

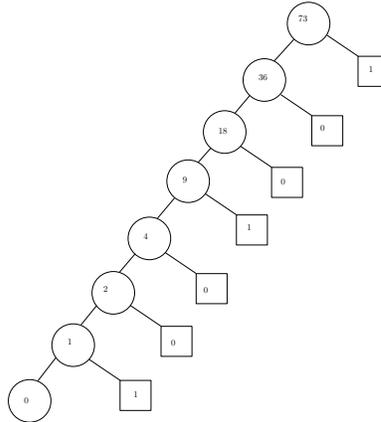
22C:16 Homework 1

Due via ICON on Wednesday, Feb 2nd, 4:59 pm

Problems 1 and 2 are based on this program.

```
number = int(raw_input("Enter a number: "))
while number > 0:
    print number%2
    number = number/2
```

1. The execution of this program can be compactly represented by an “execution tree” (as discussed in the lecture on Friday, Jan 21st). For example, the execution of the program on input 73 is shown by:



The numbers shown in the circles represent the values of the variable `number` and the numbers shown in the squares are the bits that are output.

- (a) Draw the “execution tree” for input 86.
 - (a) Draw the “execution tree” for input 141.
2. Now type this program into a Python IDE and save it as `intToBinary1.py`. Instructions on how to use the Wing IDE on DIVMS machines is posted on the course website). The following questions require that you run (execute) this program, sometimes after small modifications.
 - (a) Execute `intToBinary1.py` to find out what the binary equivalent of 1783 is. What is it?
 - (b) Execute `intToBinary1.py` and when you are prompted "Enter a number", type -50. What happens? Explain in a sentence why your program behaves in this manner.
 - (c) Execute `intToBinary1.py` and when you are prompted "Enter a number", type `hello`. What happens? Explain in a sentence why your program behaves in this manner.

3. This is based on the following improved version of `intToBinary1.py`. Type up this program in a Python IDE, save it as `intToBinary2.py` and answer the following questions by playing with this program.

```
number = int(raw_input("Enter a number: "))
binary = ""
while number > 0:
    binary = str(number%2) + binary
    number = number/2
print "The binary equivalent is", binary
```

- (a) What output do you get when you change Line 4 to

```
binary = binary + str(number%2)
```

and run the program with input 73? Is the output the binary equivalent of 73? Explain in two sentences what this change is causing the program to do.

- (b) Change Line 4 to

```
binary = number%2 + binary
```

and execute the program. What happens? Explain in one sentence why your program behaves in this manner.

- (c) Replace all the occurrences of 2 in the program by 3. Now run the program with input 73. What output do you get? What connection does this output have to the input?

4. Write a Python program that reads as input a positive integer and produces as output the *reverse* of the input integer. For example,

Input	Output
73	37
1024	4201
100	001

Your program should be very similar to `intToBinary2.py`. The basic idea of this program is to extract the decimal digits of the input, one digit at a time, and then put them together in reverse order.
