

# Computing Studies

Software Design  
and Development

## Programming Challenges



&



Buckhaven High School

Version 1



## Contents

Page 1	How to use this booklet.
Page 2	Programming Knowledge Required for Each Set of Problems.
Page 3	Input & Output Problems (String and Numeric Variables)
Page 6	Selection Problems (IF, ELSE, ELSE IF)
Page 8	Iteration (Unconditional Loops)
Page 11	Iteration (Conditional Loops)
Page 13	Arrays (Storing Multiple Examples of the Same Data)
Page 15	Predetermined Functions

## How to use this booklet

This booklet contains dozens of small programming problems designed for pupils to hone their programming skills. It should be used in conjunction with a workbook or syntax reference for the programming language you are using in class.

Make sure you save each program in an organised way as later problems occasionally refer back to earlier ones. If an earlier program has not been saved you will have to enter the code for it again.

	National 4 	National 5 
<b>Computational Constructs</b>	Exemplification and implementation of the following constructs: <ul style="list-style-type: none"> <li>expressions to assign values to variables</li> <li>expressions to return values using arithmetic operations (+, -, *, /, ^)</li> <li>execution of lines of code in sequence demonstrating input - process - output</li> <li>use of selection constructs including simple conditional statements</li> <li>iteration and repetition using fixed and conditional loops</li> </ul>	Exemplification and implementation of the following constructs: <ul style="list-style-type: none"> <li>expressions to assign values to variables</li> <li>expressions to return values using arithmetic operations (+, -, *, /, ^, mod)</li> <li>expressions to concatenate strings and arrays using the operator</li> <li>use of selection constructs including simple and complex conditional statements and logical operators</li> <li>iteration and repetition using fixed and conditional loops</li> <li>pre-determined functions (with parameters)</li> </ul>
<b>Data Types and Structures</b>	string numeric (integer) variables graphical objects	string, character numeric (integer and real) boolean variables 1-D arrays
<b>Algorithm Specification</b>		Exemplification and implementation of algorithms including <ul style="list-style-type: none"> <li>input validation</li> </ul>

**Programming Knowledge Required for Each Set of Problems**

**Pages 3 to 5 Input & Output Problems (String and Numeric Variables)**

- Ask user to enter an integer and store the value entered
- Ask user to enter a real number and store the value entered
- Ask the user to enter a string and store the value entered
- Display text or string variables in an output window
- Display numeric variables (integer & real) in an output window
- Combine variables, text and numbers in one output statement
- Perform calculations with two or more numbers and store the result
- Calculations include +,-,/,\*
- Concatenate two or more strings or string variables
- Convert an integer variable to a string
- Convert a string variable to an integer

} input

} output

} calculations

} string manipulation

**Pages 6 to 7 Selection Problems (IF, ELSE, ELSE IF)**

- Control whether lines of code are executed or not (IF)
- Control which of two blocks of code are executed (IF, ELSE)
- Use multiple decisions to control the execution of lines of code (IF, ELSE IF, ELSE)
- Use the following operators: = > < <= >= !=
- Use the following logical operators: AND OR NOT

} making decisions

} operators

**Pages 8 to 12 Iteration (Unconditional and Conditional Loops)**

- Use a loop to repeat lines of code a set number of times
- Use a variable to control the number of times a loop is executed
- Use the loop variable as a counter within a loop
- Use simple conditions to control how many times code is repeated
- Use complex conditions to control how many times code is repeated
- Use a counter to store the number of times a conditional loop has executed

} looping a fixed number of times

} looping an unknown number of times

**Pages 13 to 14 Arrays (Storing Multiple Examples of the Same Data)**

- Use an array to store multiple strings
- Use an array to store multiple numeric values
- Use a loop to control the storage of data in an array
- Use a loop to control the use of array data in a calculation
- Use a loop to output data from an array.
- Use a conditional statement and a loop to control which data is displayed in an array.

} storing multiple values

} using loops when storing multiple values

**Pages 15 to 16 Predetermined Functions**

- Use predetermined functions to:
- Generate a random number
- Convert a character to an ASCII value
- Calculate the modulus of two numbers
- Change upper case characters to lower case
- Calculate the length of a string
- Convert an ASCII value to a character
- Round a number to given decimal places
- Change lower case to upper case

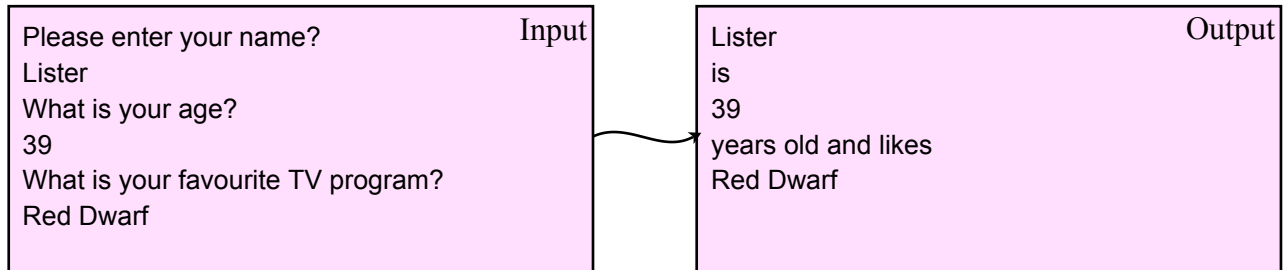
## Input & Output Problems (String and Numeric Variables)

### 1. Three In, Three Out

N4

Write a program that will allow a user to enter their name (string), their age (integer) and their favourite TV program (string). The program will then display the information entered and some additional text on separate lines.

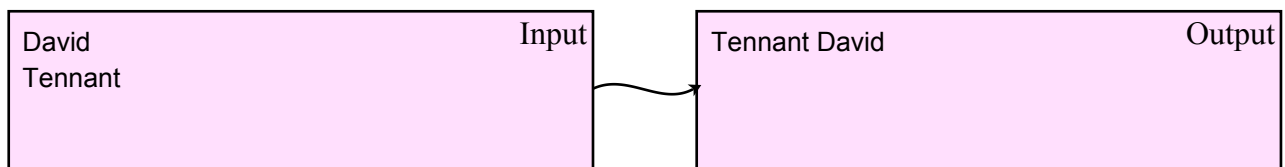
An example of the input and output from the program is shown below.



### 2. Name Swapper

N4

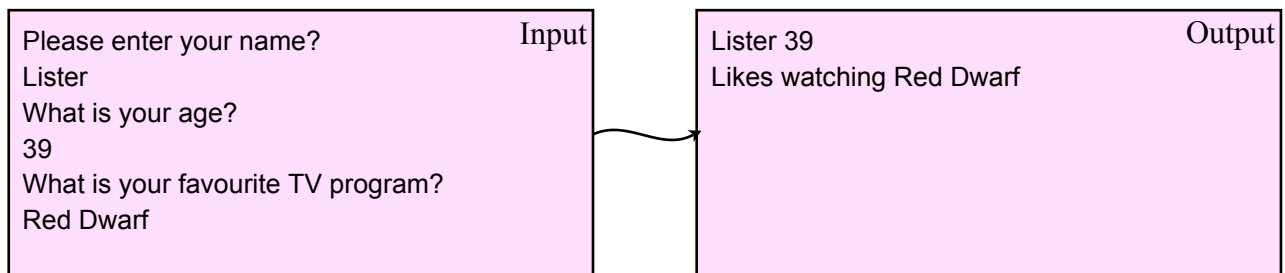
Write a program that will ask the user to type in their first name and surname. The program will then display the two names in reverse order.



### 3. Three In, Three Out (formatted)

N5

Now edit program 1 so that the information entered is displayed differently as shown in the output box below. Note - your output will now have to display variables and text together.

