Task 1: software design and development (part A)

The measurement of how easy a sentence is to read is called the reading age. One method of calculating this is to use the average length of the words in the sentence.

So, the longer the average word length, the higher the reading age.

Program analysis
A program is required to calculate the approximate reading age (junior, teen or senior) of a single sentence with a maximum of 20 words. The user will enter each word of the sentence, one at a time. The program will then calculate the average word length and display an appropriate message.

1a Complete the analysis below by identifying the missing inputs.

(2 marks)

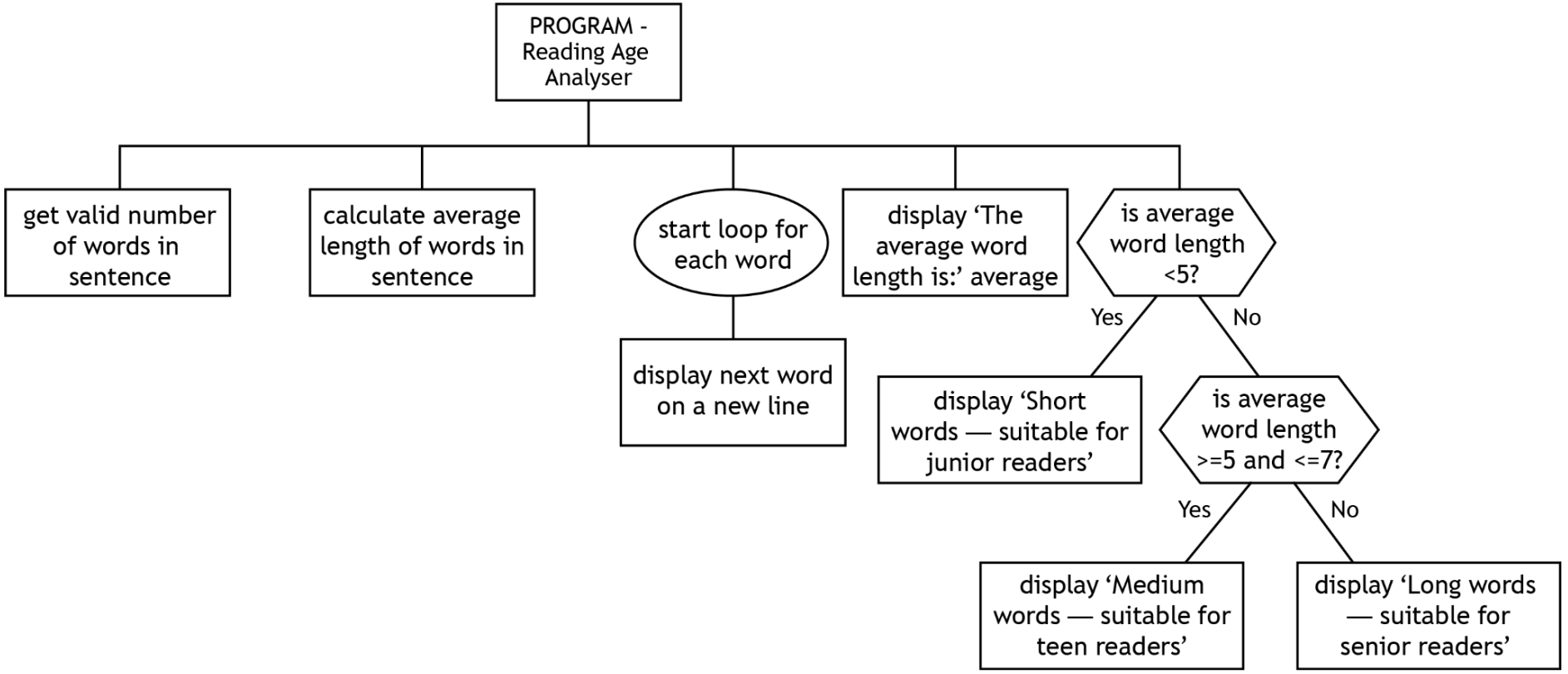
Inputs 1 mark for each Input: ♦ number of words (in sentence) to be entered ♦ each word in the sentence
Processes ♦ calculate the average word length (total number of characters divided by number of words) ♦ decide which message to display
Outputs ♦ each word in the sentence displayed on a new line ♦ one appropriate message from: — Short words – suitable for junior readers — Medium words – suitable for teen readers — Long words – suitable for senior readers

- Check your answers carefully, as you cannot return to part A after you hand it in.
- When you are ready, hand part A to your teacher or lecturer and collect part B.

