```
##### This script splits of the assembly in subcontigs wherever there is a "N" stretch long
er than 30N

from Bio import SeqIO
from Bio.Seq import Seq
from Bio.SeqRecord import SeqRecord
from Bio.Alphabet import IUPAC
import glob

Assemblies = glob.glob("/media/avneesh/AneeshHDDfat/AssembledScaffolds/*")

N_stretch_length = 100

for file in Assemblies:
        NewFILEPath = str(file) + str("_splitted")
        newAssembly = open(NewFILEPath, "a")
        for seq in SeqIO.parse(file, "fasta"):
                base = -1
                seq_end = "no"
                new_sub_number = 0
                while base < len(seq.seq)-1:
                        base += 1
                        N_count = 0
                        if seq.seq[base] != "N":
                                N_count = 0
                                start = base
                                for a in range(start, len(seq.seq),1):
                                        if seq.seq[a] != "N":
                                                if a+1 == len(seq.seq):
                                                        seq_end = "yes"
                                        else:
                                                for b in range(a, len(seq.seq)+1,1):
                                                        if seq.seq[b] == "N":
                                                                N_count += 1
                                                        else:
                                                                base = b-1
                                                                break
                                        if N_count > N_stretch_length:
                                                new_sub_number += 1
                                                stop = a

                                                old_split_ID = seq.id.split("_cov_")
                                                old_split_ID[1] = "%s%s%s" % (str(old_split
_ID[1]), str("_"), str(new_sub_number))
                                                new_sequence = SeqRecord(Seq(str(seq.seq[st
art:stop])), id = "_cov_".join(old_split_ID),description="")    ### create new SeqRecord ob
ject
                                                SeqIO.write(new_sequence, newAssembly, "fas
ta")
                                          ### and write it to the new file

                                                break
                                        elif seq_end == "yes":
                                                new_sub_number += 1
                                                stop = a + 1
                                                base = len(seq.seq)
                        ## stops while loop
                                                old_split_ID = seq.id.split("_cov_")
                                                old_split_ID[1] = "%s%s%s" % (str(old_split
_ID[1]), str("_"), str(new_sub_number))
                                                new_sequence = SeqRecord(Seq(str(seq.seq[st
art:stop])), id = "_cov_".join(old_split_ID),description="")    ### create new SeqRecord ob
ject
                                                SeqIO.write(new_sequence, newAssembly, "fas
ta")
                                          ### and write it to the new file
                                                break
                                        else:
                                                pass

                        else:
                                pass
        print "%s%s" % (str(file.split("/")[-1]), " - done!")
```