Prolog Exercises

For each of these problems, the aim is to define a predicate using few clauses. Each of the problems is a list processing problem.

1. Write a predicate **print every second**/1 to print every other element in a list, beginning at the second element —i.e. the 2nd, 4th, 6th elements etc. It should always succeed provided it is given a list as its argument.

2. Write a predicate **deconsonant**/1 to print any element of a list that isn't a consonant (i.e. we want to print out the vowels fa,e,i,o,ug). It should always succeed provided it is given a list as its argument (we assume that the input list only contains vowels and consonants).

3. Write a predicate **head**/2 which takes a list as its first argument and returns the head of the list as its second argument. It should fail if there is no first element.

4. Write a predicate **tail**/2 which takes a list as its first argument and returns the tail of the list as its second argument. It should fail if there is no first element.

5. Write a predicate **last**/2 which takes a list as its first argument and returns the last element of the list.

6. Write a predicate **max**/3 which takes three arguments, the first two are positive integer numbers and returns in the third the max of them.

7. Write a predicate **fact**/2 which takes a natural number as first argument, and returns the factorial of the number.

8. Write a predicate **fib**/2 which takes a natural number as first argument, and returns the value of the Fibonacci sequence for that position (counting from 1).

9. Write a predicate **length2**/2 which takes a list as first argument, and returns in the second one the number of elements in the list.

10. Write a predicate **power**/3 which takes a number as first argument, a natural number as second argument and returns the first number to the power of the second.

11. Write a predicate **count_vowels**/2 which takes a list as its first argument and returns an integer number (as its second argument) which consists of the count of elements of the input list which are a vowel (we assume that the input list only contains vowels and consonants).

12. Write a predicate **vowels**/2 which takes a list as its first argument and returns a list (as its second argument) which consists of the elements of the input list which are a vowel (we assume that the input list only contains vowels and consonants).

13. Define **nple**/3 to take two inputs —an integer and a list of integers. The result is to be a list of integers formed by multiplying each integer in the list by the input integer.

14. Define **del all**/3 which takes a possible element of a list and a list as inputs and returns the list with all occurences of the named element removed. (If the named element is not in the list then the result is, of course, the whole list with no deletions)

15. Write a predicate **find every second**/2 which takes a list as its first argument and returns a list (as its second argument) which consists of every other element of the input list starting at the second element.

16. Define **sum**/2 to take a list of integers as input and return the output as their sum.

17. Write a predicate **nth**/3 which takes two inputs: the first a positive integer and the second a list. The output (initially, an uninstantiated variable) will be the element that occurs at the nth position in the list.

18. Write a predicate **dupli**/2 which takes two inputs: the first is a list, and the second will be the list with every element duplicated.

19. Write a predicate **split**/4 that splits a lis tinto two parts, the length of the first part is given.

20. Write a predicate **npli**/3 which takes three inputs: the first is a list, the second is the number of times that every elements will be copied and the third element is the new list.