Discrete Structures (CS1005)

Name: Muhammad Faizan Shabir

Roll Number: 22L-6552

Section: 3H

Date: 27th November, 2023

Assignment Number: 4 (Four)

Q1) Suppose that performing arithemetic with integers less than 100 on a certain processor is much quicker than doing arithemetic with larger integers. You can restrict almost all our computations to integers less than 100 if we represent integers using their remainders modulo pairwise ~~and~~ relatively prime integers less than 100. For example, You can use the moduli of 99, 98, 97, and 95. (These integers are relatively prime pairwise, because no two have a common factor greater than 1.) Write a new addition routine/function (in the programming language of your choice), that takes as input two large numbers, breaks those numbers in 4-tuples, adds the 4-tuples componentwise and reduce each component with respect to the appropriate modulus and converts back the resultant 4-tuple into corresponding integer (using the chinese remainder theorem). You may assume that the result does not exceed the product of your chosen moduli.

```
// Extended Euclidean Algorithm
int extendedGCD(int a, int b, int &x, int &y)
{
    if (a == 0)
    {
        x = 0;
        y = 1;
        return b;
    }

    int x1, y1;
    int gcd = extendedGCD(b%a, a, x1, y1);
    x = y1 - (b/a) * x1;
    y = x1;
    return gcd;
}
```

```
// Modular Inverse Function
int modInverse(int a, int m)
{
    int x, y;
    int gcd = extendedGCD(a, m, x, y);
    if (gcd != 1)
    {
        throw runtime_error("Modular inverse does not exist");
    }

    return ((x % m) + m) % m;
}
```

```
// addition routine/function
int addition(int num1, int num2)
{
      Const int m1 = 99;
      const int m2 = 98;
      Const int m3 = 97;
      Const int m4 = 95;

      int tupleNum1[4];
      int tupleNum2[4];

      tupleNum1[0] = num1 % m1;
      tupleNum1[1] = num1 % m2;
      tupleNum1[2] = num1 % m3;
      tupleNum1[3] = num1 % m4;

      tupleNum2[0] = num2 % m1;
      tupleNum2[1] = num2 % m2;
      tupleNum2[2] = num2 % m3;
      tupleNum2[3] = num2 % m4;

      int a1 = (tupleNum1[0] + tupleNum2[0]) % m1;
      int a2 = (tupleNum1[1] + tupleNum2[1]) % m2;
      int a3 = (tupleNum1[2] + tupleNum2[2]) % m3;
      int a4 = (tupleNum1[3] + tupleNum2[3]) % m4;

      int m = m1 * m2 * m3 * m4;
      int M1 = m/m1;
      int M2 = m/m2;
      int M3 = m/m3;
      int M4 = m/m4;
```

```
int result = (a1 * M1 * modInversie(M1, m1)
            + a2 * M2 * modInverse(M2, m2)
            + a3 * M3 * modInverse(M3, m3)
            + a4 * M4 * modInverse(M4, m4))
result = result % m;
return result;
}
```

Q2) Consider this problem, which was originally posed by Leonardo Pisano, also known as Fibonacci, in the thirteenth century in his book Liber abaci. A young pair of rabbits (one of each sex) is placed on an island. A pair of rabbits does not breed until they are 2 months old. After they are 2 months old, each pair of rabbits produces another pair each month, as shown in Figure 1. Find a recurrence relation for the number of pairs of rabbits on the island after n months, assuming that no rabbits ever die.

Denote by f_n the number of pairs of rabbits after n months. We will show that f_n, $n = 1, 2, 3 ---$ are the terms of the Fibonacci sequence.

The rabbit population can be modeled using a recurrence relation. At the end of the first month, the number of pairs of rabbits on the island is $f_1 = 1$. Because this pair does not breed during the second month, $f_2 = 1$ also. To find the number of pairs after n months, add the number on the island the month, f_{n-1}, and the number of newborn pairs, which equals f_{n-2}, because each newborn pair comes from a pair at least 2 months old.