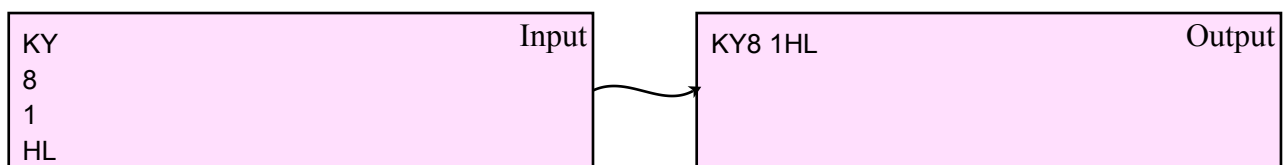


### 4. Postcode Formatter

N5

Ask your user to enter the four separate sections of a post code. Postcodes take the following form: letters, number, number, letters

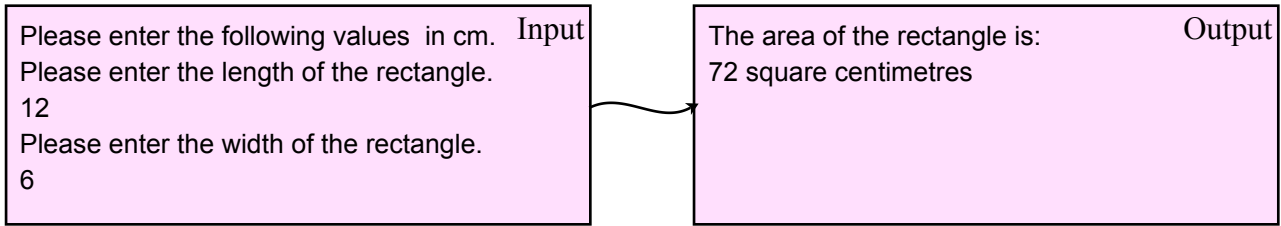
Once entered the postcode should be displayed with a space in the middle.



## 5. Calculate the Area of a Rectangle

**N4**

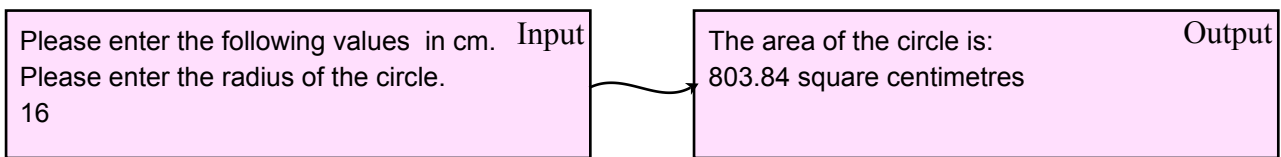
Ask your user to enter the length and width of a rectangle. Your program should calculate the area of the rectangle (length\*width) and display the result with a suitable message.



## 6. Calculate the Area of a Circle

**N5**

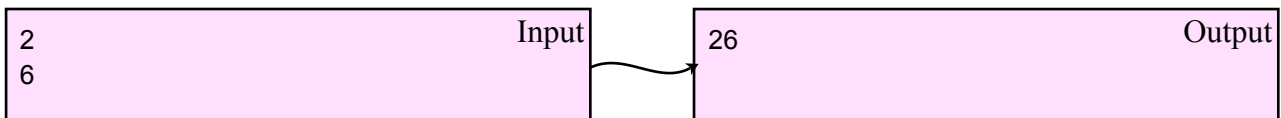
Ask your user to enter the radius of a circle. Your program should use what they have entered to calculate the area of the circle ( $3.14 \times \text{radius} \times \text{radius}$ ) and display the result.



## 7. Number Generator (2 digits)

**N4**

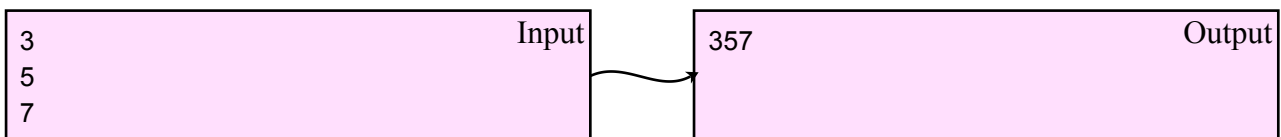
Write a program that inputs two individual integers between 0 and 9. The program should then perform a calculation and store a single number in a third variable called 'total'. As shown below, the total should then be displayed on the screen.



## 8. Number Generator (3 digits)

**N4**

Adapt program 7 to work for 3 numbers instead of 2.



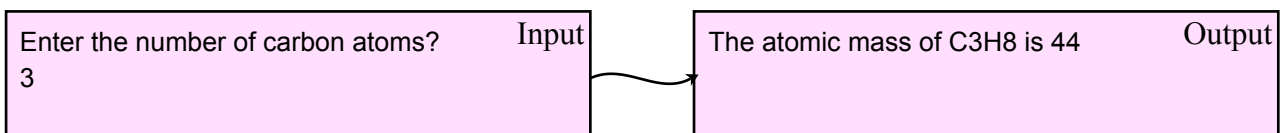
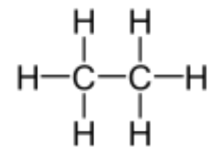
## 9. Calculating the Atomic Weight of Hydrocarbons (Alkanes)

**N5**

A hydrocarbon is a molecule made up of linked Carbon (C) atoms with Hydrogen (H) atoms branching off each Carbon. Your program will ask the user to enter the number of Carbon atoms in a hydrocarbon and use the number entered to then calculate the number of Hydrogen atoms using the formula below. Both numbers should be stored.

$$\text{number of H atoms} = (\text{number of C atoms} \times 2) + 2$$

The atomic weight of the molecule is calculated by multiplying the number of carbon atoms by 12 and adding the number of hydrogen atoms. The number of C and H atoms along with the atomic weight should be displayed as shown in the output below.



## 10. Standard Scratch

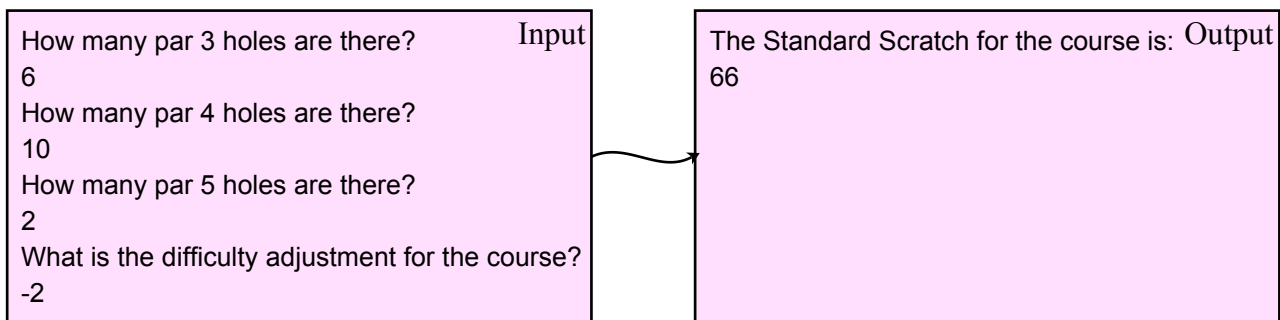
N4

The “standard scratch” of a golf course is calculated by adding together the number of shots it should take to complete each hole. This score is then adjusted depending on the difficulty of the course.

For example:

2 holes take 5 shots (par 5)	$2 \times 5 = 10$	
10 holes take 4 shots (par 4)	$10 \times 4 = 40$	
6 holes take 3 shots (par 3)	$6 \times 3 = 18$	$10 + 40 + 18 = 68$ shots in total
Difficulty adjustment	-2	$68 - 2 = 66$
Standard Scratch =	<b>66</b>	

Write a program that allows the user to enter the information required to calculate the standard scratch score of a golf course. The standard scratch should then be calculated and displayed.



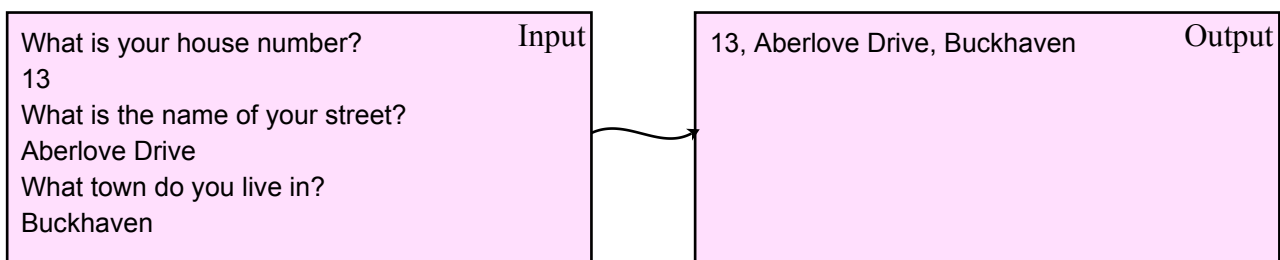
## 11. Formatting an Address

N5

A program is required to store a users address in a single string. The user should be asked three questions:

- What is your house number?
- What is the name of your street?
- What town do you live in?

The program will then combine the users answers in a single string with each answer being separated by a comma and store the result. Finally the program will display the result on the screen.



