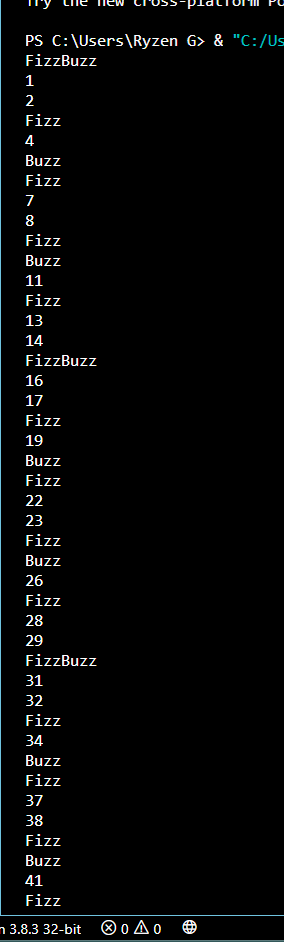
# LESSON ACTIVITIES FOR YEAR 2, TERM 1

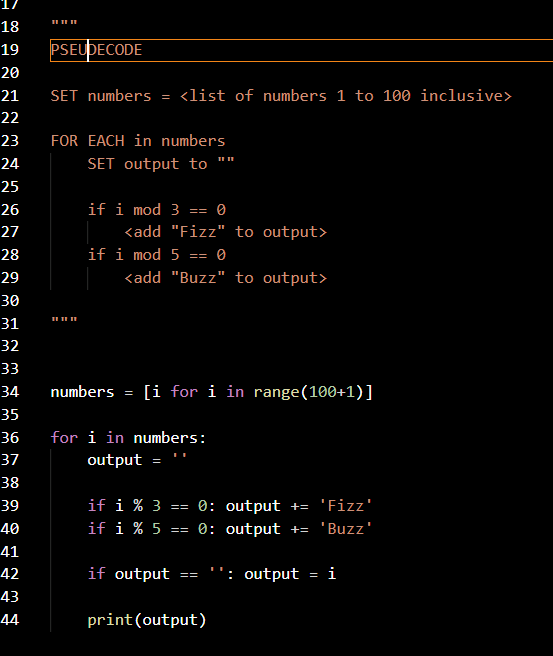
# Week 1

Lesson 1 activities

### Activity 1.1.1

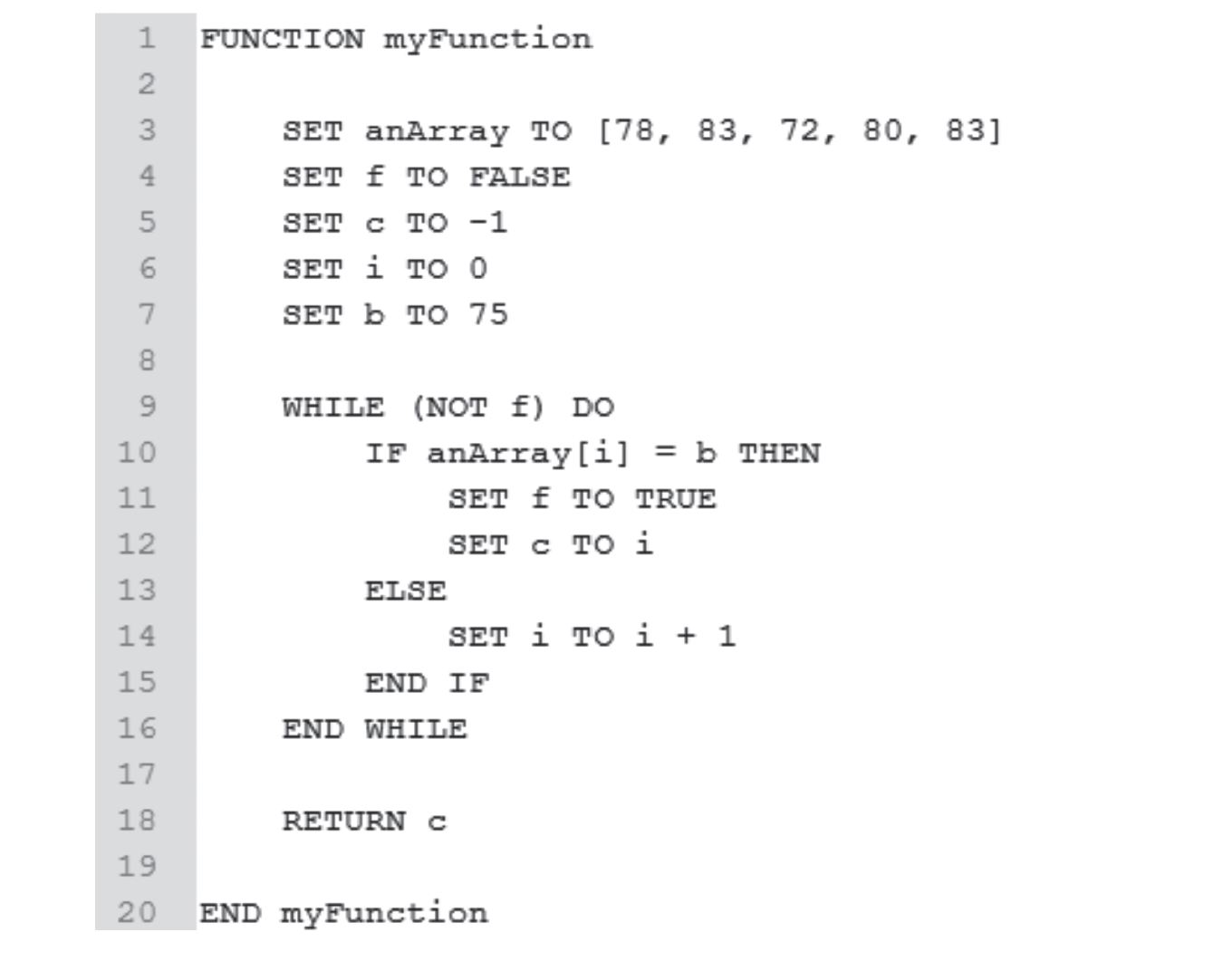
Write an algorithm in pseudocode for a program that produces a list of numbers from 1 to 100, replacing any number divisible by 3 with the word ‘Fizz’ and any number divisible by 5 with the word ‘Buzz’. Whenever a number is divisible by both 3 and 5 the word ‘FizzBuzz’ should be displayed.

Once you are happy with your solution, implement and test it in Python.

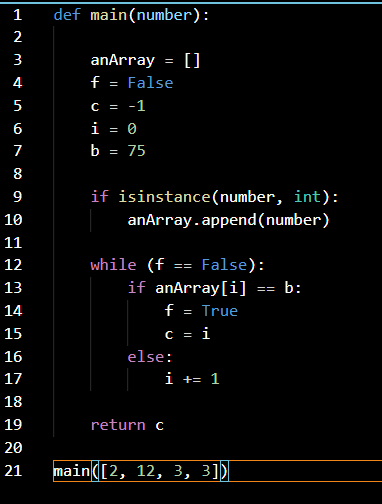


### Activity 1.1.2 (homework)

Study this algorithm and then answer the questions below.



|  |
| --- |
| State the line number that shows a declaration of a data structure. |
| 3 |
| State what type of data structure it is. |
| One dimensional array |
| State the range of lines that show a repetition. |
| 9 to 16 inclusive |
| State the range of line numbers that show a selection. |
| 10 to 15 inclusive |
| List all the variables used in the algorithm and state their data types |
| anArray: array  f: integer  c: integer  i: integer  b: integer |
| **There’s a bug in line 9. Write a line of code to fix the bug.** |
| WHILE f == False DO |
| This type of logic error is sometimes referred to as a ‘off-by-one’ error (OBOE). Explain why this is an apt name. |
|  |
| A syntax error is another type of error. Give a definition of a syntax error. |
| A mistake in the writing of a command or line, for example, forgetting to indent a code block in python |
| What will be returned to the main program when this function is run. |
| -1 |
| Amend the function so that it will work with any number and any list |
|  |
| Write the code to call the function from the main program. |
| below |



Lesson 2 activities

### Activity 1.2.1

1. Disley History Society recently held a week-long exhibition of medieval artefacts excavated from sites in the area. To help with planning the next event, the organisers want to know the maximum number of people who visited the exhibition on any one day.

This program is meant to do this, but it has two logic errors in it.