Consequently, the sequence $\{f_n\}$ satisfies the recurrence relation

$$f_n = f_{n-1} + f_{n-2}$$

for $n \geq 3$ together with the initial conditions $f_1 = 1$ and $f_2 = 1$. Because this recurrence relation and the initial conditions uniquely determine this sequence, the number of pairs of rabbits on the island after n months is given by the nth Fibonacci number.

Q3) A popular puzzle of the late nineteenth century invented by the French mathematician Edouard Lucas, called the Tower of Hanoi, consists of three pegs mounted on a board together with disks of different sizes. Initially these disks are placed on the first peg in order of size, with the largest on the bottom (as shown in Figure 2). The rules of the puzzle allow disks to be moved one at a time from one peg to another as long as a disk is never placed on top of a smaller disk. The goal of the puzzle is to have all the disks on the second peg in order of size, with the largest on the bottom. Let H_n denote the number of moves needed to solve the Tower of Hanoi problem with n disks. Set up a recurrence relation for the sequence $\{H_n\}$.

Begin with n disks on peg 1. We can transfer the top $(n-1)$ disks, following the rules of the puzzle, to peg 3 using H_{n-1} moves. We keep the largest disk fixed during these moves. Then, we use one move to transfer the largest disk to the second peg. We can transfer the $(n-1)$ disks on peg 3 to peg 2 using H_{n-1} additional moves, placing them on top of the largest disk, which always stays fixed on the bottom of peg 2.

Moreover, it is easy to see that the puzzle cannot be solved using fewer steps.

This shows that

$$H_n = 2H_{n-1} + 1$$

The initial condition is $H_1 = 1$, because one disk can be transferred from peg 1 to peg 2, according to the rules of the puzzle, in one move.

We can use an iterative approach to solve this recurrence relation. Note that

$$H_n = 2H_{n-1} + 1$$

$$= 2(2H_{n-2} + 1) + 1 = 2^2 H_{n-2} + 2 + 1$$

$$= 2^2(2H_{n-3} + 1) + 2 + 1 = 2^3 H_{n-3} + 2^2 + 2 + 1$$

$$\vdots$$

$$= 2^{n-1} H_1 + 2^{n-2} + 2^{n-3} + \cdots + 2 + 1$$

Since $H_1 = 1$

$$\Rightarrow 2^{n-1} + 2^{n-2} + \cdots + 2 + 1$$

$$= 2^n - 1$$

We have used the recurrence relation repeatedly to express H_n in terms of previous terms of the sequence. In the next to last equality, the initial condition $H_1 = 1$ has been used. The last equality is based on the formula for the sum of the terms of a geometric series.

The iterative approach has produced the solution to the recurrence relation.

$H_n = 2H_{n-1} + 1$ with the initial condition $H_1 = 1$.

Q 4) Find a recurrence relation and give initial conditions for the number of bit strings of length n that do not have two consecutive 0s. How many such ~~bits~~ bit strings are there of length five?

Let a_n denote the number of bit strings of length n that do not have two consecutive 0s. To obtain a recurrence relation for $\{a_n\}$, note that by the sum rule, the number of ~~bits~~ bit strings of length n that do not have two consecutive 0s equals the number of such bit strings ending with a 0 plus the number of such bit strings ending with a 1. We will assume that $n \geq 3$, so that the bit string has at least ~~the~~ three bits.

The bit strings of length n ending with 1 that do not have two consecutive 0s are precisely the bit strings of length $(n-1)$ with no two consecutive 0s with a 1 added at the end. Consequently, there are a_{n-1} such bit strings.

Bit strings of length n ending with a 0 that do not have two consecutive 0s must have 1 as their $(n-1)$st bit; otherwise they would end with a pair of 0s. It follows that the bit strings of length n ending with a 0 that have no two consecutive 0s are precisely the bit strings of length $(n-2)$ with no two consecutive 0s with 10 added at the end. Consequently, there are a_{n-2} such bit strings.

