Name:

Python Code

```
def makeRing( maker ):
    return "powerful mojo made by " + maker

Sauron = "a bad dude"

theOneRing = makeRing( Sauron )

elves = 50000

men = 50000

orcs = 200000

orcsStinkScaleFactor = 0.5

lastAlliance = elves + men

print "orcs remaining: " + str( lastAlliance - orcs * orcsStinkScaleFactor )
```

Questions

1. (8 points) Write down a representation of the global symbol table (name, value, type) at the last line of code. The symbol table should include all identifiers defined by the code up to that point.

2. (4 points) In what symbol table does the parameter `maker` belong?

3. (4 points) What does the last line of code print?

4. (4 points) What purpose does `str` play in the last line? Is it necessary? Why?

Name:

Python Code

```
def yearsPass( N ):
    s = "."
    for i in range(N):
        s = s + "."
    print s

def betrayedIsildur( thing ):
    print 'oops'
    return 'tricksy ' + thing

def GoesMissing( theOneRing ):
    Isildur = theOneRing
    return betrayedIsildur( theOneRing )

theOneRing = "powerful mojo"

theOneRing = GoesMissing( theOneRing )

gollum = theOneRing

yearsPass( 500 )

Bilbo = theOneRing
gollum = 'angry'

yearsPass(60)
```

Questions

1. (5 points) How many times does the for loop in the function *yearsPass* execute when the function is called?

2. (5 points) What is the value of the variable *theOneRing* at the end of the code?

3. (5 points) What is the value of the variable *Bilbo* at the end of the code?

4. (5 points) What does the function *yearsPass* print out the two times it is called (describe it, don't write it down)? Does it print the same thing each time?

Name:

Python Code

```
places = [ 'Bag End', 'Shire', 'Woody End', 'Brandywine', 'Cricket Hollow' ]

adventures = [ 'Old Forest', 'Mushrooms', 'Walking', 'Party' ]

hobbits = [ 'Merry', 'Pippin', 'Sam', 'Frodo', 'Bilbo' ]

players = ['Black riders', 'Farmer Maggot', hobbits ]

rings = [9, 7, 3, 1]

print players[2][4] + ' throws a ' + adventures[-1]

print players[-1][-1] + ' passes to ' + players[2][3] +  ' the ' + str( rings[-1] )

sentence = adventures[2] + ' to ' + places[4] + ', '

for i in range( len( hobbits ) - 2 ):
    sentence = sentence + hobbits[ i ] + ', '

sentence = sentence + 'and ' + players[-1][-2]

sentence = sentence + ' see ' + players[-3]

print sentence
```

Questions

1. (5 points) What does the first print statement write?

2. (5 points) What does the second print statement write?

3. (5 points) How many times does the `for` loop iterate, and what is the value of `i` each time through the loop?

4. (5 points) What does the last print statement write?

Name:

Python Code

```
def conjunction( wordlist ):
    s = ''
    for word in wordlist[:-1]:
        s = s + word + ", "
    s = s + 'and ' + wordlist[-1]
    return s

def BarrowDowns(wanderers):
    Wight = [ 'cold', 'dark', 'stone' ]
    return wanderers + Wight

def OldForest(wayfarers):
    oldManWillow = [ 'songs', 'roots', 'whispers' ]
    Bombadil = [ 'eldest', 'swiftest', 'merriest' ]

    sentence = 'Old Man Willow uses his ' + oldManWillow[0] + ' to bring '
    sentence = sentence + conjunction( wayfarers ) + ' to the river.'
    print sentence

    if Bombadil[0] == 'eldest':
        oldManWillow.append( 'Eat earth! Dig deep! Drink water! Go to sleep!' )

    trouble = BarrowDowns( wayfarers )

    if Bombadil[1] == 'swiftest':
        wayfarers = trouble[0:4]

    print conjunction( wayfarers ) + ' take knives from the barrow.'

if __name__ == "__main__":
    hobbits = [ 'Merry', 'Pippin', 'Sam', 'Frodo' ]
    OldForest( hobbits )
```

Questions

1. (5 points) What would the function call `conjunction( Bombadil )` return?

2. (5 points) What does the variable `trouble` contain at the end of the oldForest function?