Identify and correct the errors.

visitors = [122, 51, 147, 91, 73, 124, 61]

index = 0

maximum = 1000

while index <= len(visitors):

if visitors[index] > maximum:

maximum = visitors[index]

index = index + 1

print('The maximum number of visitors on any one day was', maximum)

|  |
| --- |
| **Error 1** |
| while index < len(visitors): |
| **Error 2** |
| Maximum = 0 |

1. Syntax and runtime are two other types of error. Complete the empty cells in the table below to define and give examples of each.

|  |  |  |
| --- | --- | --- |
| **Type of error** | **Definition** | **Example** |
| **Syntax** | When code is written is incorrectly | Forgetting to end a line with a semi colon in C/C++ |
| **Runtime** | An error that occurs when a program is running, after it has compiled | Dividing by 0 |

### Activity 1.2.2

A computer programmer has written this code to calculate how many days during the exhibition extra staff were needed.

e = ['Y', 'N', 'Y', 'Y', 'N', 'Y', 'N']

l = len(e)

n = 0

i = 0

while True:

if e[i] == 'Y':

n += 1

i += 1

if i == l:

break

print('Extra staff were needed on', n, ‘days.’))

|  |  |
| --- | --- |
| **Identify** |  |
| An integer variable initialisation | n = 0 |
| A selection construct | If == i: |
| An iteration construct | n += 1 |
| A data structure | e = ['Y', 'N', 'Y', 'Y', 'N', 'Y', 'N'] |

1. State **two** techniques the programmer could have used to make the code easier to read.

Technique 1:

Use descriptive variable names

Technique 2:

Separate the print statement and integer initializations from the main loop

1. Write down the purpose and suggest a better name for each of the variables used in the program.

|  |  |  |
| --- | --- | --- |
| **Variable** | **Purpose** | **Better name** |
| **e** | Hold the array | week |
| **l** | Length of the array, how long the main loop should repeat for | arrLength |
| **n** | Count the number of days | days |
| **i** | Hold the index of the array that is currently being iterated | index |

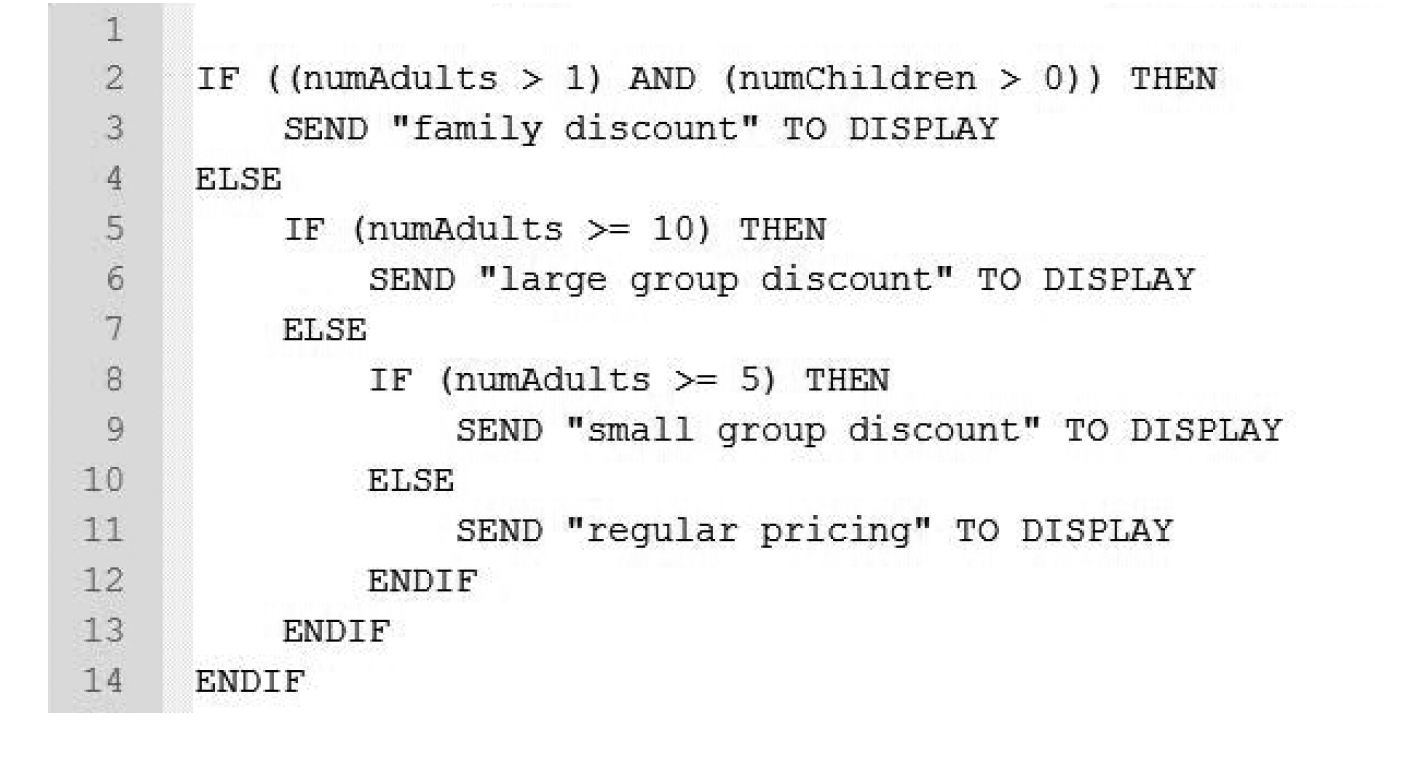
1. The program uses a ‘while True’ loop and a ‘Break’ statement. Explain how this works.

The while True loop repeats forever until something causes it to stop, like the break statement.

1. Complete this trace table showing the value of each of the variables during execution.

|  |  |  |  |
| --- | --- | --- | --- |
| **arrLength** | **Days** | **Index** | **week[index]** |
| 7 | 1 | 1 | ‘Y’ |
| 7 | 2 | 1 | N |
| 7 | 3 | 2 | Y |
| 7 | 4 | 3 | Y |
| 7 | 5 | 3 | N |
| 7 | 6 | 4 | Y |
| 7 | 7 | 4 | N |
|  |  |  |  |

### Activity 1.2.3

Here is an algorithm that determines discounts to an attraction based on the number of people in a group.

Complete the table to show the output of the algorithm, based on the given inputs.

|  |  |  |
| --- | --- | --- |
| **Input** | | **Output displayed** |
| **numAdults** | **numChildren** |  |
| 8 | 0 | Small group discount |
| 2 | 2 | Family discount |
| 12 | 0 | Large group discount |

Construct test data to meet the requirements set out in the table.

|  |  |  |
| --- | --- | --- |
| **Requirements** | Input | |
|  | **numAdults** | **numChildren** |
| A condition generating ‘regular pricing’ | 2 | 0 |
| Smallest group qualifying for a family discount | 2 | 1 |

### Activity 1.2.4 (homework)

Here is a program. *(Code available in ProgCode folder.)*

def cubed(aNum):

total = aNum \*\* 3

return total

aNum = 0

userNum = 0

while userNum == 0:

userNum = int(input('Enter a number: '))

if userNum < 10:

aNum = cubed(userNum)

else:

aNum = userNum

print(aNum)

|  |  |
| --- | --- |
| 1. Explain what the program does. | The program outputs the cube of a number that the user inputs if it is less than 10 |
| 1. State the name of the subprogram used in the code. | Cubed(aNum) |
| 1. Identify a selection. | If userNum < 10: |
| 1. Identify an iteration. | While userNum == 0: |
| 1. Identify all the lines on which the variable aNum is a local variable. | Lines 2, 3 |
| 1. List all the lines on which aNum is a global variable. | Lines 5 to 13 |
| 1. State the purpose of the ‘\*\*’ operator? | to the power of |

# Week 2

Lesson 1 activities

### Activity 2.1.1

A solid-state drive has a storage capacity of 250 GB.

What is this in:

|  |  |
| --- | --- |
| **megabytes** | 250\*1000 = 250,000 MB |
| **kilobytes** | 250,000 \* 1000 = 250,000,000 KB |
| **bytes** | 250,000,000 \* 10000 = 250,000,000,000 Bytes |
| **bits** | 250,000,000,000 \* 8 = 2,000,000,000,000 bits |

An image file has a size of 363,143,213 bits.

Construct an expression to convert its size to megabytes?

363,143,213 / (8\*1000\*1000)

A hard disc drive has a capacity of 2.8 TB.

Construct an expression to convert its size into bytes.

2.8 \* (1000\*1000\*1000\*8)

Construct an expression to calculate the size of a 24-bit image file displayed at a resolution of 400 x 250 pixels. Give your answer in kilobytes.

(480\*250\*24) / (8\*1000)

How many bits are there in 3 kilobytes?

3\*1000\*8 = 24,000 bits

Construct an expression to calculate the size of a 15.6 GB video in bits.

15.6 \* 1000 \* 1000 \* 1000 \* 8

### Activity 2.1.2

Use the Activity\_2.1.2\_Files sample files to see the difference compression makes to file size.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **File A** | **File B** | **File C** |
| How many characters does the file contain? |  |  |  |
| What type of content does the file contain? |  |  |  |
| What is its file size before compression? |  |  |  |
| What is its file size after compression? |  |  |  |
| What is its compression ratio (decompressed size/compressed file size)? |  |  |  |
|  | | | |
| Which file compresses the most? Explain why. | | | |

### Activity 2.1.3 (Homework)

1. What does file compression do?

Reduces the size of a file

1. Give **three** benefits of file compression.

Files use less space on your storage device

Takes less time to load the file

Takes less time to write on to a storage device

1. Outline the three strategies for compressing data described in the ‘Honey I shrunk the data’ game.

|  |  |
| --- | --- |
| **Compression method** | **How it works** |
| keep them all and make them small (lossless) | The data is compressed but can be reconstructed back to its original form. |
| squeeze them more and open the door’ (lossy) | Data is approximated and discarded. This method is irreversible, you cannot restore the original file. |
| ‘save the best and scrap the rest’ (superchannel) | Every single piece of data is discarded and only the one required is saved. |

1. WinZip is a well know file compression program. Find out what type of compression it uses.

WinZip uses lossless compression, specifically, the deflate algorithm.

