IO

1. Encoding

```
> (encodeStr "123abc!@#")
'(49 50 51 11 12 13 10 10 10)
```

Decoding

```
> (decodeLst '(49 50 51 11 12 13 10 10 10))
"123ABC   "
```

2. Calculating (2^2)(mod 4). Expected outcome is 0.

```
> (moduloCalc 2 2 4)
0
```

3. 3-digit N RSA Encoding NetID (N = 299, e = 7)

```
> (rsaEncode "diveya2" 299 7)
(14 19 32 15 35 11 50)
'(157 176 280 11 269 145 54)
```

3-digit N RSA Decoding NetID (N = 299, d = 151)

```
> (rsaDecode '(157 176 280 11 269 145 54) 299 151)
(14 19 32 15 35 11 50)
"DIVEYA2"
```

3-digit N RSA Decoding problem list (N = 299, d = 151)

```
> (rsaDecode '(280 220 63 220 93 220 176 244 157 176 145 43 63 145 23) 299 151)
(32 25 28 25 24 25 19 10 14 19 11 17 28 11 23)
"VORONOI DIAGRAM"
```

4. 2 to 3 Conversion

```
> (twoToThree '(12 34 56 7))
'(123 456 700)
```

3 to 2 Conversion

```
> (threeToTwo '(123 456 7))
'(12 34 56 70)
```

4-digit N RSA Encoding (N = 2257, e = 11)

```
> (fourRSAEncode "divey" 2257 11)
(141 932 153 500)
'(1320 604 1593 686)
```

4-digit N RSA Decoding (N = 2257, d = 1571)

```
> (fourRSADecode '(1320 604 1593 686) 2257 1571)
(141 932 153 500)
"DIVEY "
```

5. Decoding dessert (N = 2911, d = 1381)

```
> (fourRSADecode '(2377 1020 1652 1476 1500 2000 141 1208 2331) 2911 1381)
(122 815 111 410 263 114 141 924 170)
"BREAD PUDDING "
```