Lab 7b – Long Integers

When implementing the RSA public key cipher, we need to have the ability to work with extremely long integers (hundreds of digits in length). The problem with most programming languages is that the size of an integer is fixed to 32 or 64 bits depending on the CPU architecture.

In this lab we will create an ADT called “DigitChain” which represents very long integers as a chain of digits (i.e. a linked list filled with values between 0 and 9).

Consider the number 52669. We are going to store this in the following format:

Diagram

Description automatically generated

Why store it backwards? It will make the arithmetic operations a lot easier (remember that when we do calculations, we work with the lower significant digits first).

Create the DigitChain ADT with the following functions:

* convert\_int(x): This will convert x into a linked chain of digits. Remember to store the chain in reverse.
* convert\_str(x): This will convert the string x into a linked chain of digits. Remember to store the chain in reverse.
* \_\_str\_\_(): Return a string representing the number stored in the chain.

Write a function called “add\_digitchain(c1, c2)” that will add the two DigitChain values c1 and c2 together and return a new DigitChain representing the sum c1+c2.