Lesson 2 activities

### Activity 2.2.1 (homework)

Pearson Bicycle Hire collects data about many different aspects of the business. Complete this table by giving an appropriate data type for each variable.

|  |  |
| --- | --- |
| **Item** | **Data type** |
| Gender of an individual staff member | String |
| Whether a bike is still under guarantee | Boolean |
| The number of bikes out on hire at any one time | Integer |
| The mean number of hours a bike is hired for | Float/real |
| The charge per hour for hire of an adult bike | Float/real |

Each member of staff:

* has a 4-digit ID number, such as 3865 or 4722.
* Works a whole number of hours on the days they work.
* Works no more than 12 hours in a single day.

The business is open 7 days a week.

Draw a diagram of a data structure that shows the hours worked for each day of the week. Include data for at least **two** members of staff.

Whole number of hours every day

8 hours, 10 hours

< 12 hours

8 hours, 10 hours

7 days a week

Opening hours

Work hours

4-digit ID

3865, 4722

Staff

Business open 7 days/week

Week 3

Lesson 1 activities

### Activity 3.1.1

Use an image editor, such as pixlr’s photo editor online, to experiment with compression. Start with an uncompressed BMP file. Fill in this table.

|  |  |
| --- | --- |
| **Original BMP image** | **JPEG with 100% quality** |
| A close up of a logo  Description automatically generated | **A picture containing game  Description automatically generated** |
| **File size:** 1.02 MB | **File size:** 24.4 KB |
| Image quality: Clear and detailed. | Image quality: No noticeable loss in quality despite the heavy decrease in file size.  2.62 MB |
| **JPEG with 50% quality** | **JPEG with 25% quality** |
|  | A picture containing game  Description automatically generated |
| **File size:** 10.2 KB | **File size:** 7.77 KB |
| Image quality: Noticeable drop in quality especially when zooming into the image. | Image quality: A further drop in quality. The image is a lot less detailed and clear than the original. |

### Activity 3.1.2

Use a media player such as iTunes to experiment with music compression by encoding a music file using different bit rates.

|  |  |
| --- | --- |
| **Original uncompressed WAV file** | **Higher quality MP3  (around 192kbps)** |
| [Insert sound file here] | [Insert sound file here] |
| Sound quality: | Sound quality: |
| Number of kbps: | Number of kbps: |
| File size: | File size: |
| **High quality MP3  (around 160kbps)** | **Good quality MP3  (around 128kbps)** |
| [Insert sound file here]  Sound quality: | [Insert sound file here]  Sound quality: |
| Number of kbps: | Number of kbps: |
| File size: | File size: |

### Activity 3.1.3 (homework)

1. **A travel company has designed some brochures that contain images and text. It sends the brochures electronically to a printing company for them to be printed.**

**Explain why the travel company uses lossless compression to send the brochures.**

Using lossy compression for text will result in the loss of words/letters and meaning. For images, it will result in the loss of quality.

With lossless compression, the quality of the images and the meaning of the text will be preserved.

1. **A photographer produces ten 24-bit RGB images. He selects the JPEG compression format for the images. Compression normally reduces file size.**

**State two other characteristics of lossy compression.**

1. Loss of quality
2. Loss of metadata
3. **JPEG is a lossy image file format. MP3 is a lossy audio format. Write brief notes on how each of these works.**

JPEG averages out common colours in the image, reducing detail and the file size.

MP3 files average out similar frequencies in the audio file, reducing the detail of the audio and the file size.

Lesson 2 activities

Activity 3.2.1

Jan buys a lot of fruit in the market.

She needs a program to calculate the mean weight of the fruit she buys.

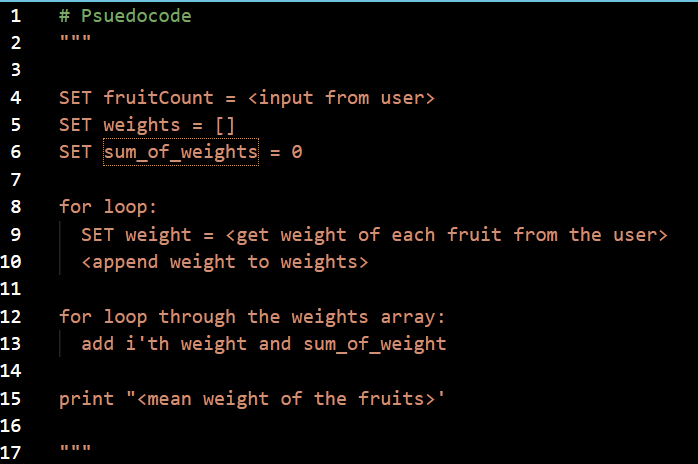
The program must prompt her to enter the number of items of fruit she has bought and then for the weight of each fruit. The program must calculate the mean weight of the fruit to two decimal places.

Design an algorithm in pseudocode for this program.

Select suitable test data for the algorithm and calculate the expected result it should produce.

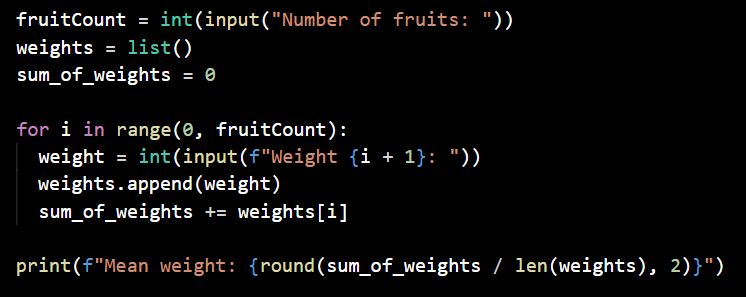
Code the algorithm in Python.

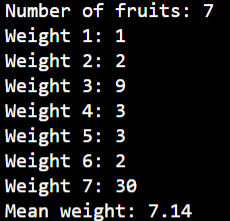
Test it using the test data you selected. Does the actual outcome match the outcome you expected? If not, why not?

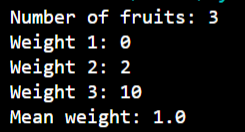
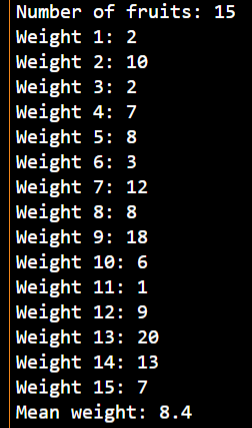
****

**Test data**

|  |  |  |
| --- | --- | --- |
| **Number of fruits** | **Weights** | **Expected result (mean)** |
| 3 | 0, 2, 10 | 4 |
| 15 | 2, 10, 2, 7, 8, 3, 12, 8, 18, 6, 1, 9, 20, 13, 7 | 8.4 |
| 7 | 1, 2, 9, 3, 3, 2, 30 | 7.14 |

