

Assignment1: Algorithms

Name: K.Rohini

Roll number: EE21B066

QUESTION 1:

(a).

```
def sum_of_sqr(n):  
    if(n<=0):  
        return "invalid"  
    else:  
        sum=0  
        for i in range(n):  
            sum=sum+(i**2)  
        return sum
```

(b).

```
def sum_of_odd_sqr(n):  
    if(n<=0):  
        return "invalid"  
    else:  
        sum=0  
        for i in range(n):  
            if(i%2!=0):  
                sum=sum+(i**2)  
        return sum
```

QUESTION2:

```
print("for sequence 60,70,80")  
for i in range(60,90,10):  
    print(i, end=" ")  
print('\n')  
print("for sequence 4,2,0,-2,-4")
```

```
for i in range(4,-6,-2):
```

```
    print(i, end=" ")
```

QUESTION 3:

```
def odd_product(lst):
```

```
    for i in range(len(lst)):
```

```
        if lst[i]%2!=0:
```

```
            if lst[i] not in odd_lst:
```

```
                odd_lst.append(lst[i])
```

```
    if(len(odd_lst)>1):
```

```
        return "there is a distinct pair of numbers in the sequence whose product is odd"
```

```
    else:
```

```
        return "there is no distinct pair of numbers in the sequence whose product is odd"
```

QUESTION4:

```
def vowels_count(str):
```

```
    v_lower=['a','e','i','o','u']
```

```
    v_upper=['A','E','I','O','U']
```

```
    count=0
```

```
    for i in str:
```

```
        if i in v_lower:
```

```
            count+=1
```

```
        elif i in v_upper:
```

```
            count+=1
```

```
    return count
```

QUESTION5:

```
input_a=input("Enter an integer for 'a':")
```

```
input_b=input("Enter an integer for 'b':")
```

```
input_c=input("Enter an integer for 'c':")
```

```
a=int(input_a)
```

```
b=int(input_b)
```

```

c=int(input_c)
if(a+b)==c:
    print("a,b,c can be used in a+b=c")
if(b-c)==a:
    print("a,b,c can be used in a=b-c")
if(a*b)==c:
    print("a,b,c can be used in a*b=c")
else:
    print("a,b,c can't be used in the arithmetic formulae a+b=c,a=b-c,a*b=c")

```

QUESTION6:

```

def distinct_list(lst):
    unique_lst=[]
    for i in lst:
        if i not in unique_lst:
            unique_lst.append(i)
    print("The distinct elements in the given sequence are:")
    print(unique_lst)
    if len(lst)==len(unique_lst):
        return "elements of the sequence are distinct"
    else:
        return "elements of the sequence are not distinct"

```

QUESTION7:

```

def birthday_problem(n):
    prob=1
    for i in range(n):
        prob*=(1-(i/365))
    return 1-prob
num_of_people=input("Enter a positive integer:")
probability=birthday_problem(int(num_of_people))

```

```
print("the probability of two persons having the same birthday =" +str(probability))
```

QUESTION8:

```
def permute(s,answer):  
    if(len(s)==0):  
        print(answer, end=" ")  
        return  
    for i in range(len(s)):  
        ch=s[i]  
        left_substr=s[0:i]  
        right_substr=s[i+1:]  
        rest=left_substr+right_substr  
        permute(rest,answer+ch)  
answer=" "  
s='catdog'  
print("All possible strings are : ")  
permute(s,answer)
```

QUESTION9:

```
num=input("Enter a positive integer greater than 2:")  
number=int(num)  
if(number>2 and isinstance(number,int)):  
    count=0  
    for i in range(number):  
        if((number/2)>=2):  
            count+=1  
            number/=2  
            continue  
    else:  
        count+=1  
        break
```

```
    print("the number of times a number must repeatedly divide this number by 2 before getting a value  
less than 2 =",str(count))
```

```
else:
```

```
    print("Enter a positive integer greater than 2")
```