1 // Find the number of units whose all three, four, and five consecutive digits sum are prime numbers

2 // Input: first line <- the number of queries(*q*), subsequent *q* lines <- the number of digits of input(*n*)

3 // Output: subsequent *q* lines <- the number of units that satisfy Chloe’s rules

4 **first function: is\_prime(n)**

5 n <- int(n)

6 if n = 0 or n = 1:

7 return False:

8 for i = 2 to sqrt(n) + 1

9 if n % i = 0:

10 return False

11 return True

12 temp = []

13 **second function: is\_three\_digits\_sum\_prime(n):**

14 for i = 0 to len(n-2)

15 seg <- int(n[i]) + int(n[i+1]) + int(n[i+2])

16 temp.append(seg)

17 if is\_prime(seg) = False:

18 return False

19 return True

20 **third function: is\_four\_digits\_sum\_prime(n):**

21 for i = 0 to len(n-3)

22 temp[i] = temp[i] + int(n[i+3])

23 if is\_prime(temp[i]) = False

24 return False

25 return True

26 **fourth function: is\_five\_digits\_sum\_prime(n):**

27 for i = 0 to len(n-4)

28 temp[i] = temp[i] + int(n[i+4])

29 if is\_prime(temp[i]) = False

30 return False

31 return True

32 **fifth function: free\_temp\_function(n):**

33 int \*temp = new temp[n];

34 delete [] temp;

35 **sixth function: final\_digits(n):**

36 lst = []

37 for i = 10n-1 to 10n -1

38 i = str(i)

39 x <- is\_three\_digits\_sum\_prime(i)

40 y <- is\_four\_digits\_sum\_prime(i)

41 z <- is\_five\_digits\_sum\_prime(i)

42 if x,y,z are all True:

43 lst.append(i)

44 delete temp

45 return len(lst)

46 **last function: main\_function()**

47 q <- first\_line\_input\_value(“Enter the number of queries: ”)

48 for i = 0 to q-1:

49 n <- input (“Enter the number of digits : ”)

50 return final\_digits(n)