1. Some challenges include checking for bounds, checking for hyphens and words longer than max line length, and for extreme cases with arbitrary spacing.
2. Pseudocode:

return 2 if linelength less than 1 or more than 300

repeatedly:

check if word is bigger than max line length

repeatedly:

check if word contains hyphen

if yes, add 1 to position of hyphen,

break

say word portion can be split, or increment variable that tracks words too big

write newline if total characters equal max line

ignore zero byte characters

for space characters:

ignore multiple consecutive spaces and if cstring holder is empty

empty out holder, or print 2 new lines if paragraph break does not start a line

empty out holder and reset puncGo, spacGo, and hyphenCount

add length of word to total characters in line

initialize wordKeep to wordLength

reset wordLength

if total characters exceed max line length,

repeatedly:

write out character of word portion if there is a hyphen and write the rest on newline, or

add space character and increment spaceTotal

add another space character and increment spaceTotal if punctuation detected

increment spaceGo if spaceCount is zero

increment puncGo when puncSpace is zero

if no hyphen, write word on new line

repeatedly:

write the rest of characters of word in new line when characters in line equal max line length

for small words, repeatedly:

print out characters of word in the line

empty out holder

reset hyphenCount and poundTrack

continue

for punctuation:

increment wordlength

concatenate character to cstring holder

restart spaceCount and puncSpace variables

for hyphens:

increment wordlength

increment hyphenCount

concatenate character to cstring holder

continue

for pound:

increment poundCount if holder is empty

increment poundCount and poundTrack if holder is length 2

concatenate character to holder

for 'P':

increment poundP if poundCount == 1, or increment wordLength

and reset spaceCount and puncSpace.

concatenate character to holder

increment wordLength

concatenate character to holder

reset spaceCount and puncSpace

if no characters left:

empty out string if last word is a paragraph break

if word is too big,

check if there is a hyphen

split word after hyphen

write space characters before word portion if needed from

space and punctuation rules

if no hyphen then

repeatedly:

write characters until total characters equal linelength,

thenwrite newline and write until no characters left in string

for smaller words repeatedly:

write until no characters left in string

if no paragraph break detected,

write new line

return 1 if word is ever bigger than max line

return 0 otherwise

C)

Test cases:

**Key:** (.) and ‘ ‘ in between words both mean space, (\t) means tab,

The bash command on announcement, both with adding space at end and not adding space at end (used it because professor gave it to us)

Let’s check. Hi (line length = 20, simple punctuation case)

Let’s check. Hi(.) (line length = 20, simple punctuation case with space ending)

Let’s check. Hi.(.) (line length = 20, multi-punctuation case with space)

Let’s check. Hi.(.) (line length = 10, same case as above but with shorter line)

Let’s check. Hi. (line length = 20, multi-punctuation without space ending)

Let’s check. Hi. (line length = 10, same case as above but with shorter line)

Pre-departure (line length = 10, simple hyphen case)

Pre-departure(.) (linelength = 10, simple hyphen with space after)

Some pre-departure (linelength = 10, hyphen case with word in front)

Some. Pre-departure (linelength = 10, hyphen case with period before)

Pre----departure (linelength = 10, multiple hyphens in a row)

(\t)pre-departure (linelength = 10, space in front of word with hyphen)

Some. Pre-departure(.) (linelength = 10, hyphen with space at end)

Let’s check. Hi (lineLength = 10, punc case)

Let’s check. Hi(.) (lineLength = 10, punc case)

(linelength = 0, check to see if program followed spec)

Presumptuous (linelength = 10, simple case with word greater than linelength)

(\t)presumptuous (linelength = 10, simple case of long word with tab)

Presumptuous(.) (linelength = 10, simple case with space after)

#P# (linelength = 10, simple paragraph break case)

(.)#P# and #P#(.) (linelength = 10, simple par break cases with space in front and back)

Stuff #P# hi

Stuff #P# hi(.)

(.)Stuff #P# hi

(all three linelength = 10, par break cases with words in between)

Failed test case:

Hi.hi(.).hi(.) (line length = 10)