3. (5 points) What does the first print statement write?

4. (5 points) What does the second print statement write?

Name:

Python Code

```
def NightAtBree( guests ):
  guests.append( [ 'Aragorn', 'Narsil' ] )
  guests[3].append( 'Letter from Gandalf' )
  guests[2].append( 'Bill' )

def Weathertop( wanderers ):
  for wanderer in wanderers:
    if wanderer[0] == 'Frodo':
      wanderer.append( 'knife fragment' )

def FordOfBruinen( pursued ):
  pursued.append( [ 'Glorfindel', 'Asfolath' ] )

def Rivendell( arrivals ):
  for safe in arrivals:
    if safe[0] == 'Frodo':
      safe.pop( -1 )
      safe.pop( 1 )

def journeyToRivendell( ):
  travelers = [ ['Merry', 'barrow knife'], ['Pippin', 'barrow knife'],
                [ 'Sam', 'barrow knife', 'big pack'], ['Frodo', 'barrow knife']  ]
  NightAtBree( travelers )
  print travelers[2], travelers[3]  # first print

  Weathertop( travelers )
  print travelers[3], travelers[4]  # second print

  FordOfBruinen( travelers )
  print travelers[5]  # third print

  Rivendell( travelers )
  print travelers[3]  # fourth print
```

Questions

1. (5 points) What does the first print statement write?

2. (5 points) What does the second print statement write?

3. (5 points) What does the third print statement write?

4. (5 points) What does the fourth print statement write?

Name:

Python Code

```
def Caradhras( fellowship ):
  print 'This shall be the death of the hobbits'
  return fellowship.split( ',' )

def Moria( intruders ):
  print 'Fly, you fools!'
  return intruders[1:]

def Lothlorien( seed ):
  branch = 'R[+FL][-FL]'
  for i in range( 2 ):
    seed = seed.replace( 'F', branch )
  return seed

def journeySouth():
  fellowship = "Gandalf,Aragorn,Frodo,Sam,Merry,Pippin,Legolas,Gimli,Boromir,Bill"

  travelers = Caradhras( fellowship )
  # mark 1

  travelers = Moria( travelers[:-1] )
  # mark 2

  goldenTree = Lothlorien( 'F' )
  fp = file( 'CerinAmroth', 'w' )
  fp.write( goldenTree + '\n' )
  fp.close()
```

Questions

1. (5 points) Show the journeySouth symbol table at # mark 1 (name, type, value).

2. (5 points) Show the global symbol table after Python has read through all of the code.

3. (5 points) What does the variable travelers contain at # mark 2?

4. (5 points) Write the name of the file the program creates and then write out what it contains.

Name:

Python Code

```
import random

class Character:
    def __init__(self, type):
        self.health = random.randint( 5, 10 )
        self.type = type

    def getType(self):
        return self.type

    def whack(self, other):
        other.health = 0

def AmonHen():
    Boromir = Character( 'Human' ) # mark 1
    Merry = Character( 'Hobbit' )
    Pippin = Character( 'Hobbit' )
    orcs = []
    for i in range( 100 ):
        orcs.append( Character( random.choice( ['Uruk Hai', 'Northern orc' ] ) ) )

    Merry.whack( orcs[ random.randint(0, 99) ] )
    for orc in orcs:
        Boromir.whack( orc )
        if orc.getType() == 'Uruk Hai' and random.random() < 0.1:
            orc.whack(Boromir)  # mark 2
            break
    orcs = orcs + [Merry, Pippin]
```

Questions

1. (5 points) Write down the global symbol table after Python has read through the code.

2. (5 points) How many times does the `__init__` function of the Character class execute?

3. (5 points) Write down the symbol table for the object referenced by Boromir at `# mark 1`.

4. (5 points) Explain, using symbol tables, what the line at `# mark 2` does.

