**PDBtoStructure:**

seq = []

curres = '---'

res = []

first = True

atomnum = 5

resnum = 0

for line in pdb.lines:

# Decode from byte array

line = line.decode()

# Grab only ATOM lines

if line[:6] == 'ATOM ':

# Get the atomname (N,CA,C,O)

atomname = line[12:16]

# First make sure you're only storing N, CA, C and O

if atomname not in (' N ',' CA ',' C ',' O ') or line[16] not in (' ','A'):

continue

# Check to see if residue changed

if line[17:20] != curres or atomnum > 4:

if self.debug:

print("%8s%8s%8s%8s" % ('CURRES','NEWRES','RESNUM','ATONUM'))

print("%8s%8s%8d%8d" % (curres,line[17:20],resnum,atomnum))

print(line)

# Check to make sure the last residue was complete

if atomnum < 5:

# Add on the missing atoms to the end of the residue

if atomnum == 1:

atom = Atom(atomName=' N ',resName=curres,elemSym=' N',missing=True)

res.append(atom)

atomnum += 1

if atomnum == 2:

atom = Atom(atomName=' CA ',resName=curres,elemSym=' C',missing=True)

res.append(atom)

atomnum += 1

if atomnum == 3:

atom = Atom(atomName=' C ',resName=curres,elemSym=' C',missing=True)

res.append(atom)

atomnum += 1

if atomnum == 4:

atom = Atom(atomName=' O ',resName=curres,elemSym=' O',missing=True)

res.append(atom)

atomnum += 1

# Append the residue to the sequence

if res: seq.append((amino3to1[curres],res))

# If this is the first atom fill in the blanks from the PDB

if first:

first = False

if self.debug:

print("Starting resnum:",int(line[22:26]))

# Residue number of first residue in PDB

for k in range(1,int(line[22:26])):

resname = '---'

res = []

res.append(Atom(atomName=' N ',resName=resname,elemSym=' N',missing=True))

res.append(Atom(atomName=' CA ',resName=resname,elemSym=' C',missing=True))

res.append(Atom(atomName=' C ',resName=resname,elemSym=' C',missing=True))

res.append(Atom(atomName=' O ',resName=resname,elemSym=' O',missing=True))

seq.append((amino3to1[resname],res))

# Decrement resnum by one to mimic the last res not in the PDB

resnum = int(line[22:26])-1

# Check for resnum increment so we don't miss residues

resinc = int(line[22:26])-resnum

if resinc != 1:

if self.debug:

print("Resiude skip:",resnum,"->",int(line[22:26]))

for k in range(1,resinc):

resname = '---'

res = []

res.append(Atom(atomName=' N ',resName=resname,elemSym=' N',missing=True))

res.append(Atom(atomName=' CA ',resName=resname,elemSym=' C',missing=True))

res.append(Atom(atomName=' C ',resName=resname,elemSym=' C',missing=True))

res.append(Atom(atomName=' O ',resName=resname,elemSym=' O',missing=True))

seq.append((amino3to1[resname],res))

# Reset the residue

res = []

# Store residue numbers

resnum = int(line[22:26])

# Store the atom number

atomnum = 1

# Make sure current residue name is set

curres = line[17:20]

'''

if self.debug:

print("Line:",line)

'''

# Check to make sure the right atom is being stored

if atomnum == 1:

if atomname != ' N ':

atom = Atom(atomName=' N ',resName=curres,elemSym=' N',missing=True)

res.append(atom)

atomnum += 1

if atomnum == 2:

if atomname != ' CA ':

atom = Atom(atomName=' CA ',resName=curres,elemSym=' C',missing=True)

res.append(atom)

atomnum += 1

if atomnum == 3:

if atomname != ' C ':

atom = Atom(atomName=' C ',resName=curres,elemSym=' C',missing=True)

res.append(atom)

atomnum += 1

if atomnum == 4:

if atomname != ' O ':

atom = Atom(atomName=' O ',resName=curres,elemSym=' O',missing=True)

res.append(atom)

atomnum += 1

continue

if atomnum > 4:

continue

# Store off the atom information if valid

atom = Atom(

atomName=atomname,

altLoc=line[16],

resName=line[17:20],

chainId=line[21],

codeForInsertion=line[26],

xcoord=float(line[30:38]),

ycoord=float(line[38:46]),

zcoord=float(line[46:54]),

occ=float(line[54:60]),

temp=float(line[60:66]),

elemSym=line[76:78],

charge=line[78:80]

)

# And add it to the residue (in the proper order)

res.append(atom)

atomnum += 1

# Add the final residue to the sequence

seq.append((amino3to1[curres],res))