

# <http://bit.ly/1V0iejh>

Function:

1. A. Write a function `double pow(int x, int y)` & test it.  
B. Calculate  $9*a^5 + 13*b^{-2} - 15*c^{10}$ . Take a, b, c as input all are integers can be negative. [don't use any library function]
2. A. Test if a given number is power of two or not `testPowTwo(int x)` & test it. [don't use any library function]  
B. Take a, b as input print all c that are power of two  $a \leq c \leq b$ .
3. A. Write a function `fibonacci(n)` returns you the n'th fibonacci. Test it.  
B. Write a function `isPrime(n)` returns if a number is prime or not.. Test it.  
C. Take k as input print first k prime fibonacci.

Array:

1. Array of size n Input/ Output.
2. Take an array of integers. Reverse the content of the array. (Don't use any additional array)  
 $[2, 3, 7] \rightarrow [7, 3, 2]$
3. Take an array of integers. Produce the multiplication array. Don't use nested loop.  
 $[1, 2, 3, 4] \rightarrow [24, 12, 8, 6]$   
 $24 = 2*3*4$   
 $12 = 1*3*4$   
 $8 = 1*2*4$
4. Take an array of integers. Number of unique element in a given array.  
 $[1, 2, 5, 2, 3, 2, 1] \rightarrow 4$

Online:

1. 0 2 5 -10 4 3 -12 5 what is the minimum initial energy needed so that no shortage occurs during the journey. +ve jump subtracts energy -ve jump adds. (A2)
2. All permutation of a given array. (B2)

Bitmask:

1. Print the bit pattern of a given number.
2. Number of 1 in given number.
3. Reverse the bit sequence.

String:

1. String input/ output.
2. String length determination.
3. Checking if a string is palindrome or not.
- 4.