Name:

Python Code

```
class Character:
    def __init__(self, attributes):
        self.att = dict( attributes )

    def modify(self, other, attribute, delta, default = 0):
        other.att[ attribute ] = other.att.get( attribute, default ) + delta

def HelmsDeep( companions ):
    orcs, lots = [], 100
    for i in range(10000):
        orcs.append( Character( [ ['type', 'orc'], ['health', 10] ] ) )
    for i in range(42):
        if i == 41:
            orcs[i].modify( companions['Gimli'], 'health', -1 )  # mark 1
        companions['Gimli'].modify( orcs[i], 'health', -10 )
    for i in range(42, 42+41):
        companions['Legolas'].modify( orcs[i], 'health', -10 )

    for i in range(42+41, 42+41+lots):
        companions['Aragorn'].modify( orcs[i], 'health', -10 )

    for i in range( 42+41+lots, 10000 ):
        Character( [ ['type', 'Huorn'] ] ).modify( orcs[i], 'health', -10 )

fellowship = {}
fellowship['Aragorn'] = Character( [ ['type', 'Dunedan'], ['health', 20] ] )
fellowship['Gimli'] =   Character( [ ['type', 'Dwarf'], ['health', 20] ] )
fellowship['Legolas'] = Character( [ ['type', 'Elf'], ['health', 20] ] )
fellowship['Gandalf'] = Character( [ ['type', 'Wizard'], ['power', 11] ] )
HelmsDeep( fellowship )
```

Questions

1. (5 points) At the end of the code, what are all of the keys in `fellowship`?

2. (5 points) Show the symbol table for the object referenced by fellowship['Aragorn'].

3. (5 points) Explain what the `modify` method does using `# mark 1` as an example.

4. (5 points) Give the range of indices for the orcs Aragorn modifies in the HelmsDeep function.

Name:

Python Code

```
def EmynMuil( climbers, elevation ):
    if elevation > 18:
        return EmynMuil( climbers, elevation-5 )
    else:
        climbers[ 'Gollum' ] = 'Smeagol'
        return climbers

def DeadMarshes( treaders, marshiness ):
    if len( marshiness ) < 15:
        print marshiness
        DeadMarshes( treaders, 'very ' + marshiness )
        print marshiness
    else:
        print marshiness

def TheBlackGate( trespassers ):
    trespassers[ 'Gollum' ] = [ 'Slinker', 'Stinker' ]
    return 'Cirith Ungol'

def Ithilien( visitors, goal ):
    visitors[ 'Faramir' ] = ['Captain']
    visitors[ 'Sam' ] = 'relaxed'
    visitors[ 'Faramir' ].append( 'Quality' )
    del visitors['Faramir']
    vistors[ 'Sam' ] = 'watchful'
    return goal + ', past Minas Morgul'

wanderers = EmynMuil( { 'Frodo' : 'powerful mojo', 'Sam' : 'cooking gear' }, 50 )
DeadMarshes( wanderers, 'marshy' )
TheWayIn = TheBlackGate( wanderers )
TheWayIn = Ithilien( wanderers, TheWayIn )
```

Questions

1. (5 points) Will the `EmynMuil` function ever stop? Explain your answer.

2. (5 points) What does the variable `TheWayIn` contain at the end of the code?

3. (5 points) What does the variable `wanderers` contain at the end of the code?

4. (5 points) What does the function DeadMarshes print when executed as called?

Name:

Python Code

```
class Location:
    def __init__( self, agents ):
        self.characters = agents[:]
    def leave( self, name ):
        self.characters.remove( name )
    def arrive( self, name ):
        self.characters.append( name )

agents = [ 'Theoden', 'Aragorn', 'Gandalf', 'Eomer', 'Merry', 'Pippin',
           'Legolas', 'Gimli', 'Saruman', 'Treebeard' ]
Isengard = Location( agents )
HollowByIsen, HelmsDeep = Location( [] ), Location( [] )  # this assigns two variables
MinasTirith, Dunharrow = Location( [ 'Denethor' ] ), Location( [ 'Eowyn' ] ) # this too
PathsOfTheDead = Location( [ 'The Dead' ] )
DruadanForest = Location( [ 'Ghan-Buri-Ghan' ] )

for agent in agents[0:8]:  # Saruman's staff is broken
    Isengard.leave( agent )
    HollowByIsen.arrive( agent )
print "Whoops, shouldn't have done that"  # Pippin looks in the stone
for agent in [ agents[2], agents[5] ]:  # A winged Nazgul passes over and the race begins
    HollowByIsen.leave( agent )
    MinasTirith.arrive( agent )

for agent in [ agents[0], agents[1], agents[3], agents[4], agents[6], agents[7] ]:
    HollowByIsen.leave( agent )
    HelmsDeep.arrive( agent )
HelmsDeep.arrive( 'Dunedain' ) # Aragorn's kin find him in Rohan

for agent in [ agents[0], agents[3], agents[4] ]:  # off to the muster
    HelmsDeep.leave( agent )

for agent in [ agents[1], agents[6], agents[7], 'Dunedain' ]: # challenging Sauron
    HelmsDeep.leave( agent )
    Dunharrow.arrive( agent )
    Dunharrow.leave( agent )
    PathsOfTheDead.arrive( agent )

for agent in [ agents[0], agents[3], agents[4] ]:  # the red arrow arrives
    Dunharrow.arrive( agent )

for agent in [ agents[0], agents[3], agents[4], 'Eowyn' ]:  # the Rohirrim head south
    Dunharrow.leave( agent )
    DruadanForest.arrive( agent )
```

