'''

Problem:

Given an array of positive numbers, find the maximum sum of a subsequence

with the constraint that no 2 numbers in the sequence should be adjacent in the array.

Requires:

efficient

how to record which object is select?

Ways:

Loop for all elements in arr[] and maintain two sums incl and excl

where incl = Max sum including the previous element

excl = Max sum excluding the previous element.

Ref:

https://www.geeksforgeeks.org/maximum-sum-such-that-no-two-elements-are-adjacent/

'''

def max\_subset\_no\_adjacent(arr):

incl = 0

excl = 0

for ind, idata in enumerate(arr):

# maximum value exclude considering current value

excl\_new = max(incl, excl)

# maximum value include considering current value

incl = excl + idata

excl = excl\_new

return max(incl, excl)

arr = [6, 7, 1, 3, 8, 2, 4]

max\_sum = max\_subset\_no\_adjacent(arr)

print(max\_sum)