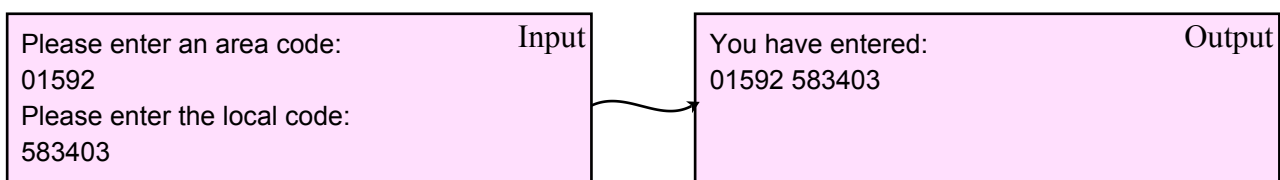
## 12. Formatting Telephone Numbers

N5

UK telephone numbers are often displayed in the following format:  
area code, space, local code

All UK telephone numbers start with a '0'.

Write a program that will allow a user to enter the area code and local code of a telephone number. The program will then display the two codes, separated by a space.



### Selection Problems (IF, ELSE, ELSE IF)

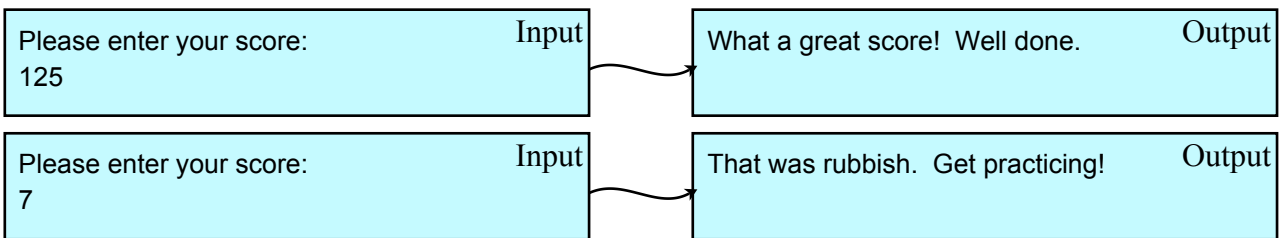
#### 13. Advice Please

**N4** Write a program that asks the user if they would like some advice. If they enter Y, provide them with an amusing message.



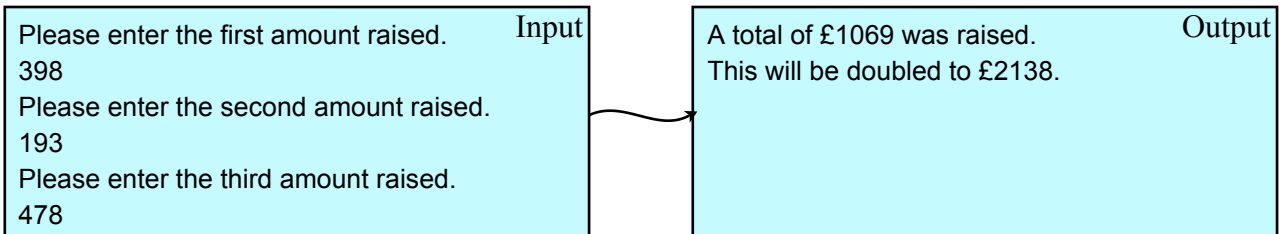
#### 14. Darts

**N4** During a game of darts, the highest score that can be achieved in a single turn is 180. The lowest is 0. Write a program that will allow a dart player to enter their score. The program should congratulate the player if their score was over 100. If the player scores less than 10 they should be told that some practice is required.



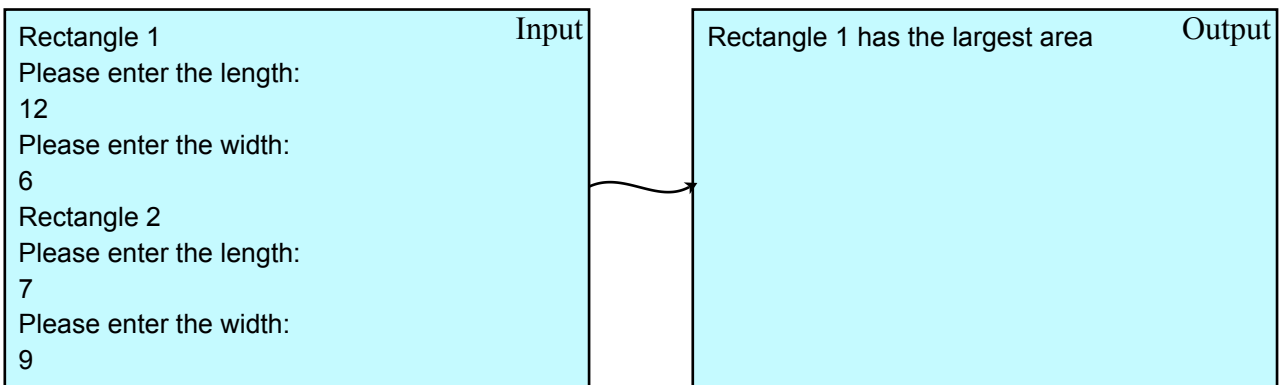
#### 15. Charity Collection

**N4** Three friends have been collecting money for charity. A local company has offered to double the amount of money they collect if they raise over £1000. Write a program that allows the friends to enter their individual amounts. The program should then add the three amounts and store the total. If the total is greater or equal to 1000 the total should be doubled. Finally the total should be displayed.



#### 16. Calculate the Area of a Rectangle (Part 2)

**N4** Program 5 asked you to calculate the area of a rectangle. Expand this program so that it calculates the area of two rectangles. Once both areas have been calculated your program should decide which rectangle has the larger area and display a suitable message.



# Programming Challenges

## 17. Advice Please (Part 2)

N4

Expand program 13 to include an alternative message if the user types N. An error message should be given to the user if they enter anything other than Y or N.



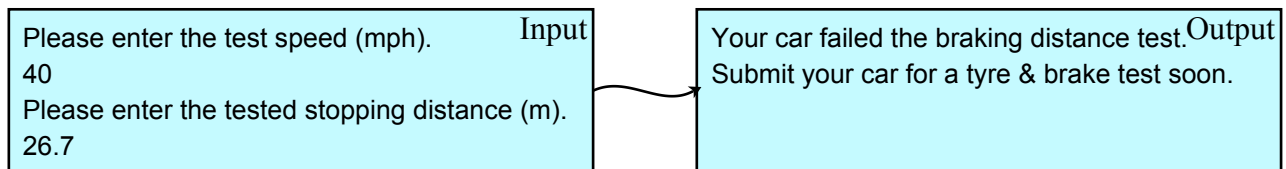
## 18. Tyre & Brake Wear

N4

As a car's tyres and brakes wear out it takes longer for a car to stop. A list of recommended stopping distances are shown below.

20 miles per hour	6 metres to stop
30mph	14m
40mph	24m
50mph	38m
60mph	55m
70mph	75m

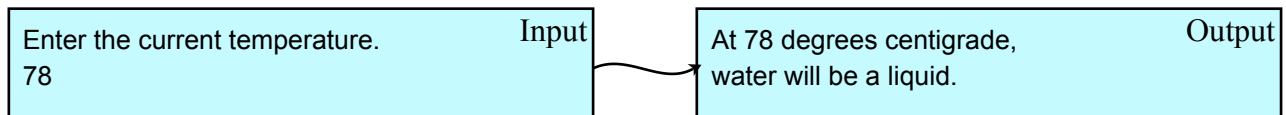
Write a program to analyse a braking distance test. The user should be asked to enter the speed (mph) they were doing and the distance (m) it took them to stop. If the distance is longer than the recommended stopping distance the user should be advised to go for a tyre and brake check.



## 19. Solid, Liquid, Gas

N5

At normal atmospheric pressure, water changes state to a solid at 0°C or below and a gas at 100°C or above. It remains a liquid at any other temperature. Write a program that will return "solid", "liquid" or "gas" to the user depending on the temperature they enter.



