CODE CHALLENGE WORKED EXAMPLE: MASTERMIND

PROVISIONAL

For each challenge, solve it using:

- A flowchart

- Pseudocode (see GCSE Pseudocode Guide)

- Programme code (any high level language will do)
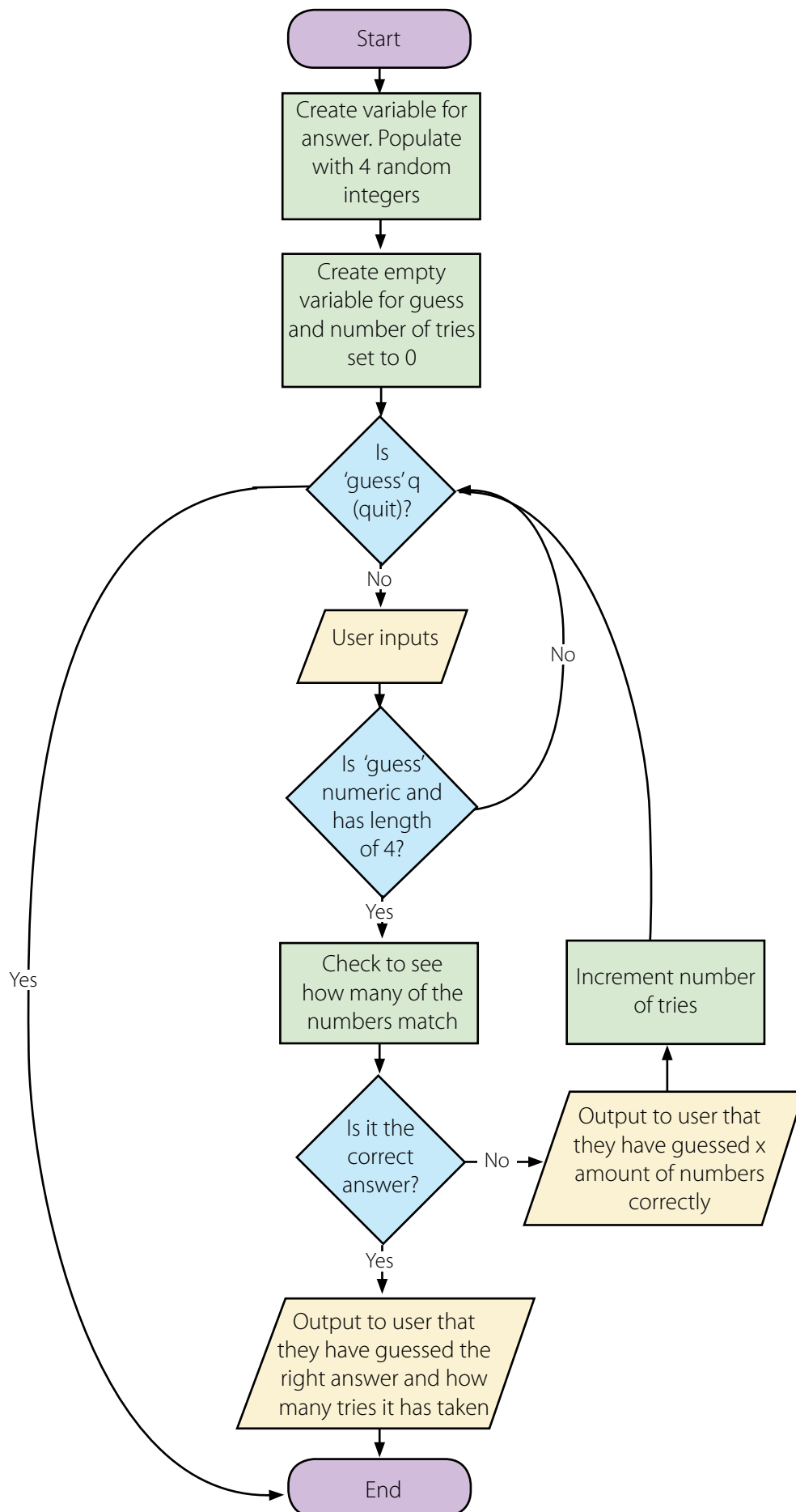
The challenge: Mastermind

Generate a random 4 digit number. The player has to keep inputting 4 digit numbers until they guess the randomly generated number. After each unsuccessful try it will say how many numbers they got correct, but not which position they got right. At the end of the game it will congratulate the user and say how many tries it took.

OCR
Oxford Cambridge and RSA

The flowchart

```
                    Start
                      │
                      ▼
          ┌────────────────────────┐
          │ Create variable for    │
          │ answer. Populate       │
          │ with 4 random          │
          │ integers               │
          └────────────────────────┘
                      │
                      ▼
          ┌────────────────────────┐
          │ Create empty           │
          │ variable for guess     │
          │ and number of tries    │
          │ set to 0               │
          └────────────────────────┘
                      │
                      ▼
                 ◇ Is 'guess' q (quit)? ◇
                      │ No
                      ▼
               ╱ User inputs ╱
                      │
                      ▼
            ◇ Is 'guess' numeric and
              has length of 4? ◇ ── No
                      │ Yes
                      ▼
          ┌────────────────────────┐        ┌──────────────────────┐
          │ Check to see           │        │ Increment number     │
          │ how many of the        │        │ of tries             │
          │ numbers match          │        └──────────────────────┘
          └────────────────────────┘                  ▲
                      │                                │
                      ▼                    ╱ Output to user that ╱
                 ◇ Is it the          ╱ they have guessed x ╱
                   correct  ◇ ── No ──▶╱ amount of numbers  ╱
                   answer? ◇           ╱ correctly          ╱
                      │ Yes
                      ▼
          ╱ Output to user that      ╱
          ╱ they have guessed the    ╱
          ╱ right answer and how     ╱
          ╱ many tries it has taken  ╱
                      │
                      ▼
                    End
```

(Is 'guess' q (quit)? — Yes → End)

Pseudocode

```
correctAnswer = ''


//generate number
for i=1 to 4
    tempChar = str(random.randint(0,9))
    correctAnswer = correctAnswer + tempChar


guess = ''
attempts = 0


while guess !="q"
    choice = input("Please enter your 4 digit number or q to quit :")

    //Make sure it is 4 digits long and contains only digits
    if guess.isnumeric()and len(guess) == 4
       numCorrect = 0
      //How many numbers did they guess correctly?
       for i=0 to 3
           if guess[i] == correctAnswer[i]
               numCorrect += 1


    //Have they guessed it?
    if(choice == correctAnswer)
        print("Well done... That took you "+ str(attempts) + " attempts!")
        break
    else:
        print("That is not right. You guessed "+ str(numCorrect) +
        " numbers correctly this time")
        attempts += 1
```

Program code (with comments):

```python
import random
correctAnswer = ''
#generate number
for i in range (4):
    tempChar = str(random.randint(0,9))
    correctAnswer = correctAnswer + tempChar

guess = ''
attempts = 0

while guess !="q":
    choice = input("Please enter your 4 digit number or q to quit :")

    #Make sure it is 4 digits long and contains only digits
    if guess.isnumeric()and len(guess) == 4:
        numCorrect = 0
        #How many numbers did they guess correctly?
        for i in range(0,3):
            if guess[i] == correctAnswer[i]:
                numCorrect += 1
        #Have they guessed it?
        if(choice == correctAnswer):
            print("Well done... That took you "+ str(attempts) +
            " attempts!")
            break;
        else:
            print("That is not right. You guessed "+ str (numCorrect) +
            " numbers correctly this time")
            attempts += 1
```