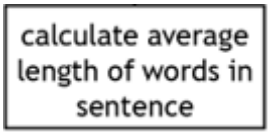
Candidate name_____Candidate number_____

Task 1: software design and development (part B)

Program design (structure diagram)



1b The structure diagram contains the following process:



Using the information provided in the program analysis, expand the design to show how this process could be carried out. You can use a flowchart, structure diagram or pseudocode design.

(3 marks)

1 mark for each:

- input of words in loop
- running total of word lengths in loop
- average calculated outside loop

Example Pseudocode:

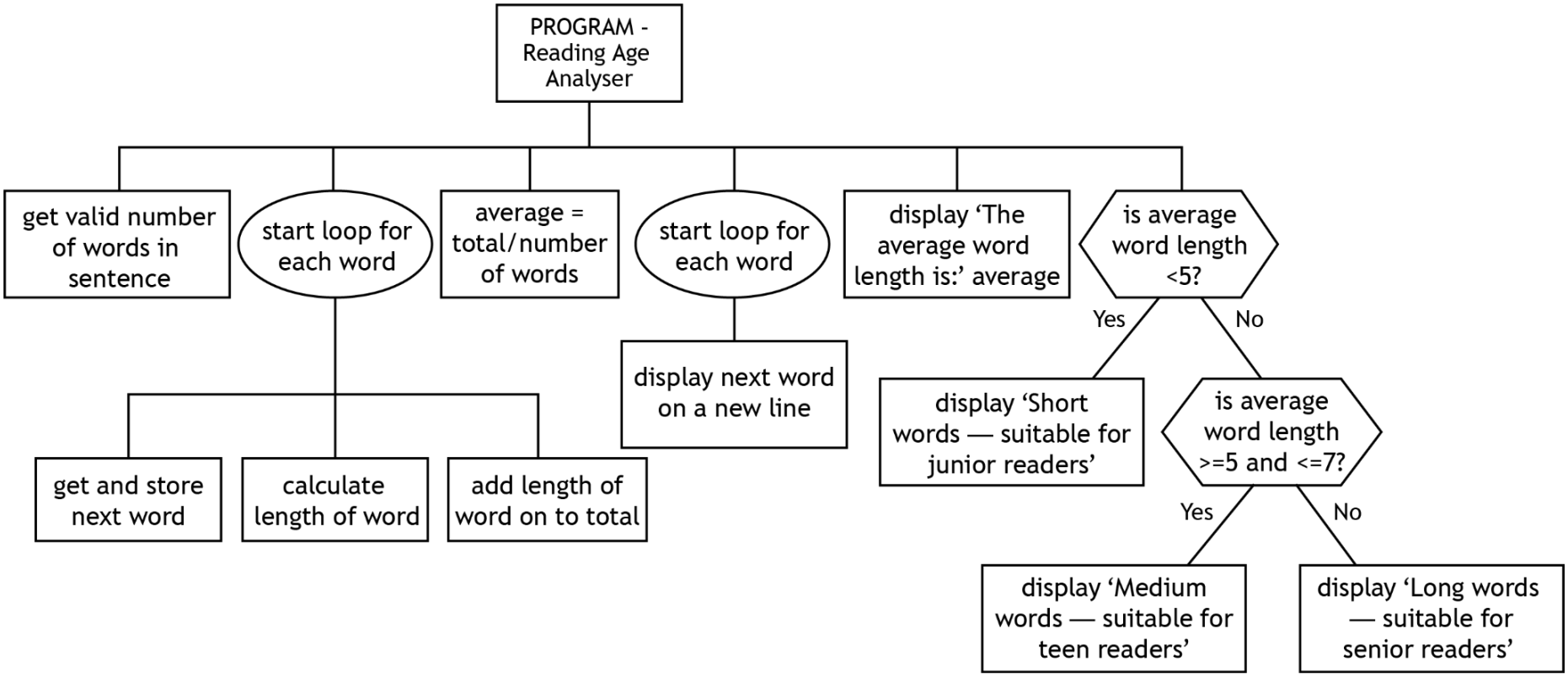
```
1. Start fixed loop for number_of_words
2.     Get and store next word
3.     Calculate length of word (LEN)
4.     Total = total + length
5. End Loop
6. Calculate average (total/number of words)
```

- Check your answers carefully, as you cannot return to part B after you hand it in.
- When you are ready, hand part B to your teacher or lecturer and collect part C.

