CPSC 1301, Computer Science I Lab Assignment

Lab 04b Solutions

Problem 1

Write a Python program that accepts an integer as a command line argument as a limit and prints powers of 2 to that limit.

```
#-----------------------------------------------------------------------
# powersoftwo.py
#-----------------------------------------------------------------------

import sys

# Accept positive integer n as a command-line argument. Write to
# standard output a table showing the first n powers of two.

n = int(sys.argv[1])

power = 1
i = 0
while i <= n:
    # Write the ith power of 2.
    print(str(i) + ' ' + str(power))
    power = 2 * power
    i = i + 1
```

Problem 2

Write a Python program that accepts an integer n as a command line argument and prints a $n \times n$ table. Print an asterisk (*) in each row and column where either the row or column is divisible by the other.

```
#-----------------------------------------------------------------------
# divisorpattern.py
#-----------------------------------------------------------------------

import sys

# Accept integer command-line argument n. Write to standard output
# an n-by-n table with an asterisk in row i and column j if either
# i divides j or j divides i.

n = int(sys.argv[1])

for i in range(1, n+1):
    print(i, '', end = '')
print()

for i in range(1, n+1):
    # Write the ith line.
    for j in range(1, n+1):
        # Write the jth entry in the ith line.
        if (i % j == 0) or (j % i == 0):
            print('* ', end = '')
        else:
            print('  ', end = '')
    print(i)
```

Problem 3

In mathematics, the n-th harmonic number is the sum of the reciprocals of the first n natural numbers. Write a Python program that accepts an integer as a command line argument and prints the *nth* harmonic number.

$$H_n = \sum_{i=1}^{i=n} \frac{1}{i} \tag{1}$$

```
#-----------------------------------------------------------------
# harmonic.py
#-----------------------------------------------------------------

import sys

# Accept integer n as a command-line argument. Write to standard
# output the value of the nth harmonic number.

n = int(sys.argv[1])

total = 0.0
for i in range(1, n+1):
    # Add the ith term to the sum
    total += 1.0 / i

print(total)
```

Problem 4

Write a Python program that accepts an integer *num* as a command line argument and computes the square root of n.

```
#-----------------------------------------------------------------
# sqrt.py
#-----------------------------------------------------------------

import sys

# Accept float c as a command-line argument. Write to standard
# output the square root of c to 15 decimal places of accuracy.
# Use Newton's method.

EPSILON = 1e-15

num = float(sys.argv[1])
t = num
while abs(t - num/t) > (EPSILON * t):
    #print("t = %f, num = %f, num/t = %f" % (t, num, num/t)) #uncomment to see what's going
        on
    # Replace t by the average of t and num/t.
    t = (num/t + t) / 2.0
print(t)
```

Problem 5

Write a Python program that accepts a positive integer as a command line argument and prints the equivalent binary number.

```
#-----------------------------------------------------------------
# binary.py
```

```
3  #-------------------------------------------------------------------
4
5  import sys
6
7  # Accept integer n as a command-line argument. Write the binary
8  # representation of n to standard output.
9
10 # Limitation: Does not handle negative integers.
11
12 n = int(sys.argv[1])
13 bin = ''
14
15 # Compute v as the largest power of 2 <= n.
16 v = 1
17 while v <= n//2:
18     v *= 2
19
20 # Cast out powers of 2 in decreasing order.
21 while v > 0:
22     if n < v:
23         bin = bin + '0'
24     else:
25         bin = bin + '1'
26         n -= v
27     v //= 2
28
29 print(bin)
```

Problem 6

Write a Python program that computes Euler's number, e.

```
1  #-----------------------------
2  # e.py
3  #-----------------------------
4
5  # Write a program that computes Euler's number e
6  # The sum of 1/factorial(n) to 20 places
7
8  print("Computing e")
9
10 e = 1.0
11 frac = 0.0
12 fac = 1
13 for i in range(1, 21):
14     fac = fac * i
15     frac =  1.0 / fac
16     e = e + frac
17     #print("i is %d, frac is %f, fac is %d, and e is %f" % (i, frac, fac, e))
18
19 print("e is", e)
```