**Code:**

**Results:**

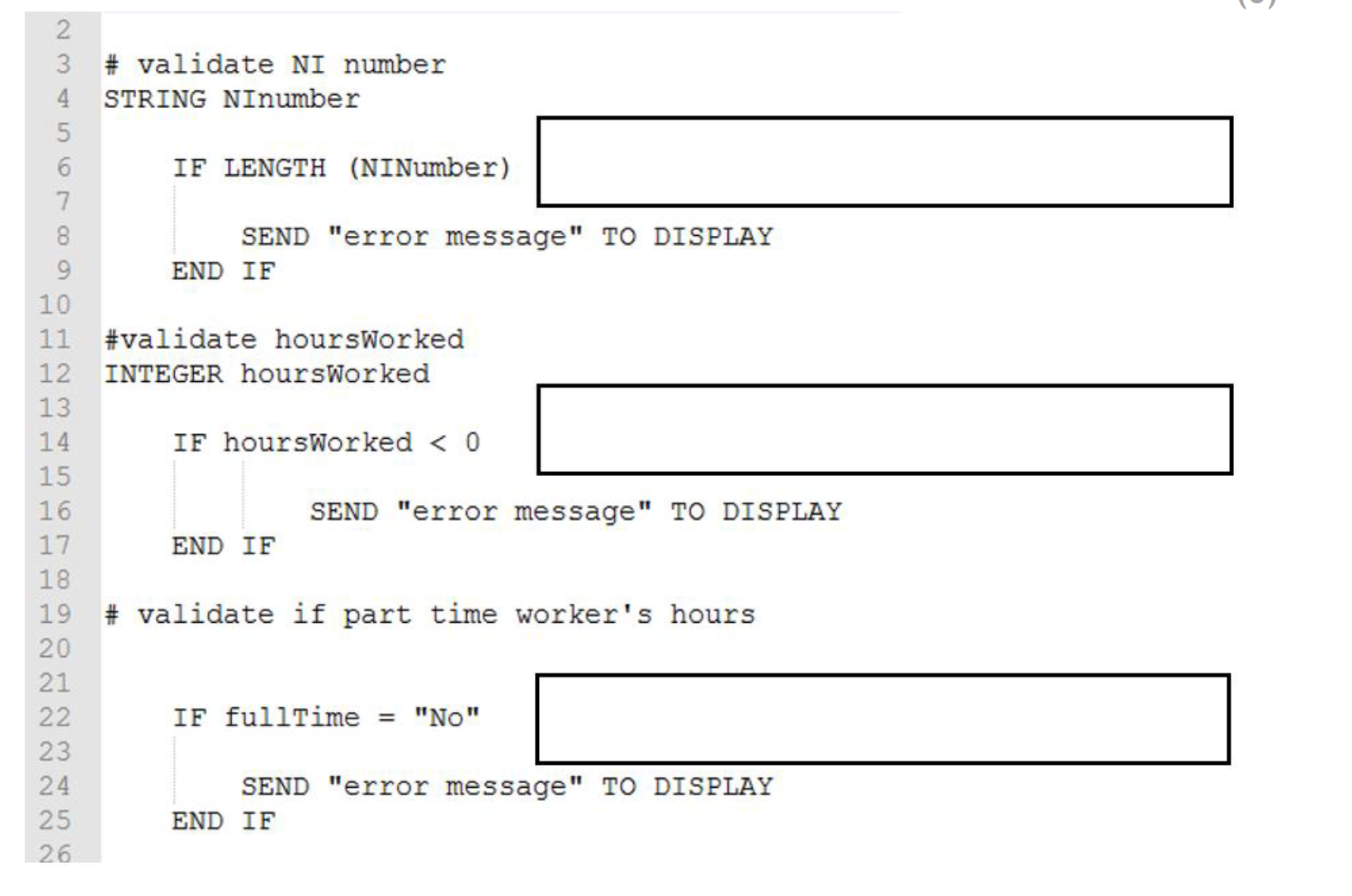
**** ****

Activity 3.2.2 (homework)

Pearson Bicycle Hire wants a program to calculate staff wages. Sample input data is shown below.

|  |  |
| --- | --- |
| **Input data** | **Notes** |
| National insurance (NI) number | Standard format: LL123456L |
| Full time | Yes or No  Full-time = maximum of 40 hours per week  Part-time = maximum of 20 hours per week |
| Hours worked | Integer  Hours worked in current week  Maximum of 10 hours overtime in one week for full-time only |
| Pay rate | Real  Hourly pay rate. |

Here is the pseudocode for an algorithm to validate the data.



AND hoursWorked < 40

OR hoursWorked > 40

!= 9

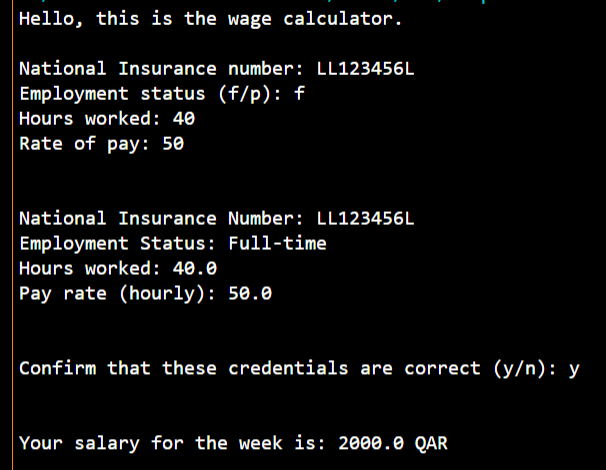
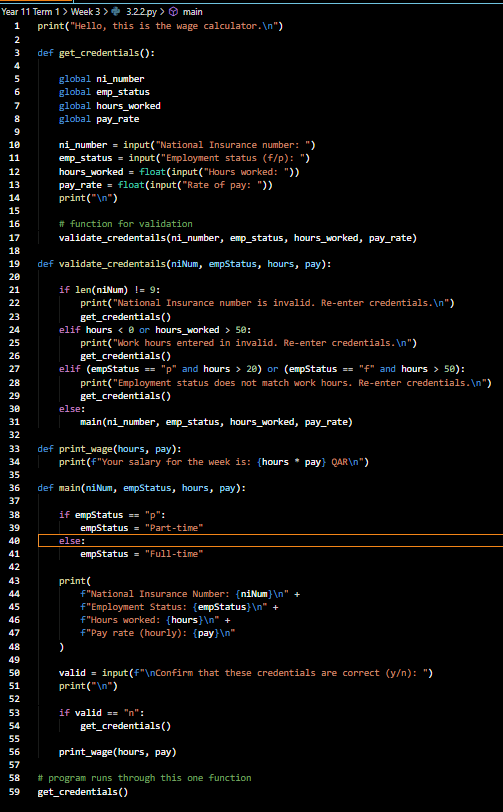
Fill in the boxes to complete the validation rules.

Create suitable error messages for each of the validation rules.

Convert the algorithm into program code.

Extend the program so that it:

* Displays a suitable welcome message for the wage calculator.
* Prompts the user to enter the details of a member of staff, i.e. their NI number, employment status (full-time or part-time), the number of hours they have worked and their rate of pay.
* Displays this information on the screen and ask the user to confirm that it is correct.
* If the information is incorrect the user should be prompted to re-enter the information.
* Display the correct information on screen.



Week 4

Lesson 1 activities

Activity 4.1.1

This series of binary numbers represents a compressed image.

|  |  |
| --- | --- |
| Line 1 | 00 0001 01 0011 00 1100 |
| Line 2 | 00 0001 01 0011 00 1100 |
| Line 3 | 00 0001 01 0011 00 0100 10 0010 00 0100 10 0001 00 0001 |
| Line 4 | 00 0111 10 0010 00 0100 10 0010 00 0001 |
| Line 5 | 01 0010 00 0101 11 0011 00 0011 10 0010 00 0001 |
| Line 6 | 01 0010 00 0011 11 0110 00 0001 10 0011 00 0001 |
| Line 7 | 00 0100 11 0010 01 0001 11 0101 10 0011 00 0001 |
| Line 8 | 00 0001 01 0001 00 0001 11 1001 10 0011 00 0001 |
| Line 9 | 00 0011 11 1001 10 0011 00 0001 |
| Line 10 | 00 0001 01 0001 00 0010 11 1000 10 0011 00 0001 |
| Line 11 | 00 0101 11 0110 00 0001 10 0011 00 0001 |
| Line 12 | 00 0111 11 0011 00 0011 10 0010 00 0001 |
| Line 13 | 00 0001 11 0001 00 0010 11 0001 00 0010 10 0010 00 0100 10 0010 00 0001 |
| Line 14 | 00 0001 11 0001 00 0001 11 0001 00 0100 10 0010 00 0100 10 0001 00 0001 |
| Line 15 | 00 0010 11 0001 00 1101 |
| Line 16 | 10 10000 |

The first pair of binary numbers represents a colour code as follows:

00 = blue 01 = white 10 = yellow 11 = green

The second set of binary numbers represents the run length of the colour. So, for example, 00, 1101 represents 13 blue pixels.

Assuming each of the squares in the grid below represents one pixel, shade in the squares. The first line of pixels has been done for you.

Line 1: 00 0001 01 0011 00 1100

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

### Activity 4.1.2

Use this image to complete these calculations.

|  |  |
| --- | --- |
| How many bytes are required to represent the uncompressed file?  Hint: number of squares in grid x 2 / 8 | Length \* width \* 2 \* / 8  16\*16\*2/8  = **64 bits** |
| How many bytes are required to represent the RLE encoded file?  Hint: number of codes x number of bits in each code / 8 |  |
| How much storage space have you saved? |  |

**Extension**

Imagine that an image contains:

* 1,024 pixels (a grid of 32 pixels by 32 pixels)
* 255 different colours (represented by 8 bits)

The image has been encoded using RLE, producing 116 binary codes comprising 8 bits (to represent the 255 colours) and 5 bits (to represent the colour run length).

Using this information, calculate the following.

|  |  |
| --- | --- |
| How many bytes are required to represent the uncompressed file? | 1024\*8/8  **1024 bytes** |
| How many bytes are required to represent the RLE encoded file? | 116\*8\*5/8  **580 bytes** |
| How much storage space have you saved? | 1024 – 580  **444 bytes** |

### Activity 4.1.3

|  |  |
| --- | --- |
| Encode these text strings using RLE. | |
| **Text string** | **Answer** |
| AAAABBBBBBBBBCADDDDEEFFFFFFFF | 4A9BCAA4DEE8F |
| ABCABCABCABCABCABCABCABCABCS | 9ABCS |
| BBGGYYAACCFFEEBBGGYYAACCFFEE | BBGGYYAACCFFEEBBGGYYAACCFFEE |
|  | |
| Which one compresses the most? | 9ABCS |
| Why is this? | Most amount of repeating data. |
| Describe in English the process the RLE calculator follows to encode a piece of text. | Repeated values are found in the piece of data.  Count the number of repeated characters or bits.  Replace the values by the number of times they are repeated. |

### Activity 4.1.4 (homework)

1. Explain how the RLE compression algorithm works.

1. Here is some data used to represent an image. Each pixel is encoded as a character.

AADACCEFABAAECFGBDGE

Explain why a RLE algorithm may not be appropriate for encoding this image.

3. Here is a partly completed RLE algorithm expressed as a written description.

Fill in the gaps to complete it.

