1. 1, 2, 3, 5, 7, 11 are the only positive non McNugget numbers since they are the only numbers that no variation of 4, 6, 9, 20 will equal to 1, 2, 3, 5, 7, 11.
2. The equation is : 4a + 6b + 9c + 20d = n
3. Prime factorization.

23 mod 15 (6+9) = 8 (4+4) (Exception)

4 + 6 = 10

4 + 9 = 13

6 + 9 = 15

4 + 20 = 24a

6 + 20 = 26

9 + 20 = 29

4, 6, 9, 20

b.

first mod n(number) by a number that can be a combination of 4, 6, 9, or 20 such as 29

26,24,15,13 or 10,then 9, 6 and 4 and see if any of them will make the number equal to 0 after using mod.

eg: for 23, 23mod15=8 , 8mod4=0 so it works

51mod 15=6, 6mod6=0

there's multiple numbers you can try to mod them first to get the final number to be 0

but if it still isn't 0 by the time you reached modding 4 and the answer is always 1,2,3

then it isn't possible to create.

31 mod 20 = 1 R 11

31 mod 9 = 3 R 4

35 mod 4 = 3

35 mod 6 = 5

35 mod 9 = 8 (3 boxes of 9 + 2 boxes of 4)

35 mod 20 = 15

15 mod 4 = 3

15 mod 6 = 3

15 mod 9 = 6 (1 box of 6 + 1 box of 20 + 1 box of 9)

151 mod 20: 7 R 11, (now run mod9, mod6, mod4 on that remainder)

11 mod 9 = 2

11 mod 4 = 3

11 mod 6 =5

151 mod 9: 16 R 7 ,

151 mod 6: 25 R 1

151 mod 4: 37 mod 3

14 (9) + 1 (4) + 1(9) + 3(6)

89 mod 20 = 4 R 9 (4 boxes of 20 + one box of 9)

31 mod 20 = 1 R 11

11 mod 9 = 1 R 2

11 mod 4 = 2 mod 3

11 mod 6 = 1 mod 5

31 mod 9 = 3 mod 4 ( 3 boxes of 9 + 1 box of 4)

Perform modular arithmetic on the number of nuggets with 4, 6, 9, to check if the remainder is divisible by any number of boxes of nuggets, or if the remainder is 0.

Also get the number of times the number gets divided by 4,6,9.

Get the remainder and continue dividing it by 4,6,9 if it is divisible.

First x=n%9, if x(remainder) != 0, y=x%6, if y !=0, k%4, if k != 0 then it isn’t part of it.

x=n%9;

if(x!=0){ y=x%6}

if(y!=0){k=y%4}

if(k!=0) return false;

—--------------------

Testing with 35 nuggets:

35 mod 20 = denominator:1 , Remainder=15 (1 box of 20 pieces)

15 mod 9 = denominator:1, Remainder:6 (1 box of 9 pieces)

6 mod 6 = denominator:1, Remainder:0 (1box of 6 pieces)

Start modding with the largest size of box: 20

If (N % 4 = 0 or N % 6 = 0 or N % 9 = 0 or N % 20 = 0)

Then the nuggets can fit into 4,6,9,20 boxes

Else if

Else

Then the nuggets cannot fit into 4,6,9,20 boxes

—-------------------------------

N = number of nuggets the customer want

If the number is :1, 2, 3, 5, 7, or 11. Tell the user **“wrong input”**

Else: Perform “denomination()”

Let a = 4

Let b = 6

Let c = 9

Let d = 20

Linear representation: (a + b + c + d)

Chosen number: 25

1. If the number can fit at the largest box (20 nuggets box) count the remaining integer and fit in lower denomination
2. If #1 fail find if the number can fit in the second largest box (9 nuggets box) then count the remaining and fir in a lower denomination
3. If #2 fail find if the number can fit in the this largest box (6 nuggets box) then count the remaining and fir in a lower denomination
4. If #3 fail, find if the number can fit in the lowest denomination box (4 nuggets box).

Using the method above:

1. The number 25 can fit at 20-box with the remaining 5 nuggets. However 5 is not a possible number to break down, then we are using #2.
2. The number 25 can fit in a 9-box with remaining 16 nuggets. We count 1 for 9-box, and since 16 is not in the “illegal list”, we will iterate again.
   1. The number 16 can fit in a 9-box with the remaining 7. However 7 is not a possible number to break down. We will continue with #3.
3. The number 16 can fit in a 6-box with the remaining 10 nuggets. We count 1 for 6-box, and since 10 is not in the “illegal list”, we will iterate again.
   1. The number 10 can fit in a 6-box and we count another 1 for 6-box. Then the remaining 4 cannot fit in a 6-box, then we will go to the #4.
4. The number 4 can fit in a 4-box. Then we solved the problem.
5. The solution is (4(1) + 6(2) + 9(1) + 20(0)) = 25

PSEUDO CODE

**#Variables**

N = input()

Illegal Numbers = (1, 2, 3, 5, 7, 11)

**#If the input is within the illegal numbers, send error**

If(N == Illegal Numbers):

print(‘Invalid Entry’)

**#If the input is valid**

Else:

While (N <= 0): **#**

**#Count 20-Box Denominations**

Modifier = True **#**

While Modifier = True **#**

Div = (N // 20) **#**

N = N - (20\*D) **#**

If(N == Illegal Numbers):

**#**

Modifier = False **#**

Else:

D += 1 #Add to D counter, and continue with loop

**#Count 9-Box Denominations**

Modifier = True **#**

While Modifier = True **#**

Div = (N // 9) **#**

N = N - (9\*D) **#**

If(N == Illegal Numbers):

**#Undo the subtraction and go to smaller denomination**

Modifier = False **#**

Else:

C += 1 **#**

**#Count 6-Box Denominations**

Modifier = True **#**

While Modifier = True **#**

Div = (N // 6) **#**

N = N - (6\*D) **#**

If(N == Illegal Numbers):

**#Undo the subtraction and go to smaller denomination**

Modifier = False **#**

Else:

B += 1 **#**

**#Count 4-Box Denominations**

Modifier = True **#**

While Modifier = True **#**

Div = (N // 4) **#**

N = N - (4\*D) **#**

If(N == Illegal Numbers):

**#Undo the subtraction and go to smaller denomination**

Modifier = False **#**

Else:

A += 1 **#**

**#The code will run until N == 0.**

**#Print the denominations of boxes**