Questions

1. (5 points) At the end, which Location object's character list holds the string 'Merry'?
2. (5 points) At the end, what is the value of the expression Isengard.characters[1]?
3. (5 points) At the end, what Python expression would have the value 'Aragorn'?
4. (5 points) At the end, what Python expression would have the value 'Gandalf'?

Code Sample 1

```
Gondor = [ 'West Osgiliath', 'Causeway Forts',
           'Pelennor Fields', 'Gate', 'Minas Tirith' ]
Mordor = [ 'Crossroads', 'Ithilien', 'East  Osgiliath' ]

Faramir = ['Captain']
Denethor = [ 'Steward of Gondor', ['morale', 0.9] ]
LordOfNazgul = [ 'Sorcerer', 'Ringwraith' ]
Gandalf = ['Wizard', 'Staff', 'Glamdring']

Orcs = 100000
MenOfGondor = 10000
morale = 1.0

Mordor.append( Gondor[0] )
Gondor[0] = ''
MenOfGondor -= 500
morale -= 0.1

Mordor.append( Gondor[1] )
Gondor[1] = ''

Faramir.append( 'wounded' )
MenOfGondor -= 500
morale -= 0.5
Denethor[1][1] = 0.2

Mordor.append( Gondor[2] )
Gondor[2] = ''
MenOfGondor -= 500
morale -= 0.2
Denethor[1][1] = 0.0

Gondor[3] = 'splinters'
LordOfNazgul.append( 'Crosses the  Gate' )
Gandalf.append( 'Blocks the way' )
Dawn = ( 'rooster crowing', 'Horns of Rohan' )
LordOfNazgul.pop()
morale += 0.7
```

Code Sample 2

```
def ChargeOfTheRohirrim( groupA, groupB, groupC, dA, dB, dC ):
    groupA[1] -= dA
    groupB[1] -= dB
    groupC[1] -= dC

def WitchKing( sideA, sideB ):
    wounded = []
    wounded.append( sideA.pop() )
    wounded.append( sideB.pop() )
    wounded.append( sideA.pop() )
    wounded.remove( 'Lord of the Nazgul' )
    wounded.append( sideA.pop() )
    wounded.remove( 'Theoden' )

    return wounded

def ChargeOfDunedain( groups, deltas ):
    for i in range( len(groups) ):
        groups[i][1] -= deltas[i]

def BattleInEarnest( gA, gB, gC, gD ):
    gA[1][1] = 3000

    gB[1][1] = 3000
    for i in range(4):
        gB.pop()

    gC[5][1] = 4000
    gC.pop()

    for i in range(4):
        gD[i][1] = 0

Rohirrim = [ 'Eomer', ['Rohirrim', 6000], 'Eowyn', 'Merry', 'Theoden' ]
Gondor = [ 'Imrahil', ['Men at arms', 5000], 'Hirluin',
           'Forlong', 'Derufin', 'Duilin' ]
Dunedain = [ 'Aragorn', 'Legolas', 'Gimli', 'Elladan', 'Elrohir',
             ['Lebennin', 5000], 'Halberad' ]
Mordor = [ ['Haradrim', 15000], ['Rhun', 10000], ['Southrons', 10000],
           ['orcs', 50000], 'Gothmog', 'Lord of the Nazgul' ]

HousesOfHealing = ['Faramir']

ChargeOfTheRohirrim( Rohirrim[1], Mordor[2], Mordor[3], 500, 8000, 10000 )

HousesOfHealing = HousesOfHealing + WitchKing( Rohirrim, Mordor )

ChargeOfDunedain( [ Dunedain[5], Mordor[0], Mordor[1], Mordor[3] ],
                  [ 1000, 5000, 5000, 30000 ] )

BattleInEarnest( Rohirrim, Gondor, Dunedain, Mordor )
```