We conclude that

$$a_n = a_{n-1} + a_{n-2} \text{ for } n \geq 3$$

The initial conditions are $a_1 = 2$, because both bit strings of length one, 0 and 1 do not have consecutive 0s, and $a_2 = 3$, because the valid bit strings of length two 01, 10 and 11.

To obtain a_5, we use the recurrence relation three times to find that

$$a_3 = a_2 + a_1 = 3 + 2 = 5$$

$$a_4 = a_3 + a_2 = 5 + 3 = 8$$

$$a_5 = a_4 + a_3 = 8 + 5 = 13.$$

Q 5) A deposit of \$ 100, 000 is made to an investment fund at the beginning of a year. On the last day of each year two dividends are awarded. The first dividend is 20% of the amount in the account during that year. The second dividend is 45% of the amount in the account in the previous year.

a) Find a recurrence relation for $\{P_n\}$, where P_n is the amount in the account at the end of n years if no money is ever withdrawn.

Amount = Original Amount + 20% of Original Amount + 45% of Amount in the previous year.

$$\Rightarrow P_n = P_{n-1} + 0.2 P_{n-1} + 0.45 P_{n-2}$$

$$P_n = 1.2 P_{n-1} + 0.45 P_{n-2} \text{ for } n \geq 2$$

Initially, a deposit of \$100,000 is made

$$\Rightarrow P_0 = 100,000$$

After the first year, we only recieve the first dividend of 20%.

$$\Rightarrow P_1 = 1.2 P_0 = 1.2(100000) = 120000$$

So

$$P_n = 1.2 P_{n-1} + 0.45 P_{n-2} \text{ for } n \geq 2$$

with the initial conditions:

$$P_0 = 100000 \text{ and } P_1 = 120000$$

b) How much is in the account after n years if no money has been withdrawn?

Let $P_n = r^n = 1.2 r^{n-1} + 0.45 r^{n-2}$ be a solution.

The characteristic equation is:

$$r^2 - 1.2r - 0.45 = 0$$

$$(r - 1.5)(r + 0.3) = 0$$

$$r - 1.5 = 0 \quad \text{or} \quad r + 0.3 = 0$$

$$r = 1.5 = \frac{3}{2} \quad \text{or} \quad r = -0.3 = -\frac{3}{10}$$

The solution is of the form:

$$P_n = \alpha_1 r_1^n + \alpha_2 r_2^n$$

$$P_n = \alpha_1 \left(\frac{3}{2}\right)^n + \alpha_2 \left(-\frac{3}{10}\right)^n$$

$$P_0 = 100000 = \alpha_1 + \alpha_2$$

$$P_1 = 120000 = \frac{3}{2}\alpha_1 - \frac{3}{10}\alpha_2$$

Multiply the first equation by $\frac{3}{2}$

$$150000 = \frac{3}{2}\alpha_1 + \frac{3}{2}\alpha_2$$

$$120000 = \frac{3}{2}\alpha_1 - \frac{3}{10}\alpha_2$$

Subtract the previous two equations

$$30000 = \left(\frac{3}{2} + \frac{3}{10}\right)\alpha_2$$

$$30000 = \frac{9}{5}\alpha_2$$

$$\boxed{\alpha_2 = \frac{50000}{3}}$$

$$100000 = \alpha_1 + \alpha_2$$

$$100000 = \alpha_1 + \frac{50000}{3}$$

$$\boxed{\alpha_1 = \frac{250000}{3}}$$

Thus the solution of the recurrence relation is:

$$P_n = \frac{250000}{3}\left(\frac{3}{2}\right)^n + \frac{50000}{3}\left(\frac{-3}{10}\right)^n$$

Q 6) A model for the number of lobsters caught per year is based on the assumption that the number of lobsters caught in a year is the average of the number caught in the two previous years.

a) Find a recurrence relation for $\{L_n\}$, where L_n is the number of lobsters caught in year n, under the assumption for this model.

$$\text{Average of two numbers} = \frac{\text{Sum of two numbers}}{2}$$

$$\Rightarrow L_n = \frac{L_{n-1} + L_{n-2}}{2}$$

$$L_n = \frac{1}{2} L_{n-1} + \frac{1}{2} L_{n-2}$$

b) Find L_n if 100,000 lobsters were caught in year 1 and 300,000 were caught in year 2.