1. Start with the first character in the string.

2. Write down the number 1.

3. Compare the first character with the next character on the right.

4. If they are the same, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

5. If they are not the same, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

6. Move on to the next character on the right.

7. Go back to step 2 and repeat until you reach the end of the string.

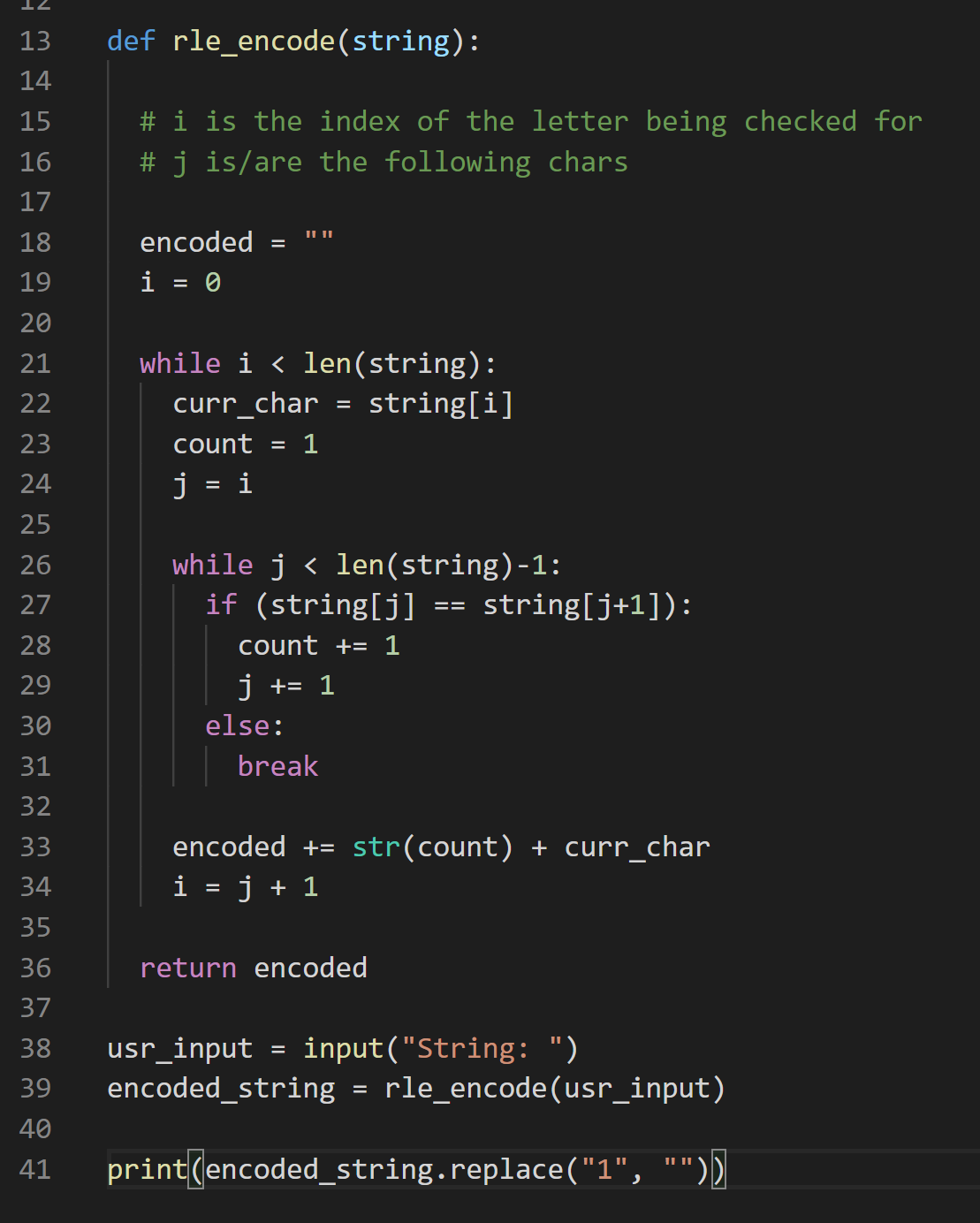
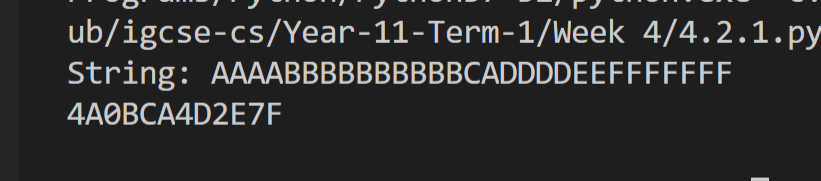
8. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Lesson 2 activities

### Activity 4.2.1

Implement the RLE compression algorithm in Python.

Test out your program using these data strings:

* AAAABBBBBBBBBCADDDDEEFFFFFFFF
* ABCABCABCABCABCABCABCABCABCS
* BBGGYYAACCFFEEBBGGYYAACCFFEE
* 
* **Activity 4.2.2 (homework)**

**Lossless and lossy compression summary sheet**

|  |  |
| --- | --- |
| Explain what is meant by lossless compression. | A type of data compression.  None of the original data is lost and the compressed file can be reconstructed back to its original form. |
| What type of data can be compressed using a lossless compression algorithm? | Any kind of data that has repeating values like text. |
| Describe how a lossless RLE algorithm works. | Looks for repeating values, adds a count to how many times that character repeats for. That character will be shown beside the count of how many time it repeats, thus, only 2 bytes will be needed. |
| Some lossless compression algorithms use a lookup table. Explain what the lookup table is for. | The table contains the repeating data with corresponding code. That code is used in the compressed file and that file is stored. |
| Explain how lossy compression differs from lossless compression. | Lossy compression results in a file that cannot be reconstructed back to its original form. The quality of the file is lost.  Lossy compression is also more effective at reducing file size than lossless compression. |
| Explain why lossy compression is usually used for media files. | Media files like videos take up a lot of space compared to other file types.  The difference in quality after compression may also be unnoticeable so the reduction in file size is worth the minimal loss in quality. |
| Outline the process of compressing an audio file using a lossy compression algorithm. |  |
| Outline the process of compressing a bitmap image using a lossy compression algorithm. |  |

### 

Week 5

Lesson 1 activities

### Activity 5.1.1

Complete the table by researching properties of secondary storage devices.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type of storage** | **Examples** | **Capacity** | **Speed** | **Portability**  How easy it is to carry around the storage device between different devices. | **Reliability**  How long a storage device would work before failing. | **Cost**  Price per unit of storage |
| Magnetic | Internal and external hard drive  Tape drive | Very large.  1-2 TB is common for internal hard disc drives.  A tape cartridge can hold around 7 TB of data (nowadays used mainly for archiving). |  |  |  |  |
| Optical | CD, DVD and Blu-ray drives | A CD stores 700 MB of data, a DVD 4.7 GB and a Blu-ray disc up to 50 GB |  |  |  |  |
| Solid state | Internal and external hard drives, flash memory sticks, SD cards | 250 - 512 GB internal hard drives are common for laptop machine.  1 TB USB memory sticks are common. |  |  |  |  |

### Activity 5.1.2 (extended homework)

Produce a presentation using an application of your choice (e.g. PowerPoint, Photo Story, Prezi, PowToon) explaining the three types of data storage (magnetic, optical and solid state).

This homework will extend over three lessons.

# Week 6

Lesson 1 activities

### Activity 6.1.1

Martin is an architect. He uses a desktop PC in the office and takes a laptop with him on site visits. He wants to back up his data on an external storage device but doesn’t know which one to choose.

Summarise the pros and cons of each of these storage devices:

* Magnetic hard drive
* Solid state drive (SSD)
* USB flash drive (memory stick)
* Optical drive
* Network attached storage (NAS) drive

Summarise security issues associated with each of these devices.

Recommend and justify which kind of external storage Martin should choose.

Lesson 2 activities

### Activity 6.2.1

Study this algorithm.

SET classMarks TO ………

SET total TO 0

SET highest TO 0

SET lowest TO 999

SET number TO LENGTH(classMarks)

FOR index = 0 TO number – 1 DO

SET total TO total + classMarks[index]

IF classMarks[index] > highest THEN

SET highest TO classMarks[index]

END IF

IF classMarks[index] < lowest THEN

SET lowest TO classMarks[index]

END IF

END FOR

SET mean TO total/number

SEND highest, lowest, mean TO DISPLAY

What does the algorithm do?