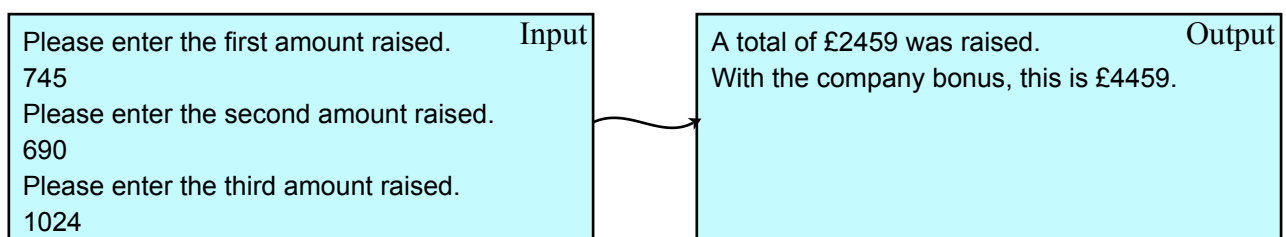
## 20. Charity Collection (Part 2)

N5

The local company have decided that due to current financial pressure than can not afford to double any amount of money raised over £1000. The following new decisions are made:

- any amount raised less than £1000 has a £100 bonus (for example £345 raised = £445 total)
- the company will still double the amount raised between £1000 and £2000 (for example £1282 raised = £2564 total)
- if the amount is over £2000 the initial £2000 is doubled but any amount after that is not (for example £2054 raised =  $2 * £2000 + £54 = £4054$  total)

Rewrite program 15 to take account of the above decisions.

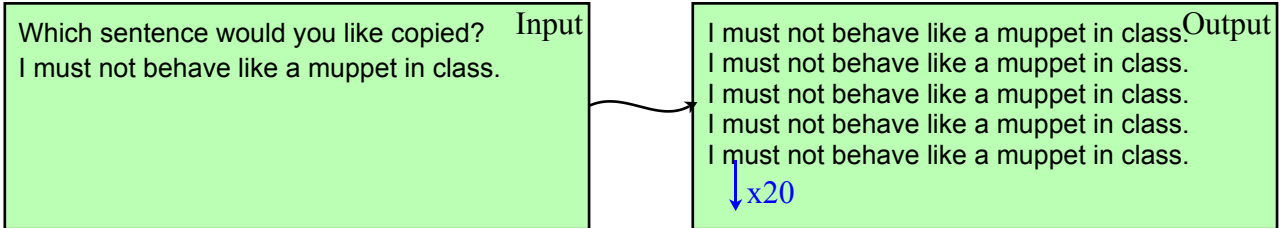




## Iteration (Unconditional Loops)

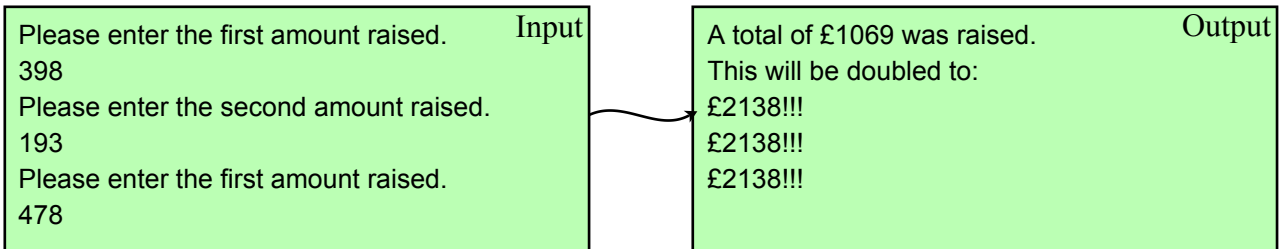
### 21. Lines Cheat

**N4** A naughty pupil has been given lines to copy as a punishment from their Computing teacher. They have been asked to type out “I must not behave like muppet in class” 20 times. Write a program that asks a pupil to enter a sentence. The same sentence should then be displayed 20 times.



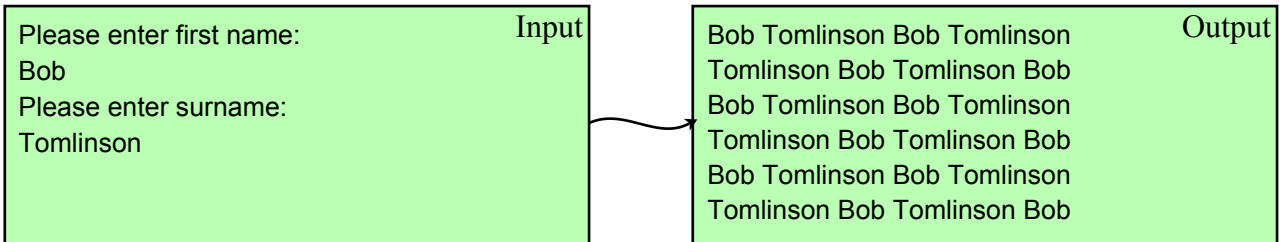
### 22. Charity Collection (Part 3)

**N4** Adapt Program 20 so that the total is displayed 3 times to emphasise the amount of money raised.



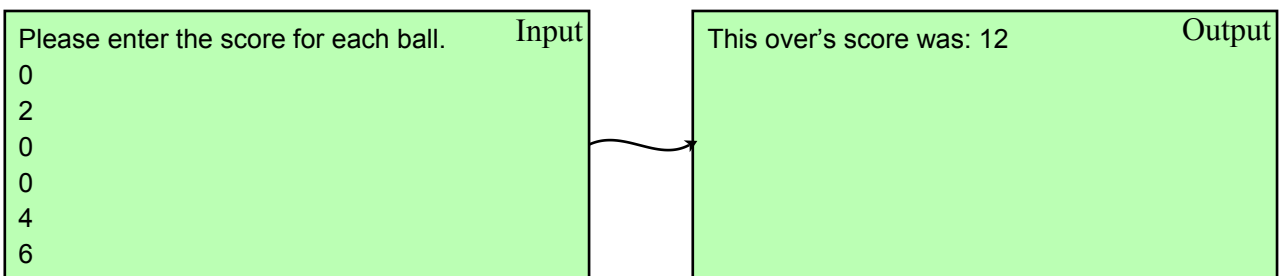
### 23. Name Switch

**N4** An artist has been experimenting with making art from people's names. Their first attempt involves repeatedly swapping someone's first name and surname to create a pattern. Write a program that asks for two names to be entered and then uses the input to display the correct pattern.



### 24. Cricket Over

**N4** In cricket a bowler bowls 6 balls at a time. This is called an 'over'. Write a program that allows 6 scores to be entered (for example, 0,2,0,0,4,6). One for each ball in the over. The total scored in that over should then be displayed.



# Programming Challenges

## 25. Average Temperature

N4

A weather experiment is set up to calculate the average temperature on a mountain peak during the course of a week. The following measurements are taken at 1pm every day.

Mon 12°C

Tue 14°C

Wed 7°C

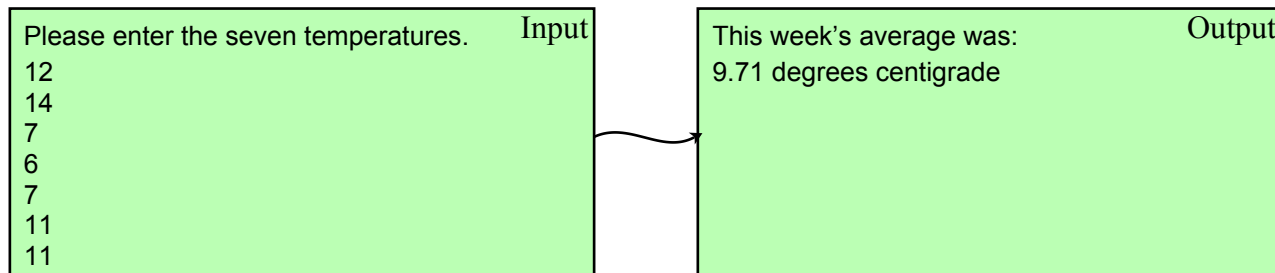
Thur 6°C

Fri 7°C

Sat 11°C

Sun 11°C

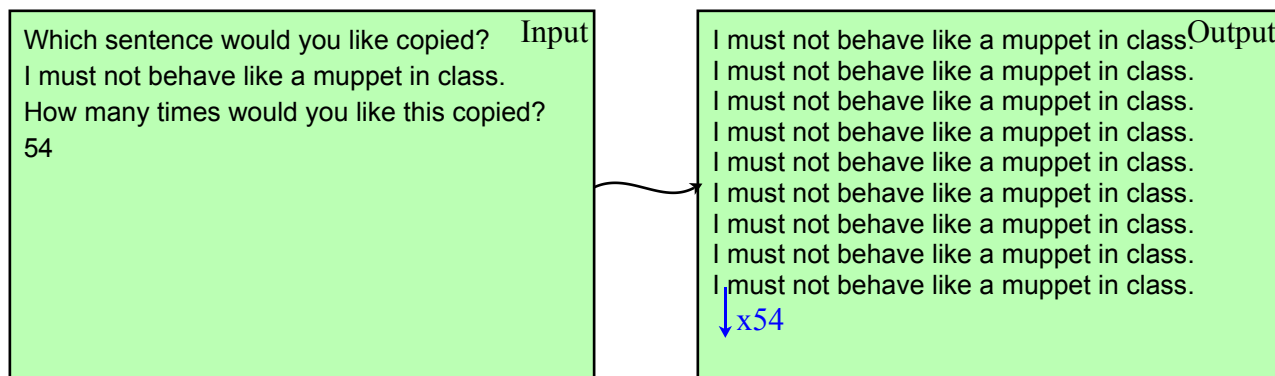
A program is required to allow the experimenter to enter the 7 temperatures. The average for the week should then be displayed, to two decimal places, as shown below.



