Deduper Psuedocode

In order to properly quantify RNA counts, PCR duplicates must be removed from our SAM file (which is sorted by chromosome), but differentiating between PCR duplicates and true duplicates is difficult. Using UMIs, as well as some comparison checks, we can write out only true biological duplicates to an output file. This script will loop through each line of the file, get all the important identifying info from the line, and save this info to a set, if another line is reached where all identifying info, including the UMI, is the same, then it is not written out. In order to save memory space, whenever we hit a new chromosome, the duplicate set is cleared out, and the program continues on.

```
functions:

get_line_info(line):
'''Pulls out needed info from SAMfile line, returns array of that info'''
        get chromosome, position(un-softclipped), strand, and UMI from the
line
        return [chr, pos, strand, umi]


parse_bitflag(line):
'''Takes a file line, parses out the bitwise flag, then makes bitwise
comparison to determine the strand, returns + or -'''
        get the bitwise flag from line
        make bitwise comparison of flag to determine strand
        return positive or negative strand


softclip_adjustment(cigar_string):
'''This takes the cigar string and goes through letter by letter, if it runs
into an M before an S, it will output 0 for the adjustment, if it runs into
an S, it'll return the integer value of clip_num, which holds the value of
all the numbers left of S'''
        clip_num = ""
        for letter in cigar_string:
                if letter is "S"
                        return int(clip_num)
                if letter is "M"
```

```
                        return 0
                else:
                        clip_num.append(letter)



de_softclip(line):
'''Takes in line, pulls position and cigar string from the line, returns the
position minus the needed softclip adjustment'''
        get position from line
        get cigar_string from line
        return (position - softclip_adjustment(cigar_string))



validate_umi(umi):
        '''checks if umi is within our set or not'''
         returns True or False




dupset = {} # set with all of the unique line_infos that we have seen
chr_num = 1 # current chromosome number, once we get to the next chromosome
in the file we can wipe the dupset so it doesn't get too big

while True:

        line = readline(file)
        if line is empty:
                break

        line_info = get_line_info(line) # this is all the stuff we can use
as unique identifiers for a line, the chr, pos, strand, and umi, so we know
if its a duplicate

        chr = line_info[0]

        if chr != chr_num: # if the line is the next chromosome, clear the
dupset
                empty dupset
                chr_num +=1
        if line_info not in dupset and umi is valid:
                add line_info to dupset
                write out line to file
```