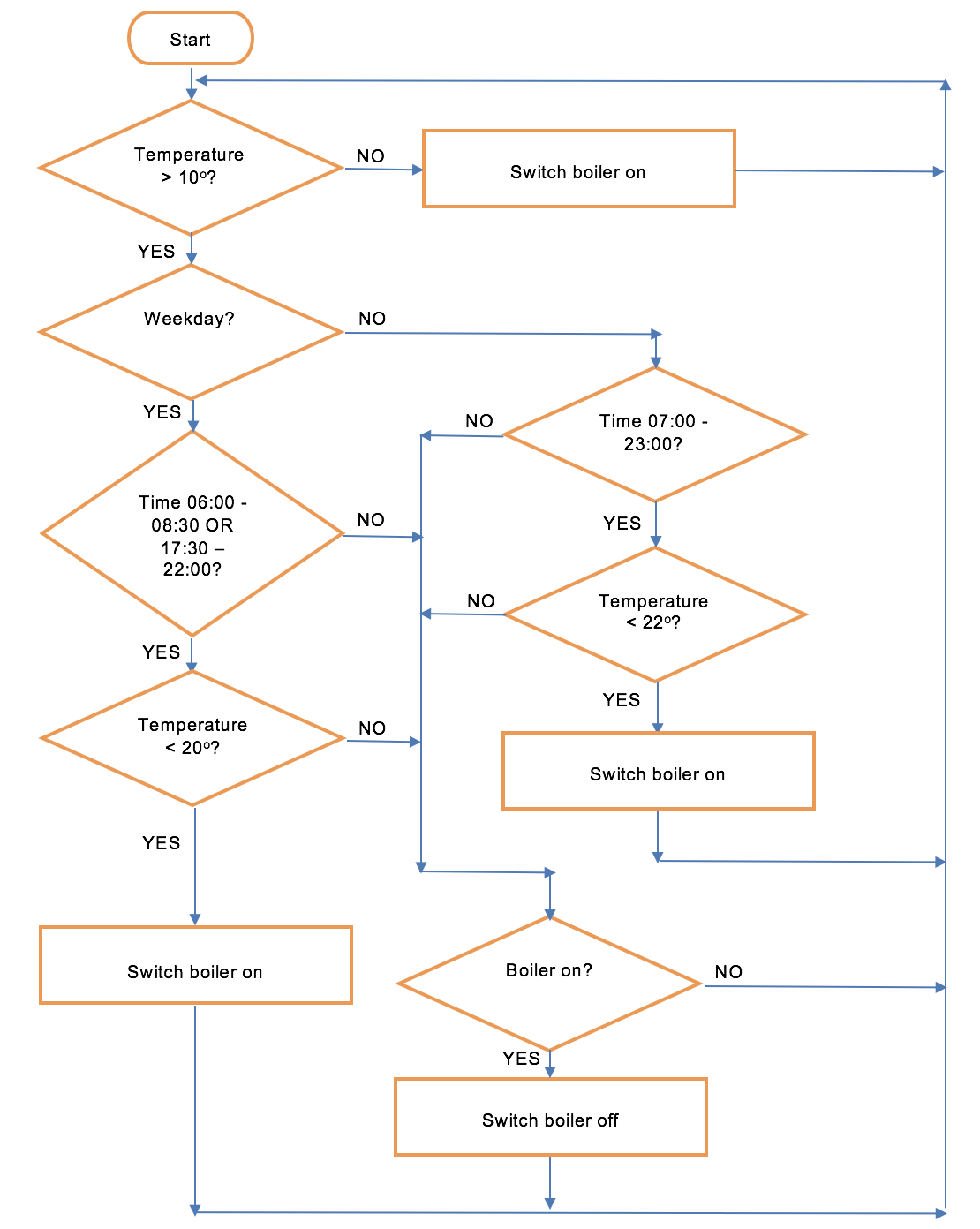
What would be the outcome of the algorithm if classMarks = [55, 62, 34, 78, 89, 43, 29, 94, 82, 79, 66, 67, 79, 22, 95, 19, 56, 75, 91, 18]

What relational operators does it use?

What type of iteration does it use, definite or indefinite?

### Activity 6.2.2

Study this algorithm for a program to control the central heating boiler in a house.



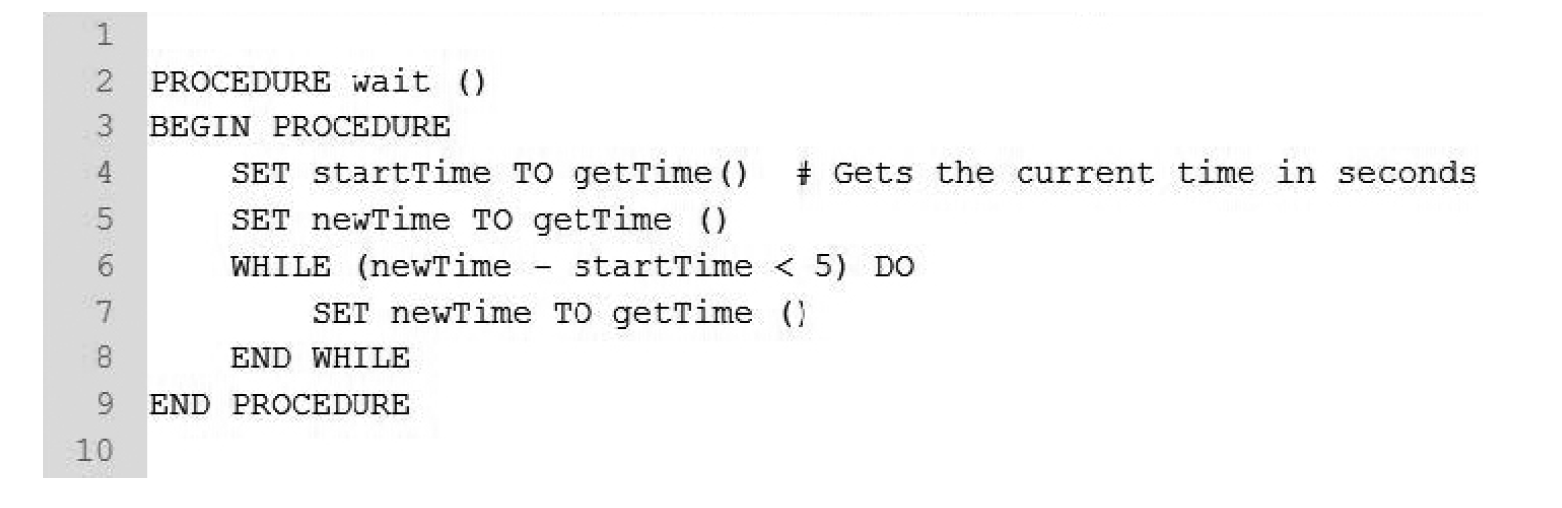
What temperature should it be in the house during the week?

What should the temperature be at the weekend?

How many selection constructs are there?

### Activity 6.2.3

Here is an algorithm that repeatedly checks the time. When the procedure is called, the calling code waits for five seconds.



### Amend the procedure so that the length of the wait can be specified when the procedure is called.

### Activity 6.2.4

Sales staff want to encourage customers to renew their annual magazine subscriptions.

Details of customers’ subscriptions are stored in a subscription file. The month the subscription starts is recorded.

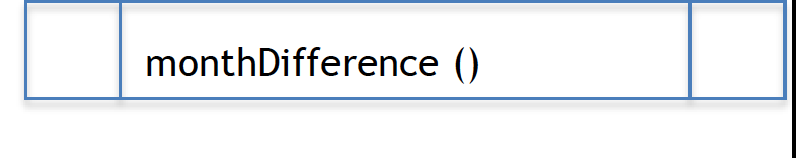
One month before a subscription expires, the sales team send the customer an email inviting them to renew their subscription.

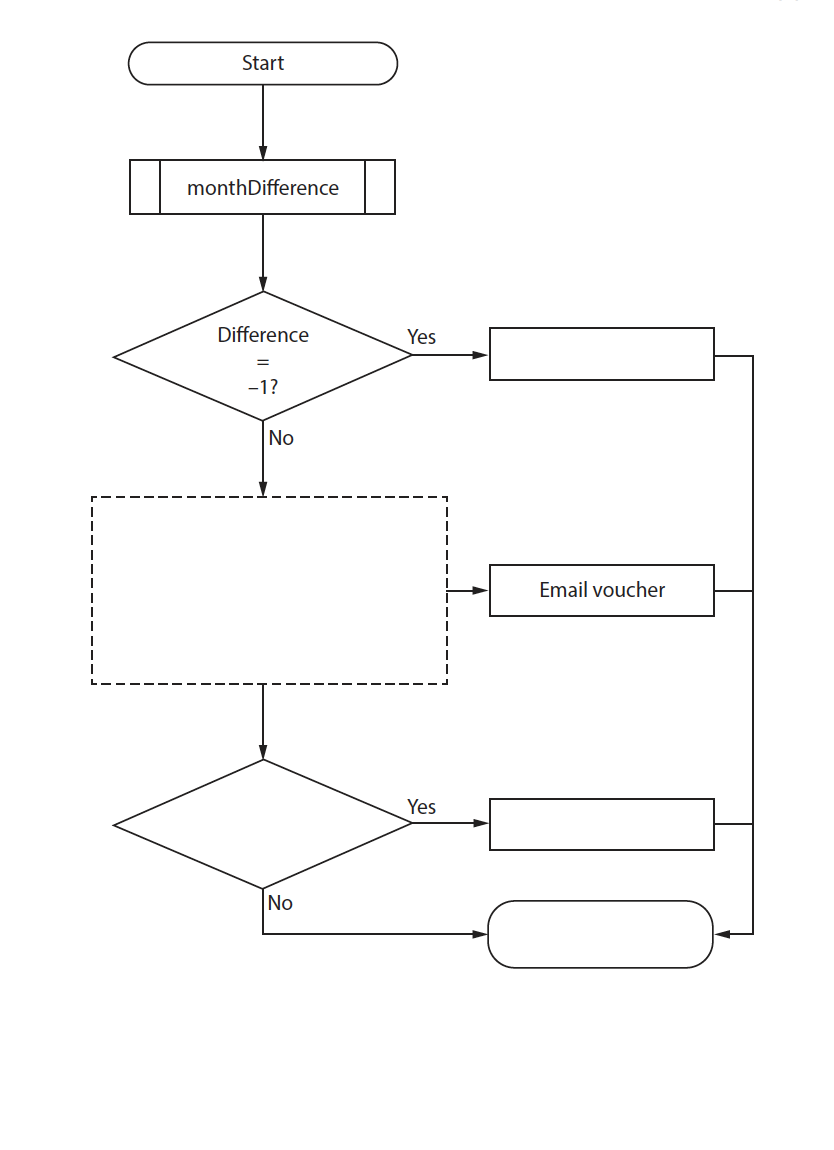
If the subscription is not renewed within 3 months after its expiry, the sales team send the customer an email with a voucher as an incentive to renew.

