

Computational thinking

Q1. The identifier `plants` is used for an array of values.

`len(plants) // 2` is used to find the index position of the middle item in `plants`

Explain **one** reason why integer division, rather than division, is used to do this.

(2)

Q2. A constant is shown here on line 3.

```
2 # Prototype for the main swimming pool
3 MAX_CAPACITY = 120      # Maximum number of swimmers
4
5 numAdult = 14          # Current number of adults swimming
6 numChild = 73         # Current number of children swimming
```

Programmers can use all capitals to show that a value is a constant.

Explain **one** reason why programmers use signals indicating a value is a constant, rather than repeating the same fixed value throughout an algorithm.

(2)

Q3. Programmers use different types of operators in their programs. Name the **type** of operator for each example.

(3)

`+`, `*`

`AND`, `NOT`

`==` `!=` `<=` `>=` `<` `>`

Q4. An algorithm is designed for a 'guess the roll of a dice' game. The player decides the number of sides on a single dice, between 6 and 12, and guesses the number rolled.

(i) Name **two** inputs required by the algorithm.

(2)

- 1
- 2

(ii) The player of the game can have three guesses. The variable `roll` is set to 1 at the start of the game. One is added to `roll` after each guess.

Identify the statement with the correct relational operator used to check whether the player can have another guess.

(1)

- A** `roll = 3`
- B** `roll == 3`
- C** `roll > 3`
- D** `roll <= 3`

ANSWERS

1. Integer division is used when the programmer doesn't want the result to be a fraction (decimal). This occurs in situations where having a fraction makes no sense....eg. half a book, half a person or halving a list. **Remember** integer division uses 2 divide signs: // the other one is **modulus** division which returns the remainder as a whole number and in Python uses the % sign.
2. Using constants makes the program easier to read and maintain due to meaningful constant names. Also if the value needs to change in the future (for example, adjusting the maximum length of a string), updating the constant in one place automatically updates the change throughout the entire program, reducing the risk of errors.
3. +, */ these are arithmetic operators
AND, NOT, OR these are logical operators (Boolean operators)
== != <= >= < > these are relational operators (comparison operators)
4. Inputs would be the number of sides on the dice and a player's guess
5. C greater than 3 because the starting value was 1 so it would be 2 after the first guess, 3 after the second guess and 4 after the 3rd and final guess.