Code Sample 3

```
class Agent():
    def __init__( self, name, health ):
        self.name = name
        self.health = health

    def whack( self, other, howhard ):
        other.health -= howhard

    def water( self, amount ):
        self.health += amount

    def food( self, amount ):
        self.health += amount

def TowerOfCirithUngol( eye, moon, sam ):
    for i in range( len( moon ) ):
        eye[i].whack( moon[i], moon[i].health )
        moon[i].whack( eye[i], eye[i].health )

    snaga = len( eye ) - 1
    sam.whack( eye[snaga], eye[snaga].health )

def MarchNorth( frodo, sam, gollum ):
    for i in range(3):
        frodo.water( 1 )
        sam.water( 1 )
        gollum.water( 1 )
        gollum.food( -1 )

def MarchEast( frodo, sam, gollum ):
    for i in range(3):
        frodo.water( 1 )
        sam.water( -1 )
        sam.food( -1 )
        gollum.water( -1 )
        gollum.food( -1 )

Sam = Agent( 'Sam', 20 )
Frodo = Agent( 'Frodo', 10 )
Gollum = Agent( 'Gollum', 10 )
EyeCompany = []
MoonCompany = []

for i in range( 200 ):
    EyeCompany.append( Agent( 'orc', 10 ) )
    MoonCompany.append( Agent( 'orc', 10 ) )
EyeCompany.append( Agent( 'Shagrat', 20 ) )
EyeCompany.append( Agent( 'Snaga', 10 ) )

TowerOfCirithUngol( EyeCompany, MoonCompany, Sam )
MarchNorth( Frodo, Sam, Gollum )
MarchEast( Frodo, Sam, Gollum )
```

Code Sample 4

```
class Thing:
    def __init__(self, name, attributes ):
        self.name = name
        self.attr = attributes[:]

class Agent( Thing ):
    def __init__(self, name, attributes, stuff):
        Thing.__init__(self, name, attributes)
        self.stuff = stuff[:]

class Place( Thing ):
    def __init__(self, name, attributes, who):
        Thing.__init__(self, name, attributes)
        self.who = who[:]

Sauron = Agent( 'Sauron', ['bad dude'], ['slaves', '8 Nazgul'] )
Frodo = Agent( 'Frodo', ['hobbit'], ['powerful mojo'] )
Sam = Agent( 'Sam', ['hobbit'], ['wooden box', 'star glass', 'sting'] )
Gollum = Agent( 'Gollum', ['hungry'], [] )
Aragorn = Agent( 'Aragorn', ['Chief of Dunedain'], ['Anduril'] )
Arwyn = Agent( 'Arwyn', ['Daughter of Elrond'], ['Gift of Eldar'] )
Gandalf = Agent( 'Gandalf', ['Wizard'], ['Glamdring'] )

BaradDur = Place( 'Barad Dur', ['impenetrable'], [Sauron] )
SammathNaur = Place( 'Sammath Naur', ['heart of power'],
                     [Frodo, Sam, Gollum] )
TowersOfTheTeeth = Place( 'Towers of the Teeth', ['Iron Gate'],
                          [Aragorn, Gandalf] )

Gollum.stuff.append( Frodo.stuff.pop() )
print 'Whoops' # mark 1

Gollum = None
Sauron = None
BaradDur = None
SammathNaur = None
TowersOfTheTeeth = None

Aragorn.attr.append( 'King of Gondor' )
Aragorn.attr.append( 'Married' )
Arwyn.attr.append( 'Married' )

Frodo.stuff.append( Sam.stuff.pop() )
Frodo.stuff.append( Sam.stuff.pop() )
Frodo.stuff.append( Arwyn.stuff.pop() )

Rivendell = Place( 'Rivendell', ['House of Elrond'], [Frodo, Sam, Gandalf] )
```