## 26. Lines Cheat (Part 2)

N4

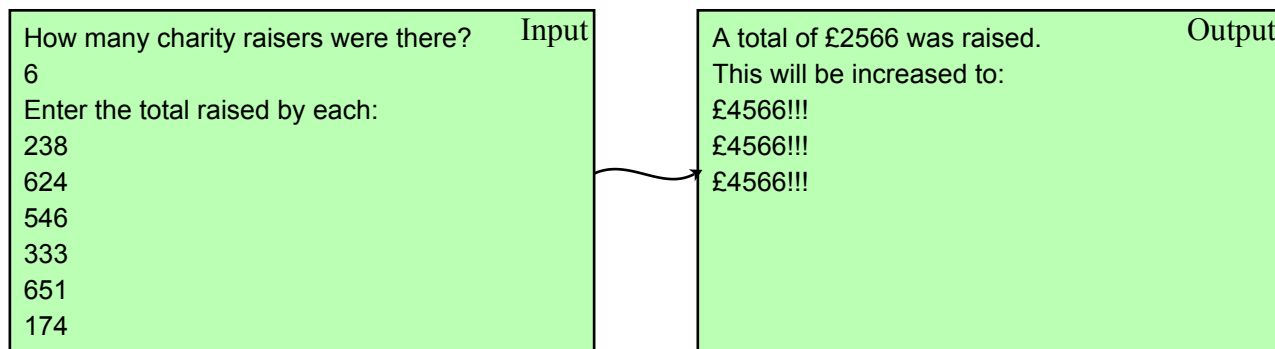
The naughty pupil forgot to hand their lines in and now has more to do. Adapt program 21 to allow the pupil to select how many lines the program produces.



## 27. Charity Collection (Part 4)

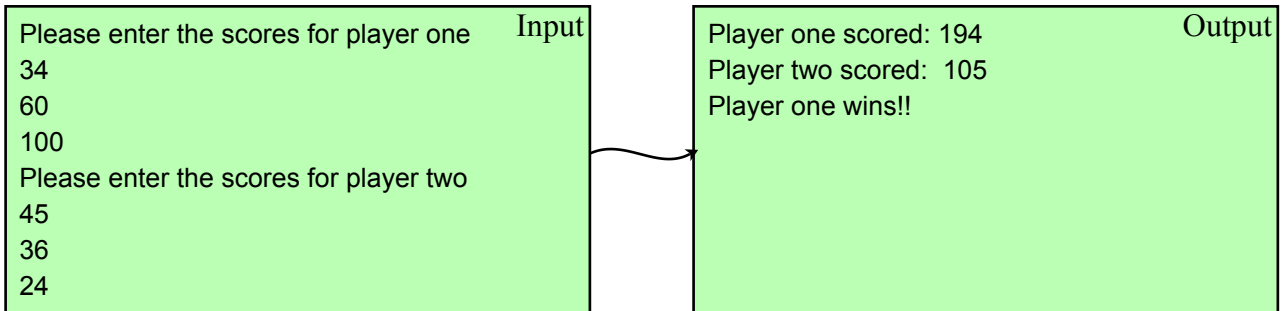
N4

The following year the three friends recruit many more charity raisers. Adapt program 22 to ask for the number of people raising money to be entered. The program will then total up the money raised by the group and display the output as before.



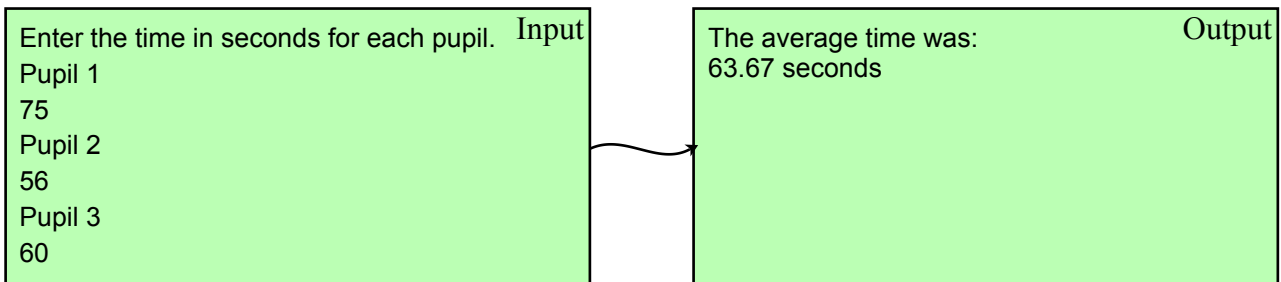
**28. Pailwater Darts Tournament**

**N4** Jack and Jill are competing in the annual Pailwater darts competition. Each competitor throws 9 darts (3 lots of 3 darts) each. The player with the highest total score is the winner. Write a program to calculate both scores and announce the winner.



**29. Press Up Challenge**

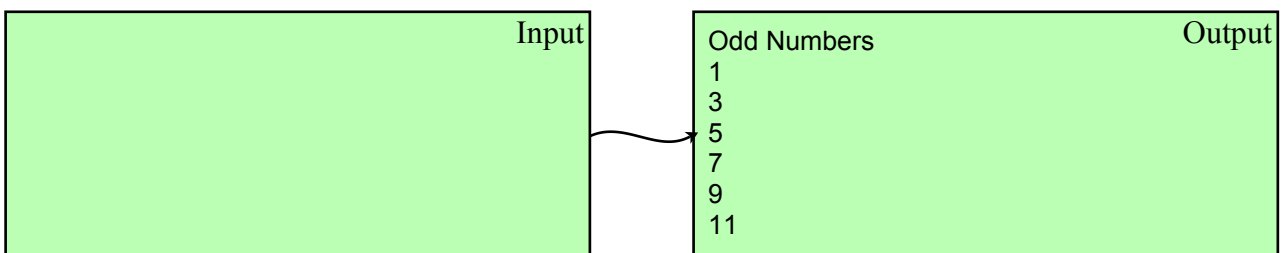
**N5** Naebor High School have organised a fitness challenge with nearby Wisnaeme Academy. Each school picks their three fittest pupils and record the average time it takes them to do 50 press ups. Write a program that asks a user to enter a time for pupil as shown below (note that the pupils are numbered). The program should then display the average time (to 2 decimal places).



**30. Number Patterns**

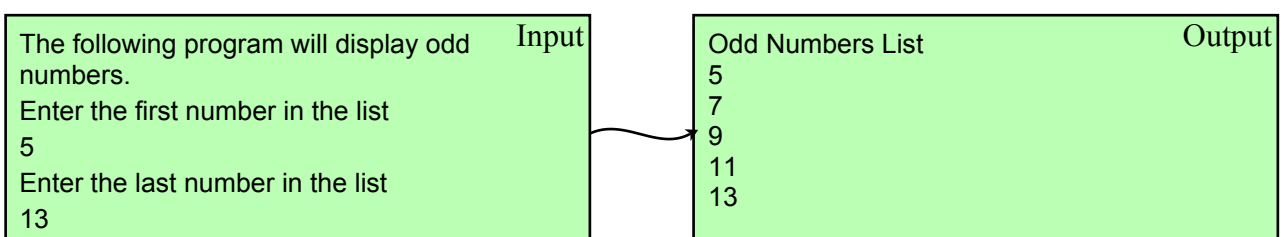
**N5** A math teacher wishes to show her class a variety of number patterns. She starts by showing her class a list of odd numbers, explaining that the pattern starts at 1 and then misses out every second number:  
 1 3 5 7 9 11

Write a program to display the list of odd numbers shown above.



