))
        print("Overtime Pay: $", overtime(hours_worked, rate_per_hour))
main()
```

P1.screenshot



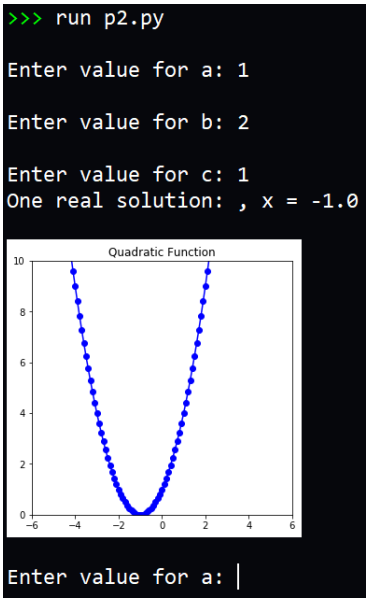
P2

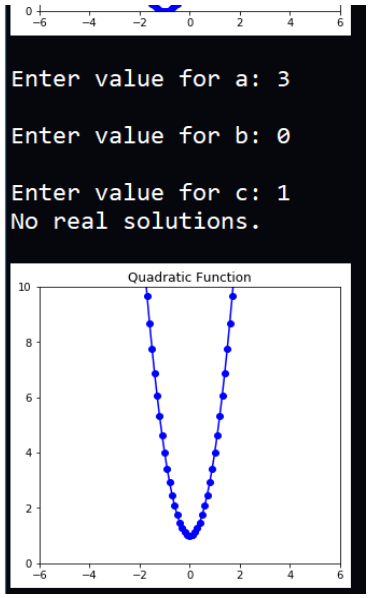
```
# -*- coding: utf-8 -*-
import pylab
import math

''' Adjust Graph-Image Size '''
from pylab import rcParams
rcParams['figure.figsize'] = 5, 5


def get_val(val):
```

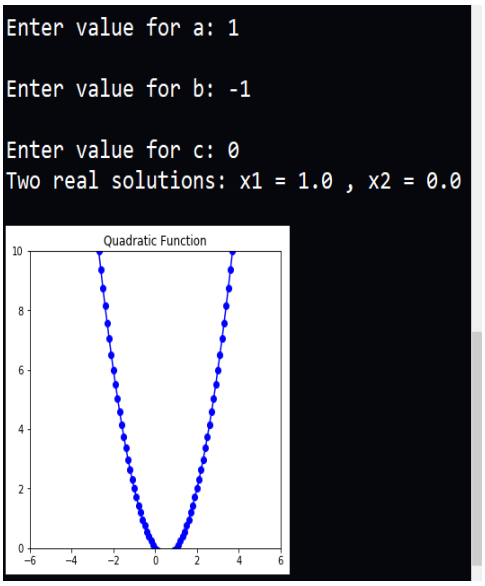
```
    return float(input('Enter value for ' + val + ': '))

def discriminant(a,b,c):
    return b**2 - (4 * a * c)

def plus_quadratic(a,b,d):
    return (-b + math.sqrt(d)) / (2 * a)

def minus_quadratic(a,b,d):
    return (-b - math.sqrt(d)) / (2 * a)

def one_solution(a, b):
    return -b / (2 * a)

def output(a,b,d):
    if d < 0:
        print('No real solutions.')
    if d == 0:
        print('One real solution: , x =', one_solution(a,b))
    if d > 0:
        print('Two real solutions: x1 =', plus_quadratic(a,b,d)\
        ,', x2 =', minus_quadratic(a,b,d))

def main():
    while(True):
        a = get_val('a')
        b = get_val('b')
        c = get_val('c')
        # important for number of solutions and their values
        d = discriminant(a,b,c)
        # prints number of solutions and their values
        output(a,b,d)

        # Lists will hold the coordinates
        xs = []
        ys = []

        # Outer bounds
        x_0 = -5.0
        x_1 =  5.0

        # Initial point and Number of points
        x = x_0
        n = 100
```

```
        # Step value
        dx = (x_1 - x_0) / n

        # Load the lists
        while x <= x_1 :
            xs.append(x)
            y = a*x**2 + b*x + c
            ys.append(y)
            x += dx

        pylab.title('Quadratic Function')
        pylab.plot(xs,ys,'-bo')
        pylab.xlim(-6, 6)
        pylab.ylim(0, 10)
        pylab.show()
main()
```

P2 – screenshot







P3

Variables:

- **amount_str**
  - string
  - user enters a dollar amount
- **amount**
  - float
  - **amount_str** is now a floating point number
- **quarters**, **dimes**, **pennies**
  - int
  - will combine so that the sum of their values will equal **amount**
- **total_coins**
  - int
  - Sum of all denominations of coins used to represent **amount.**

Convert-to-change (Pseudocode) :

1. Divide
2. Step 2
   a. Step 2a

Compute-Change (Pseudocode):

1. Take input as a string. Save the original copy
2. Create a new variable to convert cast it to a float.
3. Multiply by 100 to move the decimal to the end.
4. Result to an integer.
5. Quarters = cash / 25
6. Cash = cash – (quarters * 25)
7. Dimes = Cash / 10
8. Cash = cash – (dimes * 10)
9. Pennies = cash

P3 Code

```
'''Turn dollars into coins'''

# ctrl-c ctrl-z don't affect the console for me
while True:
    try:
        cash_str = input('Enter Amount: ')
        cash = float(cash_str) * 100
```

```
        cash_int = int(cash)
        q = 25 # quarter
        d = 10 # dime
        p = 1  # penny
        quarters = cash_int // q
        cash_int -= (quarters * q)
        dimes = cash_int // d
        cash_int -= (dimes * d)
        pennies = cash_int // p
        cash_int -= (pennies * p)
        print('Quarters: ', quarters)
        print('Dimes: ', dimes)
        print('Pennies: ', pennies)
        print('Total: ',(quarters+dimes+pennies))
        print('Total money in coins:', cash_str)

    except ValueError:
        print("Invalid Input: Terminating...")
        break
```

P3 Screenshot

IPython console

Console 1/A

```
>>> run p3.py

Enter Amount: 99.99
Quarters:  399
Dimes:  2
Pennies:  4
Total:  405
Total money in coins: 99.99

Enter Amount: 10
Quarters:  40
Dimes:  0
Pennies:  0
Total:  40
Total money in coins: 10

Enter Amount: .24
Quarters:  0
Dimes:  2
Pennies:  4
Total:  6
Total money in coins: .24

Enter Amount: 5.25
Quarters:  21
Dimes:  0
Pennies:  0
Total:  21
Total money in coins: 5.25

Enter Amount: 2.45
Quarters:  9
Dimes:  2
Pennies:  0
Total:  11
Total money in coins: 2.45

Enter Amount: |
```