# Question 1:

Write a program that calculates the first 12 terms of fibonacci sequence and stores them in an array

Question 2:  
(Modified version of exercise from chapter 10)  
To sort an array A of N elements by the bubblesort method, we  
proceed as follows:  
  
Pass 1: For j= 2 ... N, If A(j) <A(j - 1) then swap A(j) and A(j-1).  
This will place the largest element In position N.  
Pass 2: For j = 2 ... N -1, if A(j)< A(j-1) then swap A(j) and A(j -1).  
This will place the second largest element in position N -1.  
.  
.  
.  
Pass N -1: If A(2) < A(1), then swap A[2) and A(1).  
At this point the array is sorted.  
  
Demonstration :  
initial data 7 5 3 9 1  
pass 1 5 3 7 1 9  
pass 2 3 5 1 7 9  
pass 3 3 1 5 7 9  
pass 4 1 3 5 7 9  
  
Write a program segment to sort a byte array (DATA DB 7 5 3 9 1)by the bubblesort algorithm. The program should put the offset address of the array in Sl and the number of elements in variable N.  
  
Question 3:  
Implement the following sorting algorithm for a byte array  
  
  
i = N  
FOR i=N-1 times DO  
Find the position k of the largest element among A[1] ….A[i]  
Swap A[k] and A[i]  
i=i-1  
END FOR  
  
Usually algorithms are evaluated on speed by observing how many steps they took to sort a certain data set. For the data set given in question 1 , which algorithm you feel is faster, Bubble sort or select sort? (Hint : See which algorithm does less swapping)