Test A - UAI 655

|  |  |
| --- | --- |
| Name of student |  |
| Date |  |

# 1 What are results, what is meaning of both.

|  |
| --- |
| >>> m = 28  >>> n = 5 |
| >>> m // n  >>> m % n |
| Meaning //  Meaning % |

# 2 What is result

|  |
| --- |
| >>> m > n |
|  |

# 3 What are results

|  |
| --- |
| >>> m = True  >>> n = False |
| >>> m and n  >>> m or n |

# 4 What are results

|  |
| --- |
| >>> 7 \* 5 + 8 |
|  |
| >>> "5" + "8" |
|  |
| >>> "5" + 8 |
|  |

# 5 What are types

|  |
| --- |
| >>> m = input("Enter number ")  Enter number 5  >>> type(m) |
| What type is m ? |
| >>> m = (4,5,"abc",0xaa)  >>> type(m) |
| What type is m ? |
| >>> m = [4,5,"abc",0xaa]  >>> type(m) |
| What type is m ? |
| >>> m = [{1:{1,2,3},2:{9,8,0}}]  >>> type(m) |
| What type is m ? |
| >>> type(m[0]) |
| What type is m[0] ? |
| >>> type(m[0][1]) |
| What type is m[0][1] ? |
| >>> len(m) |
| What length is m ? |
| >>> len(m[0]) |
| What length is m[0] ? |

# 6 What is result

|  |
| --- |
| >>> m = {"mammal":"cat","fish":"trout","bird":"owl"}  >>> print(m.keys()) |
|  |
| >>> print(m.items()) |
|  |
| >>> for k,v in m:  print(k) |
|  |

# 7 Better solution

|  |
| --- |
| >>> if i > 10 and i < 19:  print(“something”) |
| Suggest better solution |

# 8 What is output, describe difference

|  |
| --- |
| >>> i=0  >>> while i < 15:  if i % 3 == 0:  print(i)  i += 1  >>> i=0  >>> while i < 15:  if i % 3 == 0:  print(i)  i += 1 |
| Outputs |
| Describe differences - which one is wrong |

# 9 Suggest a small python program

|  |
| --- |
| >>> a="CCGGCGGCGCCGGCCCCCCGGCGGCGCCGGCCCCGGGGGGGGG"  >>> b="TAATTTTAAATATATATTTTTAAAAATTTATATTTTAAAAATA" |
| Print a python program which will print pairs of a and b strings   * Every 5th character in a and b string, but b is moved 3 character left |
|  |

# 10 Substrings

|  |
| --- |
| >>> m = "I have a cat"  >>> m[7:9] |
|  |
| >>> m[:] |
|  |
| >>> [i\*2for i in m[:5]] |
|  |

# 11 What is result

|  |
| --- |
| >>> m = {"a","b","c"}  >>> m.add("c")  >>> m.add("d")  >>> m |
|  |
| >>> m=["a","b","c"]  >>> m.append("c")  >>> m.append("d")  >>> m |
|  |

# 12 How it works, what is the result

|  |
| --- |
| >>> m = "Unless you're also worried about newlines, in which case a-zA-Z \n. I'm trying to find a regex that would lump both into one but using \\w or \\W isn't giving me the desired behavior. You might just need to add \n if that's the case. – limasxgoesto0 Mar 20 '14 at 0:51"  >>> ''.join([i for i in m if i.isalpha() or i == ' ' or len(i) > 1]) |
|  |

# 13 Make it better

|  |
| --- |
| >>> def abc(x):  print(20/x)  >>> abc(5)  >>> abc(0) |
|  |

# 14 What is output i – print all i

|  |
| --- |
| >>> for i in range(3,9):  if i == 6:  break  print(i) |
|  |
| >>> for i in range(3,9):  if i == 6:  continue  print(i) |
|  |
| >>> for i in range(3,9):  if i == 6:  pass  print(i) |
|  |

# 15 Two dictionaries

|  |
| --- |
| >>> a = { 'x' : 1, 'y' : 2, 'z' : 3 }  >>> b = { 'w' : 10, 'x' : 11, 'y' : 2 } |
| >>> print('This means:', a.keys() & b.keys())  Output is:  >>> print('This means:', a.keys() - b.keys())  Output is:  >>> print('(key,value) this means:', a.items() & b.items())  Output is: |