Candidate name_____Candidate number_____

Task 1: software design and development (part C)

Program design (completed structure diagram)



1c Using the program analysis and the design, implement the program in a language of your choice.

Ensure the program matches the completed structure diagram.
Print evidence of your program code.

(15 marks)

```
#SQA Assignment 2020-21
#Example Solution

words=[""]*20
totalLength=0
NoOfWords =0
wordLength=0
averageLength =0

#Input Validation - get valid number of words Max 20
NoOfWords = int(input("Please enter the number of words in the sentence"))
while NoOfWords < 0 or NoOfWords > 20:
  NoOfWords = int(input("Invalid - please re-enter. Max 20 words"))


for counter in range(NoOfWords): #Start loop for each word

  #fill the array with words entered from user
  words[counter]=input("Please enter word " + str(counter + 1))

  #return the length of the word entered
  wordLength=len(words[counter])

  #keep a running total for total length of sentence
  totalLength = totalLength + wordLength

#calculate average word length
averageLength = totalLength/NoOfWords

print("_____")

#start loop to display each word on new line
for index in range(NoOfWords):
  print(words[index])

print ("The average word length is " +str(averageLength))

#determine reading age
if averageLength < 5:
  print("Short words - suitable for Junior readers")
elif averageLength >=5 and averageLength <=7:
  print("Medium words - suitable for teen readers")
else:
  print("Long words - suitable for senior readers")
```

1d (i) You should test your program to ensure it produces the expected output.

Use the following data to check that the message 'Long words — suitable for senior readers' is displayed:

Number of words:	5
Words in sentence:	Distressed
	tourists
	wandering
	around
	aimlessly

Run your program to show that it produces the correct message.
Print evidence of the test run showing inputs and outputs.

(1 mark)

```
Please enter the number of words in the sentence 5
Please enter word 1 Distressed
Please enter word 2 tourists
Please enter word 3 wandering
Please enter word 4 around
Please enter word 5 aimlessly
------------------------------
Distressed
tourists
wandering
around
aimlessly
The average word length is 8.4
Long words - suitable for senior readers
```

(ii) Additional test data is required to check that the other two messages are also displayed correctly.

Complete the test table below with data that could be used to produce the other two messages.

Test data	Expected results
Number of words: 3 Sentence: How are you?	'Short words — suitable for junior readers' is displayed.

Number of words: 4 Sentence: weather outside is cold	'Medium words – suitable for teen readers' is displayed.

(2 marks)

```
Powered by trinket
Please enter the number of words in the sentence 3
Please enter word 1 how
Please enter word 2 are
Please enter word 3 you?

_____________________________
how
are
you?
The average word length is 3.33333333333
Short words - suitable for Junior readers
```

```
Powered by trinket
Please enter the number of words in the sentence 4
Please enter word 1 weather
Please enter word 2 outside
Please enter word 3 is
Please enter word 4 cold

_____________________________
weather
outside
is
cold
The average word length is 5.25
Medium words - suitable for teen readers
```

Candidate name_____ Candidate number_____

1e With reference to your code, evaluate your program by commenting on the following:

Efficiency of your program code **(1 mark)**

Evaluation of efficiency of constructs within the candidates own code may include:

efficiency of use of

- an array instead of separate variables to store words
- loops to reduce code
- nested if statements preventing unnecessary code execution

inefficiency: discussion of own code if it varies from the supplied design

Readability of your program code **(1 mark)**

My code is readable as I have used good programming techniques.
I have used:

- Meaningful variable names
- Good use of white space. Python uses indentation by default but this also helps readability.

Internal commentary throughout to explain the main parts of my code. This could help with maintenance in the future.

Candidate name_____ Candidate number_____