If the subscription is not renewed within 6 months after its expiry, the customer record is deleted from the file.

A function called monthDifference() returns the number of months between the current month and the expiry month.

The function is shown as:



Complete the flowchart to show this process.

### Activity 6.2.5 (homework)

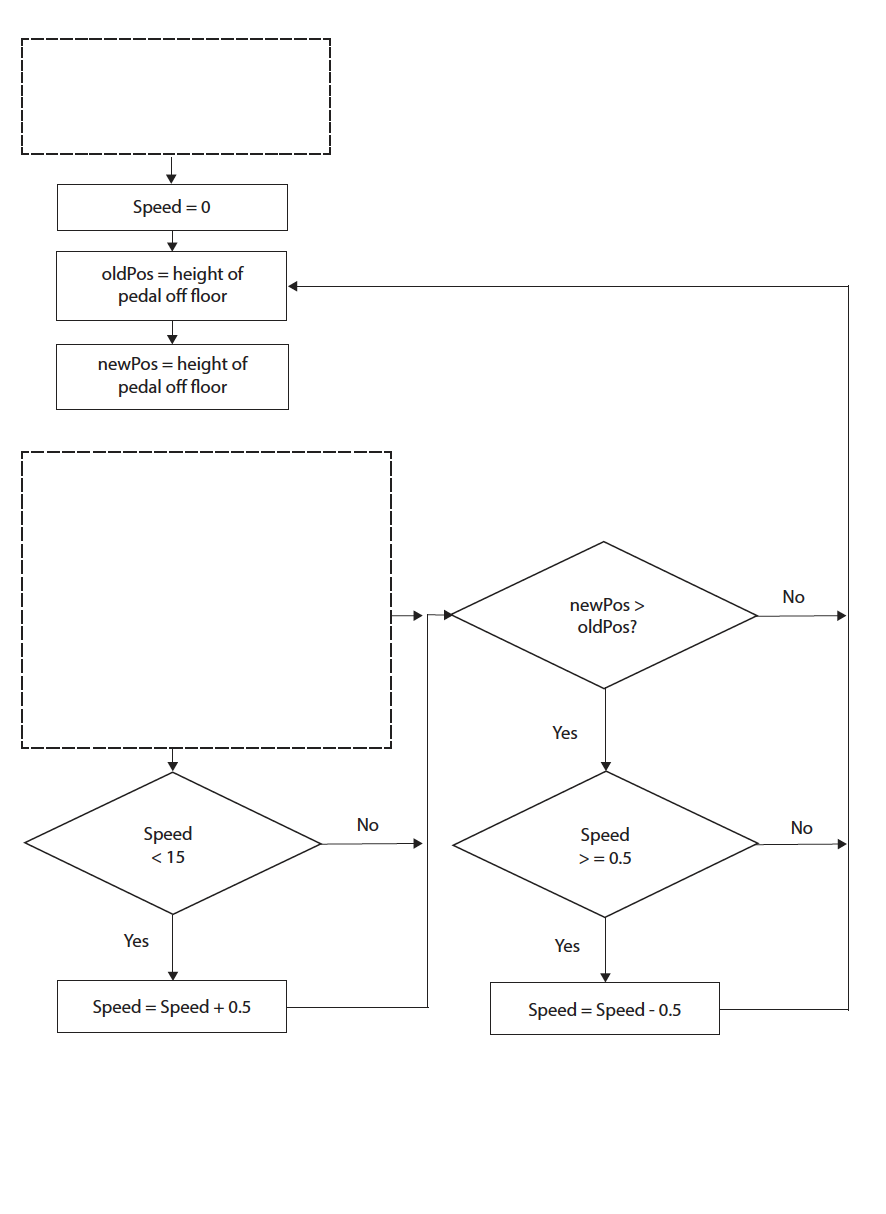
Sparky Autos is a family owned business. It offers visitors the opportunity to drive small electric cars around a race track.

Each car has a driver-operated pedal to make it go forward.

The closer the pedal is to the floor, the faster the car goes.

The further the pedal is from the floor, the slower the car goes.

Each car has its speed limited to 15 kilometres per hour.

Complete the flowchart to show this process.

# Week 7

Lesson 1 activities

### Activity 7.1.1

The following cards should be printed and laminated if desired to make a simple card sort activity. One set is required per group of students in a class.

Ask students to sort the cards into two piles: questions that have answers that are advantages for cloud storage and questions that have answers that are disadvantages for cloud storage.

To increase challenge, ask students to rank each pile, by putting the ‘best’ advantage at the top and so on. Ask them what they mean by ‘best’. A further challenge could be added by asking students to consider the cons, in terms both of risk and impact (see solutions for example).

|  |  |
| --- | --- |
| Where can I access my data from? | Which computing devices can I use to access my data? |
| Is my data backed-up easily? | Will my data stay private? |
| How much does it cost to start using? | How easy is it to upgrade (increase capacity)? |
| What happens if a disk has a hardware fault? | Can I change service provider easily? |
| What happens if the service provider goes bankrupt? | What if my network is ‘down’? |
| What happens if the service provider servers go down? | Will very large files cause problems? |
| What is the cost to upgrade (increase capacity)? | Will I still ‘own’ my data? |

### Activity 7.1.2

The senior leadership team at a school is discussing different storage media.

The administrative staff want to use the ‘cloud’. The technical team wants to use servers located on the school grounds.

Compare storing data in the ‘cloud’ with storing data on hard discs connected to the school’s servers.

Lesson 2 activities

### Activity 7.2.1

Your task is to produce a computer-based online safety quiz for kids.

Players will be presented with a set of questions to test their knowledge of online safety.

**Requirements**

* The quiz must have a text based interface.
* Players must enter their name before starting the quiz.
* Five questions on online safety must be displayed, one after another, on the screen.
* Questions must be presented in a different order each time the quiz is run.
* Each question must be shown with three possible answers, one of which must be correct.
* Players must be able to select an answer.
* A correct answer scores 4 points.
* Once the quiz has been completed, the player’s name and score should be displayed on screen.

A set of possible quiz questions is available in the file quizQuestions.txt, but if you prefer, you can create your own.

Week 8

Lesson 1 activities

### Activity 8.1.1

Research the following questions:

IP addresses in the form 192.168.11.1 are called IP version 4 addresses. Each number is a single byte so the entire address is 4 bytes.

1. How many possible IP (version 4) addresses are there?

There are so many devices on the internet (about 10 billion) that we have run out of addresses.

Recently IP version 6 has been introduced, which uses 16 bytes for each address.

1. How many possible IP (version 6) addresses are there?
2. Find out how many people there are on Earth.
3. Is that enough for each person on the planet to have their own IP address?
4. How do you think this will affect the number of IP addresses needed? Do you think IP version 6 will do?
5. Find out what is meant by the ‘Internet of Things’.

### Activity 8.1.2

Split the class into two halves, one on each side of the room, with one student in the middle to act as the router.

Give each student an IP address card.

Explain to the router that ‘devices’ on one side of the room have addresses starting 192.168 and the other side 10.10.

Get each of the ‘devices’ to write a message to another ‘device’, starting with their own IP address (e.g. FROM: 192.168.11.3) and the recipient’s IP address (e.g. TO: 10.10.10.5).

The completed messages should be thrown into a pile on their side of the room.

Next each ‘device’ should look for any messages on their side of the room addressed TO themselves and take them out.

The router must then pick up the remaining messages and pass them across to the other side.

Possible simple IP address cards to cut out and use:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 192.168.11.10 | 192.168.11.16 | 192.168.11.22 | 10.10.10.13 | 10.10.10.19 |
| 192.168.11.11 | 192.168.11.17 | 192.168.11.23 | 10.10.10.14 | 10.10.10.20 |
| 192.168.11.12 | 192.168.11.18 | 192.168.11.24 | 10.10.10.15 | 10.10.10.21 |
| 192.168.11.13 | 192.168.11.19 | 10.10.10.10 | 10.10.10.16 | 10.10.10.22 |
| 192.168.11.14 | 192.168.11.20 | 10.10.10.11 | 10.10.10.17 | 10.10.10.23 |
| 192.168.11.15 | 192.168.11.21 | 10.10.10.12 | 10.10.10.18 | 10.10.10.24 |

### Activity 8.1.3 (homework)

Use a visual trace route tool to find out the ‘hops’ to your favourite websites and to take a screen shot of the most interesting or surprising route.