**31. Number Patterns (Part 2)**

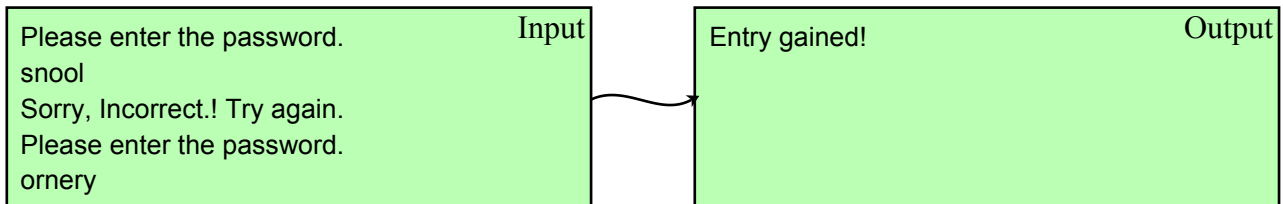
**N5** Adapt program 26 so that the user can choose the first and last odd numbers displayed.



## Iteration (Conditional Loops)

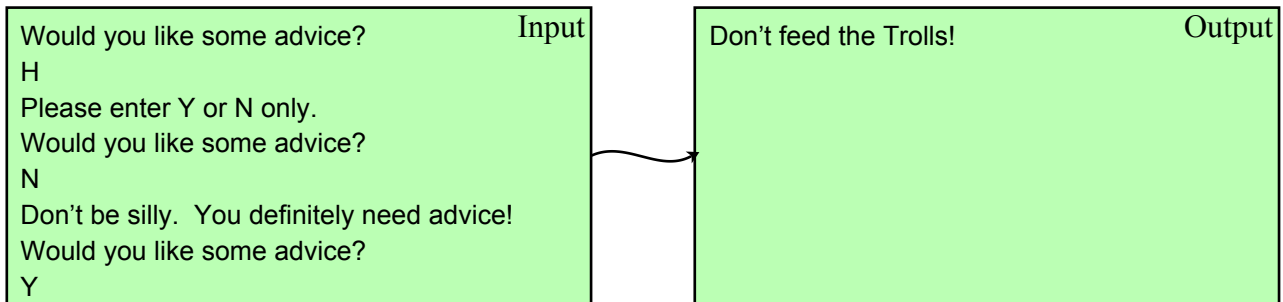
### 32. Password

**N4** A program is required to continually ask a user to enter a password until it has been entered correctly. The program should give the user an error message if they enter the wrong password. A message “Entry gained!” should be displayed when the password is entered correctly. You may choose the password.



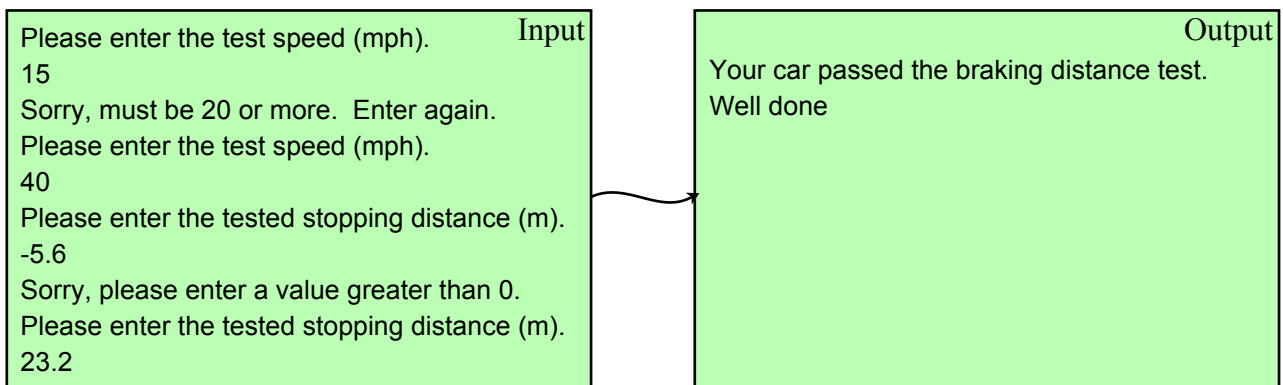
### 33. Advice Please (Part 3)

**N4** You have decided that your user definitely requires some advice. Edit program 17 so that will continue to ask if your user wants advice until they enter Y. Include an amusing message if they enter N and an error message if they don't enter Y or N.



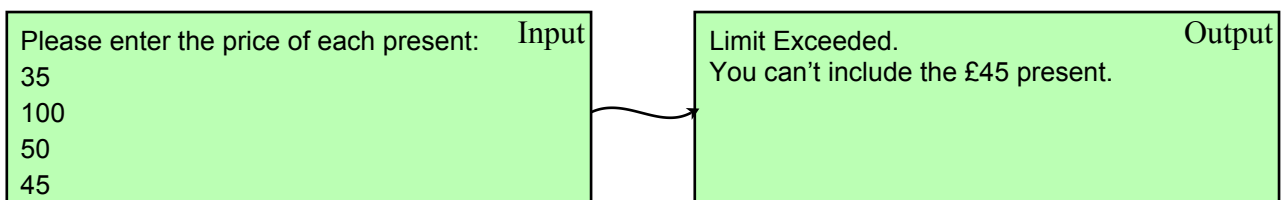
### 34. Tyre & Brake Wear (Part 2)

**N4** Edit program 18 to ensure that the speed that is entered for the brake distance test is always greater than or equal to 20mph and the stopping distance is always greater than 0.



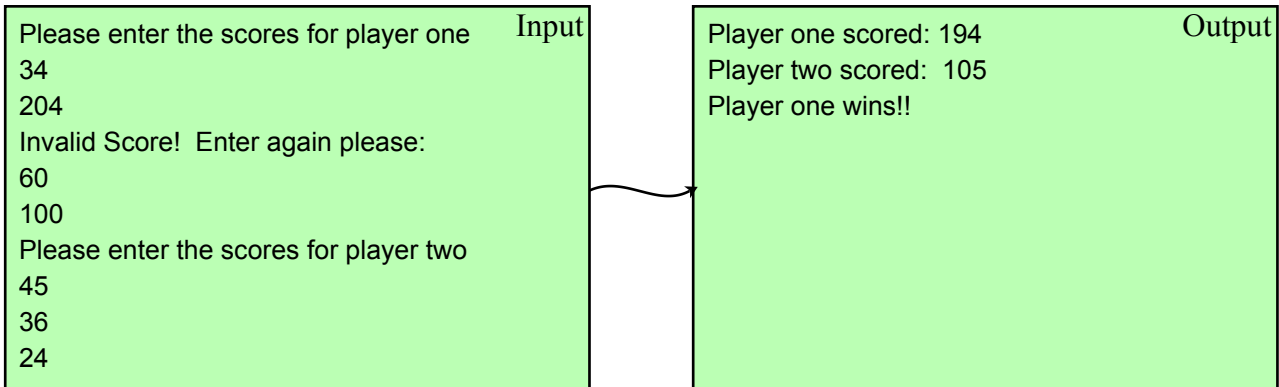
### 35. No More Presents

**N4** You have £200 to spend on your birthday. Write a program that will ask you to enter the price of each present you want until your total is over £200. The program should produce the output shown.



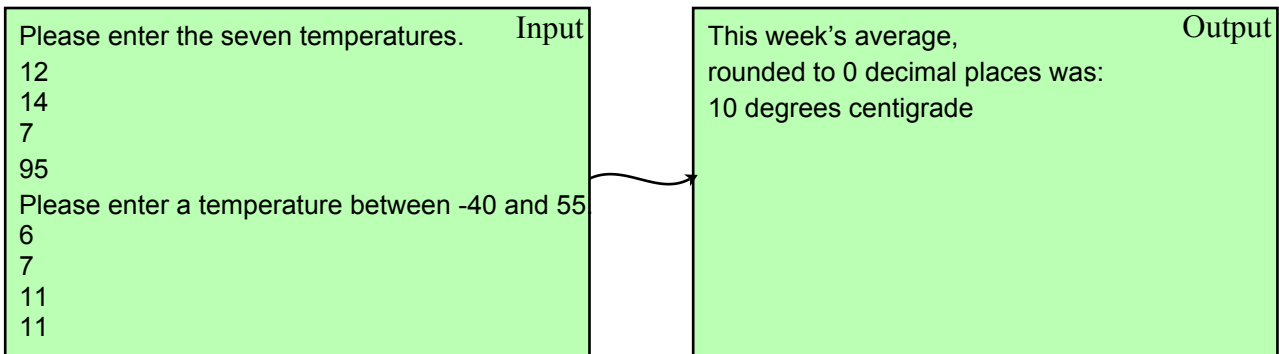
**36. Pailwater Darts Tournament (Part 2)**

**N5** The minimum and maximum scores that can be achieved with three darts are 0 and 180. Edit program 28 to validate your input (make sure the scores entered are between 0 and 180). Ensure you give your user an appropriate error message if they type in a wrong value.



