Assignment-4

1. Write a C function to find whether a positive integer is a power of two, using a function "void is_power_of_two (int number)".
   Hint: use the shift operator, and iterate over each bit. DO NOT COMPARE WITH SUCCESIVE POWERS OF TWO.
   Input: A positive integer.
   Output : "yes" if the number is power of two, else "no". [2]

2. Write a C Program to find the Greatest Common Divisor (GCD) of two positive integers, using a function "int GCD (int a, int b)" which returns the GCD of two positive integers "a" and "b" passed as arguments. Note that the values of "a" and "b" might be arbitrary, and proper arrangements should be made to ensure that the function works for both the cases: "a >= b" and "a < b".
   Input : two positive integers in the "main()" portion of the program.
   Output: the GCD, as returned by the "GCD()" function. [2]

3. Write a C function to check whether a positive integer is a palindrome or not, using the C function "void is_palindrome (int number)".
   A "palindromic number" is a number that remains the same when its digits are reversed. Eg: 16461
   Input: A positive integer.
   Output : "yes" if the number is a palindrome, else "no". [3]

4. You are given an array "A" of "n" integers. It is given that the elements of "A" satisfy the following inequalities:
   A[0] < A[1] < ... < A[m - 1] < A[m] > A[m + 1] > A[m + 2] > ... > A[n - 1]
   for some (unknown) index "m" in the range [1,n-2].
   Let us call such an array a "hill-valued" array. The sequence A[0], A[1], ... , A[m - 1], A[m] is called the "ascending part" of the hill, and the remaining part A[m], A[m+1], . . ., A[n - 1] is called the "descending part" of the hill. The element A[m] is the "peak" of the hill and is the largest element in the array. Your task is to locate the peak, i.e., to determine the values of "m" and "A[m]".
   Write a C function "int binarysearchpeak (int array[], int first_index, int last_index)" to perform the task, by performing a (non-recursive) binary search. The initial call to the function is of the form "m = binarysearchpeak(A, 0, n-1)" where "A" is a hill-valued array holding "n" integers. The function returns the value of "m". Write a complete C program to demonstrate the working of the function. You might have a hard-coded hill-valued array of (say) 10 integers inside your "main()" body.
   Input: Nothing
   Output : the value of "m" and the value of "A[m]", printed from inside "main()". [3]