Code Sample 5

```
def BattleForTheShire( hb ):
    print hb[1]['name']+" scares the ruffians with his "+hb[1]['stuff'][1]
    print hb[0]['name']+' uses the '+hb[0]['stuff'][0]+' to raise the Shire'

def RebuildingOfTheShire( hb ):
    print "Hey " + hb[2]['name'] + ", what's in your " + hb[2]['stuff'][0]

def ElvenHome( guests ):
    for guest in guests:
        guest['happiness'] = guest.get('happiness', 10) + 1

    if guests[0]['happiness'] > 1:
        ElvenHome( guests )

Bilbo = { 'name' : 'Bilbo', 'stuff' : ['books'], 'location' : 'Rivendell' }
Frodo = { 'name' : 'Frodo', 'stuff' : ['star glass', 'Gift of Eldar', 'sting'],
          'location' : 'Rivendell' }
Sam = { 'name' : 'Sam', 'stuff' : ['wooden box'], 'location' : 'Rivendell' }
Gandalf = { 'name' : 'Gandalf', 'stuff' : [ 'staff', 'Glamdring', 'narya' ],
            'location' : 'Rivendell' }
Saruman = { 'name' : 'Saruman', 'stuff' : [ 'knife' ], 'location' : 'Shire' }
Merry = { 'name' : 'Merry', 'stuff' : ['Horn of Rohan'], 'location' : 'Rivendell' }
Pippin = { 'name' : 'Pippin', 'stuff' : ['Armor of Gondor', "troll's bane" ],
           'location' : 'Rivendell' }

travelers = [ Merry, Pippin, Sam, Frodo, Gandalf ]
for t in travelers:
    t['location'] = 'Bree'

Gandalf['location'] = 'House of Bombadil'
for t in travelers[0:4]:
    t['location'] = 'Shire'

BattleForTheShire( travelers[0:4] )

Saruman = None

RebuildingOfTheShire( travelers[0:4] )

Bilbo['location'] = 'Shire'
for t in travelers + [Bilbo]:
    t['location'] = 'Grey Havens'

for t in [ Bilbo, Frodo, Gandalf]:
    t['location'] = 'Grey Ship'

for t in travelers[0:3]:
    t['location'] = 'Shire'

print "Well, I'm back" # mark end

ElvenHome( [Bilbo, Frodo, Gandalf] )
```

Name:

Code Sample 1 Questions

1. (10 points) List all of the entries and their types in the global symbol table at the end of the code.

2. (5 points) What are the entries in the list Gondor at the end of the code?

3. (5 points) What are the entries in the list Denethor at the end of the code?

Code Sample 2 Questions

1. (5 points) Write out the contents of the list Mordor at the end of the code.

2. (5 points) Write out the contents of the list HousesOfHealing at the end of the code.

3. (5 points) Write out the contents of the list Rohirrim at the end of the code.

4. (5 points) Write the function symbol table–name, type, and value–at the end of WitchKing.

Code Sample 3 Questions

1. (5 points) Write the name and type for each entry in the Agent class symbol table.

2. (5 points) What is the value of the variable snaga in TowerOfCirithUngol?

3. (5 points) What is the value of the expression `Sam.health` at the end of the code?

4. (5 points) What is the value of the expression `EyeCompany[200].health` at the end of the code?

Code Sample 4 Questions

1. (5 points) Write the symbol table–name, type, value–for the object Sauron at `# mark 1`.

2. (5 points) Write the value of the expression `Frodo.stuff` at the end of the code.

3. (5 points) Write the value of the expression `Aragorn.attr` at the end of the code.

4. (5 points) Write the symbol table–name and type only–for the class Place.

Code Sample 5 Questions

1. (5 points) Write out the value of the expression `travelers[4]` at `# mark end`.

2. (5 points) What does the function BattleForTheShire print?

3. (5 points) What does the function RebulidingOfTheShire print?

4. (5 points) Does the function ElvenHome ever terminate? Explain your answer.