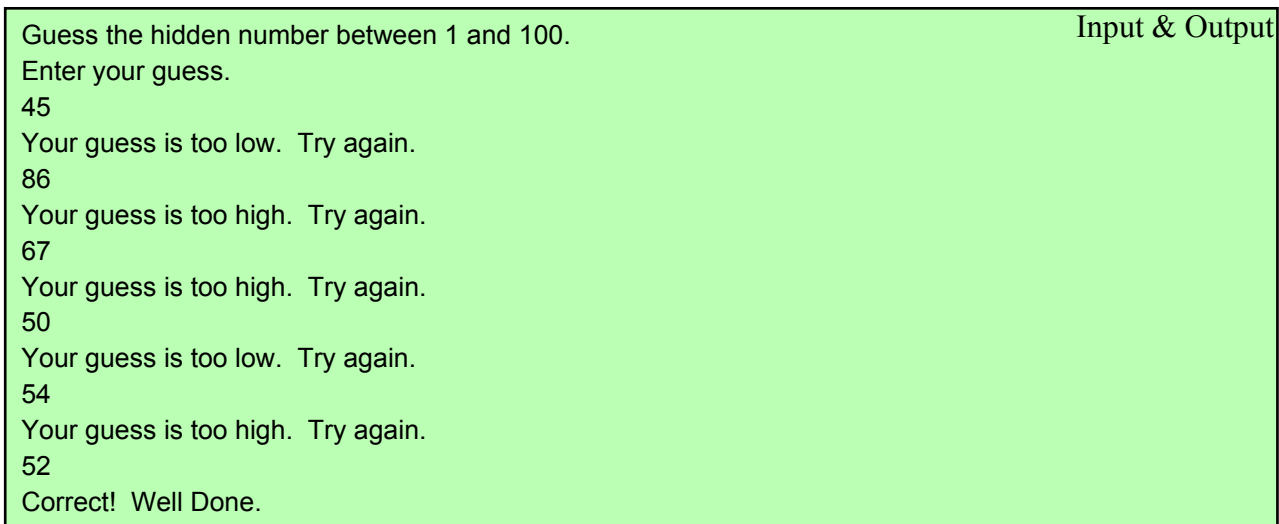
**37. Average Temperature (Part 2)**

**N5** Adapt program 25 to validate each temperature. Inputs should be between -40°C and 55°C. As before, the program should output the average temperature for the week but this time the output should be displayed as an integer.



**38. Guess the Number**

**N5** A game is created where a user is required to guess an unknown number between 1 and 100. Each time the user guesses the program informs them if their guess is too high, too low or correct. The guessing game only finishes when the user's guess matches the unknown number. An example of the program running (input and output) is shown below.



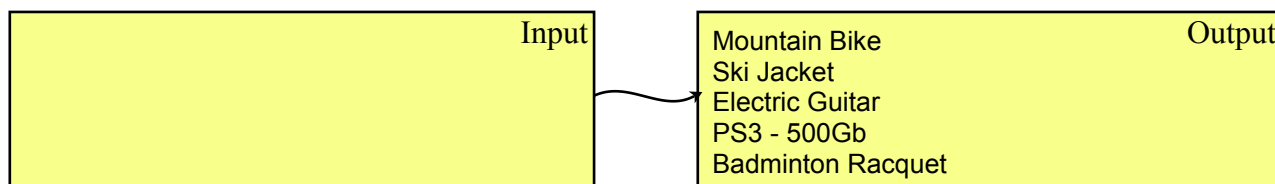
Extra Challenge - adapt the program to show how many guesses the user had before they were correct.

## Arrays (Storing Multiple Examples of the Same Data)

### 39. For Sale

N5

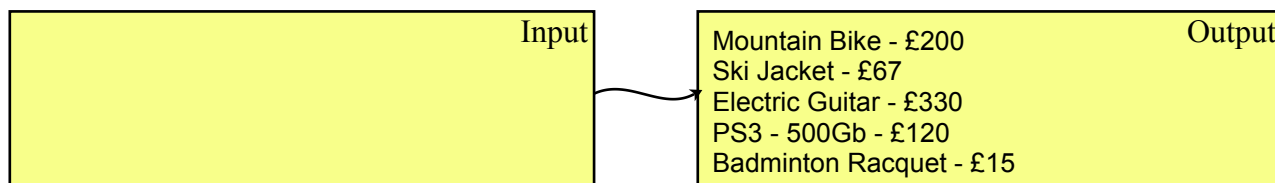
A program is required to display a list of 5 items for sale on a monitor. Store a list of items in an array and then display these items as output.



### 40. For Sale (Part 2)

N5

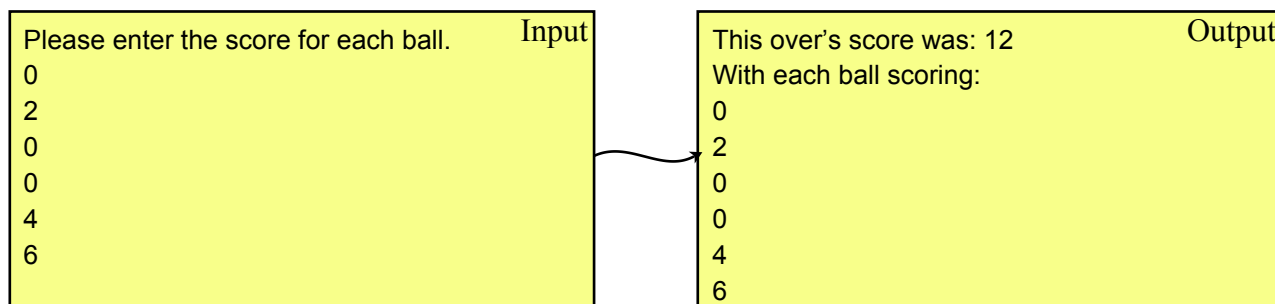
Edit program 39 above to store a second list of 5 prices, one for each of the sale items. Display both the name and price of each sale item on the same line.



### 41. Cricket Over (Part 2)

N5

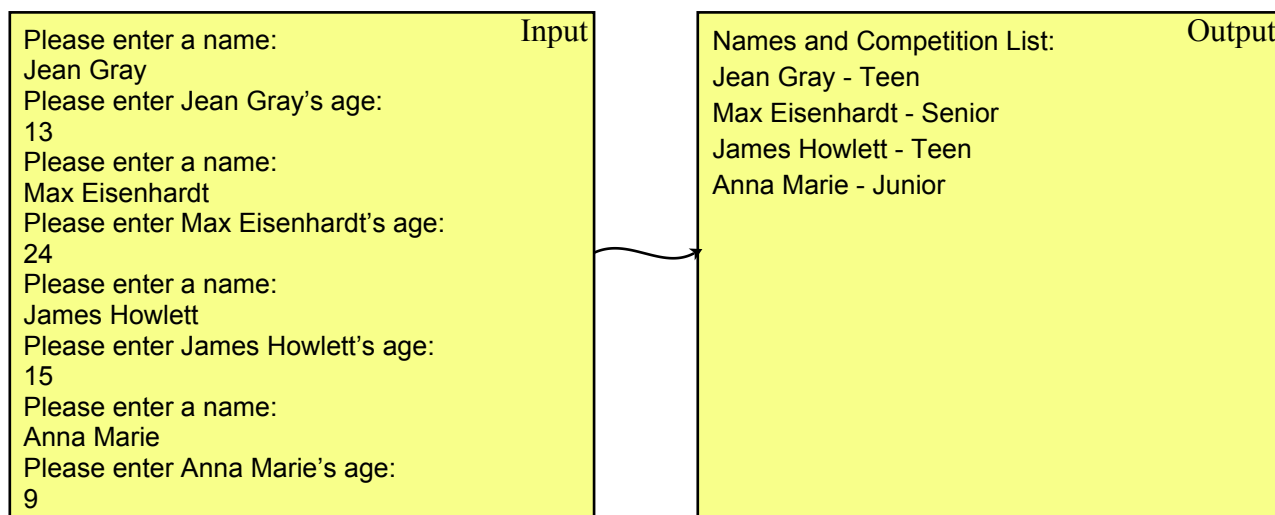
Edit program 24 to store the score for each of the six balls in the over in an array. Use the array to display the 6 scores, as well as the total, in the output.



