

G12_11_CS_Term2_Test-Solutions

January 13, 2020

1 The Dragon Academy

2 CS G11_12 Term 2 Test

2.1 Mo Jan 13 2020

3 Name:

Note: All questions weigh the same towards the final mark. Last question is only for G12.

4 1 Implement as an object the finite sum $F(x, \{a_i\}) = \sum_{n=0}^{100} a_n x^n$ with the following requirements

1. Instantiating an object for a specific set of values of a_i returns a callable function, $f1$, such that it computes the sum for any given x and those a_i .
2. When we print any specific instance it shows the list of values a_i , starting with $i = 0$

5 2 In Python the statement `s={3,3,4,5}` defines a set of only 3 values, 3,4,5. Implement sets as objects and add the union and intersection of sets as a operation.

1. Note you cannot use the built-in set object `{}`.
2. Also, use the syntax `A*B` for the intersection of sets A and B,
3. and `A+B` for the union.
4. When printing a set, say `s`, it should show `{3,4,5}`

6 3 For the finite sums of problem 1, implement the sum of finite sums.

7 Solutions

```
[50]: class F:
      def __init__(self,a):
          self.a = a
```

```

def f(self,x):
    s=0
    L = min(len(self.a),100)
    for n in range(L):
        s += self.a[n]*x**n
    return s
def __str__(self):
    return str(self.a)
def __call__(self,x):
    return self.f(x)
def __add__(self,g):
    h=[ a + b for a,b in list(zip(self.a,g.a)) ]
    d = len(self.a) - len(g.a)
    if d > 0:
        for i in range(len(g.a),len(self.a)):
            h.append(self.a[i])
    elif d < 0 :
        for i in range(len(self.a),len(g.a)):
            h.append(g.a[i])
    return F(h)

```

```

f1 = F([1,1,2])
f1(2)

```

[50]: 11

[12]: `print(f1)`

[1, 1, 2]

[46]: `g = F([0,0,0,1])`
`print(g)`

[0, 0, 0, 1]

[51]: `print(f1+g)`

[1, 1, 2, 1]

[39]: `class mSet:`
`def __init__(self,a):`
 `self.e = []`
 `for v in a:`
 `if v not in self.e: self.e.append(v)`
`def __str__(self):`
 `sstr='{'`

```

        for i in range(len(self.e)-1):
            sstr += str(self.e[i]) + ', '
        if len(self.e) > 0: sstr += str(self.e[-1] )
        sstr += '}'
        return sstr
    def __add__(self,B):
        s = [ e for e in self.e]
        for e in B.e:
            if e not in s: s.append(e)
        return mSet(s)
    def __mul__(self,B):
        si = []
        for e in B.e:
            if e in self.e: si.append(e)
        return mSet(si)

```

```

A = mSet([1,1,2,3])
B = mSet([3,4,7])
print(A,B)
print((A+B),(A*B))

```

```

{1, 2, 3} {3, 4, 7}
{1, 2, 3, 4, 7} {3}

```

```

[24]: a = [1,2]
      b=[3,4]
      c = a
      c

```

```

[24]: [1, 2]

```

```

[44]: c = list(zip(a,b))
      for x,y in c:
          print(x,y)

```

```

1 3
2 4

```

```

[:]:

```