Let $L_n = r^n = \frac{1}{2} r^{n-1} + \frac{1}{2} r^{n-2}$

be a solution.

The characteristic equation is:

$$r^2 - \frac{1}{2} r - \frac{1}{2} = 0$$

$$r = \frac{\frac{1}{2} \pm \sqrt{\left(-\frac{1}{2}\right)^2 - 4(1)\left(-\frac{1}{2}\right)}}{2}$$

$$r = \frac{\frac{1}{2} \pm \sqrt{9/4}}{2}$$

$$r = \frac{\frac{1}{2} \pm (3/2)}{2}$$

$$r = \frac{1}{4} \pm \frac{3}{4}$$

$$\Rightarrow \quad r = 1 \quad \text{or} \quad r = -\frac{1}{2}$$

The solution is of the form:

$$L_n = \alpha_1 r_1^n + \alpha_2 r_2^n$$

$$L_n = \alpha_1 (1)^n + \alpha_2 \left(\frac{-1}{2}\right)^n$$

$$100000 = L_1 = \alpha_1 - \frac{1}{2}\alpha_2$$

$$300000 = L_2 = \alpha_2 + \frac{1}{4}\alpha_2$$

Subtract the previous two equations

$$200000 = \left(\frac{1}{4} + \frac{1}{2}\right)\alpha_2$$

$$200000 = \frac{3}{4}\alpha_2$$

$$\boxed{\alpha_2 = \frac{800,000}{3}}$$

$$100000 = \alpha_1 - \frac{1}{2}\alpha_2$$

$$100000 = \alpha_1 - \frac{1}{2}\left(\frac{800000}{3}\right)$$

$$\boxed{\alpha_1 = \frac{700000}{3}}$$

Thus the solution is:

$$L_n = \frac{700000}{3} + \frac{800000}{3}\left(-\frac{1}{2}\right)^n$$

Q 7) Find the solution to the recurrence relation

$$a_n = 6a_{n-1} - 11a_{n-2} + 6a_{n-3}$$

with the initial conditions $a_0 = 2, a_1 = 5$ and $a_2 = 15$

Let $r^n = 6r^{n-1} - 11r^{n-2} + 6r^{n-3}$ be the solution.

The characteristic equation becomes:

$$r^3 - 6r^2 + 11r - 6 = 0$$

$$(r-1)(r-2)(r-3) = 0$$

So the roots are: $r_1 = 1, r_2 = 2$ and $r_3 = 3$

The solution is of the form:

$$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n + \alpha_3 r_3^n$$

$$a_n = \alpha_1 (1)^n + \alpha_2 (2)^n + \alpha_3 (3)^n$$

$$a_0 = 2 = \alpha_1 + \alpha_2 + \alpha_3$$

$$a_1 = 5 = \alpha_1 + 2\alpha_2 + 3\alpha_3$$

$$a_2 = 15 = \alpha_1 + 4\alpha_2 + 9\alpha_3$$

Solving the last three equations simultaneously

$\alpha_1 = 1$, $\alpha_2 = -1$ and $\alpha_3 = 2$

Thus the solution is:

$a_n = (1)^n - 1(2)^n + 2(3)^n$

$a_n = 1 - (2)^n + 2(3)^n$

Q 8) Find the solution to the recurrence relation

$$a_n = -3a_{n-1} - 3a_{n-2} - a_{n-3}$$

with initial conditions $a_0 = 1$, $a_1 = -2$ and $a_2 = -1$

Let $r^n = -3r^{n-1} - 3r^{n-2} - r^{n-3}$
be the solution

The characteristic equation becomes:

$$r^3 + 3r^2 + 3r + 1 = 0$$

$$(r+1)^3 = 0$$

The roots are: $r_1 = -1$, $r_2 = -1$ and $r_3 = -1$

The solution is of the form

$$a_n = \alpha_1 r_1^n + \alpha_2 n (r_2)^n + \alpha_3 n^2 (r_3)^n$$

$$a_n = \alpha_1 (-1)^n + \alpha_2 n (-1)^n + \alpha_3 n^2 (-1)^n$$

$$a_0 = 1 = \alpha_1$$

$$a_1 = -2 = -\alpha_1 - \alpha_2 - \alpha_3$$

$$a_2 = -1 = \alpha_1 + 2\alpha_2 + 4\alpha_3$$

Solving the last three equations simultaneously

$\alpha_1 = 1$, $\alpha_2 = 3$ and $\alpha_3 = -2$

Thus the solution is:

$$a_n = (1)(-1)^n + 3n(-1)^n - 2n^2(-1)^n$$

$$a_n = (-1)^n + 3n(-1)^n - 2n^2(-1)^n$$

Q 9) Find the solution to the recurrence relation

$$a_n = a_{n-1} + n$$

with initial conditions $a_1 = 1$

This is a non-homogeneous relation.

The associated homogeneous recurrence relation is:

$$a_n = a_{n-1}$$

The characteristic ~~eq~~ equation is:

$$r - 1 = 0$$

$$\Rightarrow r = 1$$

Root is $\underline{1}$

$$a_n^{(h)} = \alpha_1 r_1^n$$

$$a_n^{(h)} = \alpha_1 (1)^n$$

$$\boxed{a_n^{(h)} = \alpha_1}$$

Finding particular solution

We have $F(n) = n(1)^n$

Since 1 is also a root of the associated homogeneous relation's characteristic equation, the form of the particular solution will be:

$$n(P_0 + P_1 n) = P_1 n^2 + P_0 n$$

Inserting the particular solution into the recurrence relation:

$$P_1 n^2 + P_0 n = P_1 (n-1)^2 + P_0 (n-1) + n$$

$$P_1 n^2 + P_0 n = P_1 (n^2 - 2n + 1) + nP_0 - P_0 + n$$

$$P_1 n^2 + P_0 n = n^2 P_1 - 2nP_1 + P_1 + nP_0 - P_0 + n$$

$$P_1 n^2 + P_0 n = P_1 n^2 - P_1 2n + P_1 + P_0 n - P_0 + n$$

$$0 = -P_1 2n + P_1 - P_0 + n$$

$$0 = -P_1 2n + n + P_1 - P_0$$

$$0 = n(-2P_1 + 1) + P_1 - P_0$$

$$\Rightarrow \quad -2P_1 + 1 = 0 \quad \text{and} \quad P_1 - P_0 = 0$$

$$P_0 = P_1$$

$$P_1 = \frac{1}{2} \qquad P_0 = \frac{1}{2}$$

Hence $a_n^{(P)} = \frac{1}{2}n^2 + \frac{1}{2}n$

$$= \frac{n(n+1)}{2}$$

Hence, all solutions of the original recurrence relation $a_n = a_{n-1} + n$ are given by:

$$a_n = a_n^{(h)} + a_n^{(P)} = \alpha_1 + \frac{n(n+1)}{2}$$

We are given $a_1 = 1$

$$a_1 = 1 = \alpha_1 + \frac{1(1+1)}{2}$$

$$1 = \alpha_1 + 1$$

$$\Rightarrow \alpha_1 = 0$$

Hence the final solution is :

$$a_n = \alpha_1 + \frac{n(n+1)}{2}$$

$$a_n = 0 + \frac{n(n+1)}{2}$$

$$a_n = \frac{n(n+1)}{2}$$