Lesson 2 activities

### Activity 8.2.1

Here is the beginning of a test plan for the quiz. Add any other tests you think are needed along with sample test data and expected results.

| **Test numb** | **Description of test** | **Test data** | **Expected result** |
| --- | --- | --- | --- |
| 1 | Can the player enter their name? | ‘Andy Williams” | ‘Welcome to the quiz Andy Williams’ should be displayed on the screen. |
| 2 | Are instructions displayed correctly? | N/A | The rules of the game should be displayed clearly on the screen, i.e.  You will be asked a series of questions about online safety.  Select the correct answer by typing in the number when prompted.  Each correct answer will be awarded 4 points. |
| 3 | Are quiz questions and answers displayed correctly? | N/A | Questions should be displayed on the screen in this format:  Question  Q1 Answer1  Q2 Answer2  Q3 Answer3  Please enter 1, 2 or 3: |
| 4 | Can the player select an answer? | 1  2 | Accepted and player told whether or not answer is correct. |
| 5 | Are inappropriate answers rejected? | ‘s’  12 | Rejected and player asked again to input a number between 1 and 3. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Week 9

Lesson 1 activities

### Activity 9.1.1

Answer these questions about the video.

|  |  |
| --- | --- |
| **Question** | **Answer** |
| What chewed through Andrew’s internet? |  |
| What transmission medium is used for submarine cables to transfer bits under the oceans? |  |
| What does the router at 60 Hudson Street in New York do? |  |

Week 10

Lesson 1 activities

### Activity 10.1.1 (homework)

Explain the difference between the internet and the WWW.

Write definitions for each of these components of the WWW.

|  |  |
| --- | --- |
| **Component** | **Definition** |
| URL |  |
| ISP |  |
| HTTP |  |
| HTTPS |  |
| HTML |  |

Week 11

Lesson 1 activities

### Activity 11.1.1

|  |
| --- |
| Crack this code! |
| WKH WUHDVXUH LV KLGGHQ XQGHU WKH SDOP WUHH |
|  |

Hint: H = E and W = T

### Activity 11.1.2 (homework)

|  |  |
| --- | --- |
| Who uses encryption and what do they use it for what? | |
| **Users of encryption** | **What they use encryption for** |
| Businesses |  |
| Individuals |  |
| Governments |  |
| E-traders |  |
| The military |  |

Lesson 2 activities

### Activity 11.2.1 (homework)

1. Annotate your quiz program code to identify:

* A user-defined subprogram
* A built-in subprogram
* A library subprogram
* A subprogram that uses input parameters
* A subprogram that does not use input parameters
* A local variable
* A global variable

2. List all the benefits of using subprograms.

3. Computer games and quizzes make use of abstractions.

A computer game involves the use of two dice. A subprogram called ‘roll’ simulates the rolling of a die.

Explain why ‘roll’ is an abstraction.

Week 12

Lesson 1 activities

### Activity 12.1.1

|  |  |  |
| --- | --- | --- |
| Use a Caesar cipher to decode and encode these messages. | | |
| **Plain text** | **Shift** | **Encrypted text** |
| THE ENIGMA MACHINE WAS INVENTED BY THE GERMANS | +3 |  |
| COLOSSUS WAS THE WORLD’S FIRST DIGITAL COMPUTER | +4 |  |
|  | +5 | YMJ HFJXFW HNUMJW NX FS JCFRUQJ TK WTRFS NSLJSZNYD |
|  | -3 | QEB HBV FP EFAABK RKABO QEB CILTBO MLQ |

### Activity 12.1.2

With a partner, produce a step-by-step guide to using a Caesar cipher to produce an encoded message.

### Activity 12.1.3 (homework)

Produce a step-by-step guide to decrypting messages that have been encoded using a Caesar cipher.

Lesson 2 activities

### Activity 12.2.1

Run the program 12\_2\_1.py.

Try different messages and different keys. What happens when a message includes a character that can’t be encrypted? What happens when the character to be encrypted is a ‘z’?

### Activity 12.2.2

Study the program code.

What arguments does the function caesar\_cipher\_encrypt use?

What does the function return?

What is the purpose of the array characters?

What is the purpose of the inList variable?

InList is a local variable. List all the other local variables.

List all the global variables.

Explain what is meant by the scope of a variable.

What happens if the message to be encrypted includes a character other than a letter of the alphabet?

How would the program need to be amended to allow it to encrypt the numbers 0 to 9 as well as letters of the alphabet?

### Activity 12.2.3

A Caesar cipher is a simple encryption algorithm based on shifting.

Explain whether a shift of +7 followed by a shift of -2, is more secure than a single shift when applied to the word ‘pink’.

You should include a diagram in your answer.

### Activity 12.2.4 (homework)

1. Explain what a cipher is.
2. Explain, using an example, how the Caesar cipher works.
3. A brute force attack can be used to crack a message encoded using the Caesar cipher. Explain what is meant by a ‘brute force attack’.
4. The Caesar cipher is an example of a substitution cipher. Explain what this means.
5. Produce an algorithm expressed as a flowchart for a program to encrypt a message using a Caesar cipher.

Week 13

Lesson 1 activities

### Activity 13.1.1

A Caesar cipher has been used to encrypt these secret messages.

Work with a partner to ‘crack’ at least two of them, using letter frequency analysis to speed up the process you.

**Message 1**

qiix ex wmb fc xli kexi

**Message 2**

wkh sodqv duh klgghq lqvlgh wkh juhhq mdgh gudjrq

**Message 3**

phhw dw wkh juhhq gudjrq wrqljkw

**Message 4**

aol hnlua dpss il jhyyfpun h ibujo vm ylk yvzlz

Can you suggest a way of producing a substitution cipher that can’t be cracked by analysing letter frequency?

### Activity 13.1.2 (homework)

1. Explain how the Pigpen cipher got its name.
2. Explain, using an example, how the Pigpen cipher works.
3. Both the Pigpen and Caesar and ciphers are mono-alphabetic substitution ciphers. Explain what this means.
4. Using letter frequency analysis, it’s possible to ‘crack’ a message encrypted using a Caesar cipher relatively quickly. Explain what is meant by ‘letter frequency analysis’.
5. Describe two ways in which the Pigpen cipher can be made more difficult to ‘crack’.

Lesson 2 activities

### Activity 13.2.1

### In pairs open the Caesar cipher program (Activity12\_2\_1.py).

### Amend the caesar\_cipher\_encrypt function so that it encrypts the numbers 0 to 9 as well as the letters of the alphabet.

### Add comments to the function to explain how it works.

### Activity 13.2.2

Rewrite the caesar\_cipher\_encryptfunction so that it can both encrypt and decrypt messages.

Add a menu to the main program that will allow the user to select one of three options, i.e. encrypt a message, decrypt a message or exit the program.

### Activity 13.2.3 (homework)

Study chapter 14 of ‘Invent with Python’. It takes you through the code for a Caesar cipher that makes use of the built-in Python functions chr() and ord(). Study the explanation of how the code works. Type in the source code and add in the brute force mode.

https://inventwithpython.com/chapter14.html

Week 14

Lesson 1 activities

### Activity 14.1.1

1. Use the Vigenère cipher with the keyword ‘mermaid’ to encode these messages. Omit the spaces between words to make the encrypted messages harder for hackers to crack

**Message 1**

meet at six on the beach =

**Message 2**

the plans are hidden inside the sleeping buddha =

**Message 3**

danger tonight beware =

**Message 4**

the decoy is wearing a purple wig =

Now encode the same messages again using a different keyword.

**Message 1**

meet at six on the beach =

**Message 2**

the plans are hidden inside the sleeping buddha =

**Message 3**

danger tonight beware =

**Message 4**

the decoy is wearing a purple wig =

1. Explain the purpose of the keyword in the Vigenère cipher.
2. Write down a set of step-by-step instructions for encoding a message using the Vigenère cipher. This YouTube video might help you: https://www.youtube.com/watch?v=K1SuiUu4kG0

### Activity 14.1.2

1. This message has been encrypted using a Vigenère cipher and the key word ‘chocolate’.

Decrypt the message:

guqtmatbsprsgdddtxczseice

1. Write down a set of set of step-by-step instructions for decrypting Vigenère encrypted messages.

### Activity 14.1.3 (homework)

Study the Vigenère cipher program code (Activity14\_1\_13.py). Add comments to explain how it works.

How would you amend the code so that spaces and punctuation marks can be included in the message to be encrypted?

Why isn’t it a good idea to do this?

Week 14

Lesson 2 activities

### Activity 14.2.1

1. Explain, using an example, how the Rail Fence cipher works.
2. The Rail Fence cipher is a transposition cipher whereas the Caesar, Pigpen and Vigenère ciphers all use some form of substitution. Explain how a transposition cipher works.
3. Explain why the number of the Rail Fence cipher is relatively easy to crack.
4. Complete this summary table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Cipher** | **Type** | **Key** | **Security rating** |
| Pigpen |  |  |  |
| Caesar |  |  |  |
| Vigenère |  |  |  |
| Rail Fence |  |  |  |