### 42. Dance Group

N5

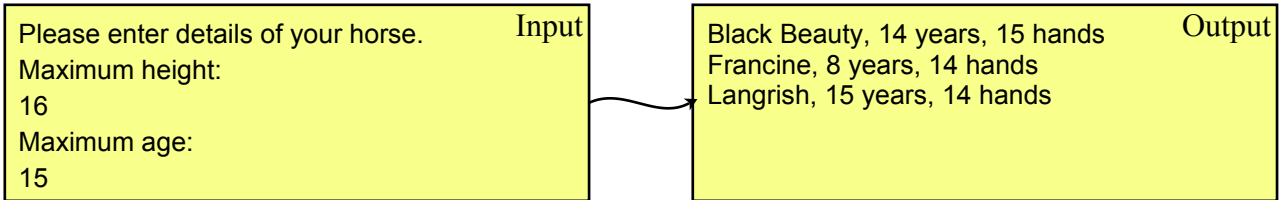
You have been asked to write a program to store the names and ages of 4 competitors in a dance competition. The program should display the name of the competitor and which level of competition they should be entered in. 'Junior' competitors are less than 12 years old, 'senior' competitors are at least 18 years old. 'Teen' competitors are aged 12-17.



### 43. Horse Hands



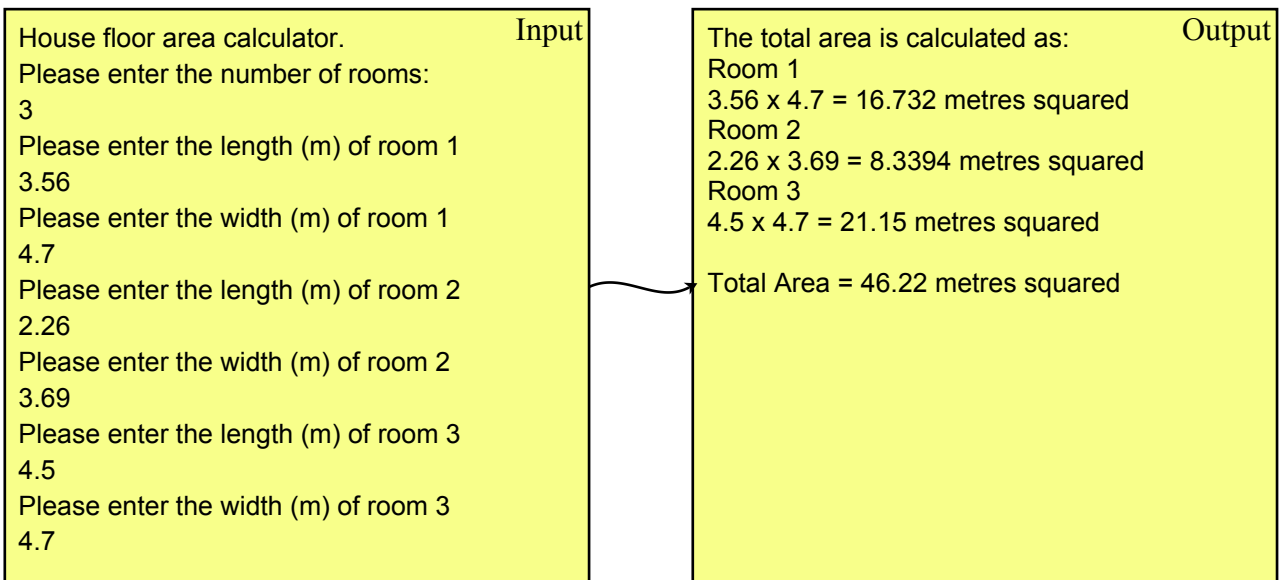
A computer program stores the names, ages and height (the height of horses is measured in 'hands' - for example, 16) of fifteen horses in a riding school. The user of the program will be asked to select a horse by entering a maximum age and height of the horse they wish to ride. The data for the fifteen horses will be used to provide the user with a list of suitable. A horse is suitable if its age and height are both less than or equal to the values entered by the user.



### 44. House Size



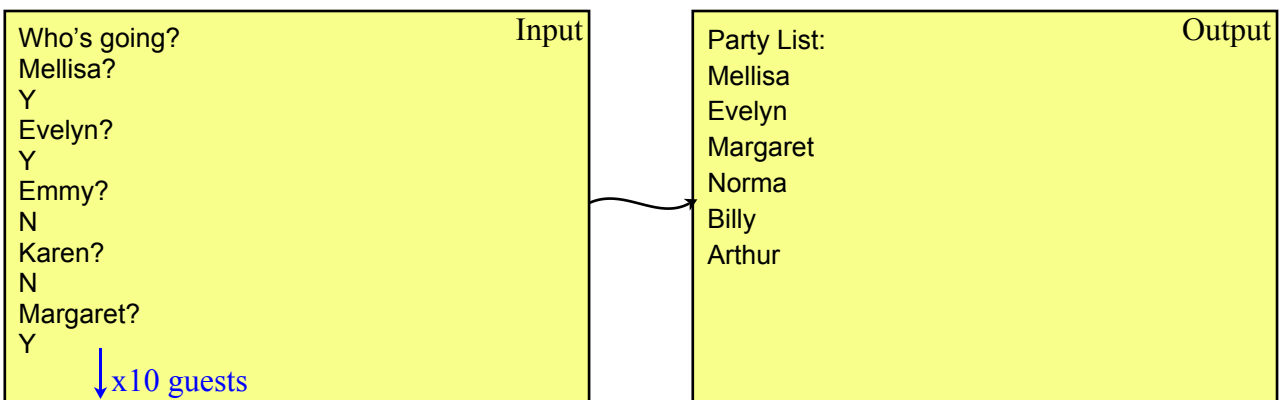
The size of a house is calculated by adding together the floor area of each room. Write a program that asks the user how many rooms are in a house, stores the length and width of each room (as entered by the user) and finally calculates the total floor area of a house. The output should be formatted as shown below.



### 45. Who's Going



Katy is having a birthday party. She sends invites to 10 of her friends (Mellisa, Evelyn, Emmy, Karen, Margaret, Norma, Agnes, Billy, Robert & Arthur). Katy requires a computer program to store her friends names and whether or not they will be attending the party. Write a program that asks the user if each guest is attending. The program should store true or false for each person. The output from the program should be a list of names of everyone who is going to the party.



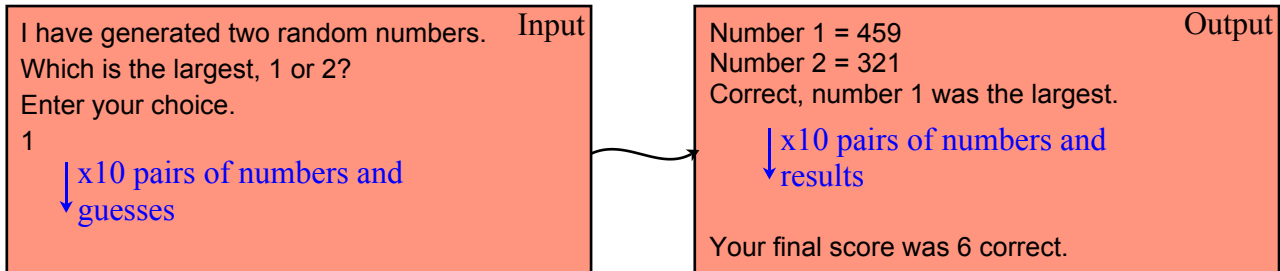
## Predetermined Functions

### 46. Guess the Number (Part 2) Random

**N5** Using the random function, adapt program 38 to make the program generate the number (from 1 to 100) that the user is asked to guess. The program should run as before, so the output should not change.

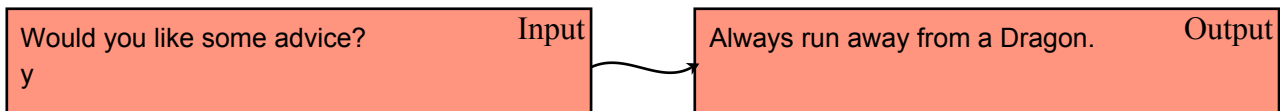
### 47. Which is Bigger? Random

**N5** A program generates two random integers between 1 and 1000. A user is asked to guess which of the two numbers is the largest. If they are correct, they score a point. The program repeats this process 10 times and then displays the players final score.



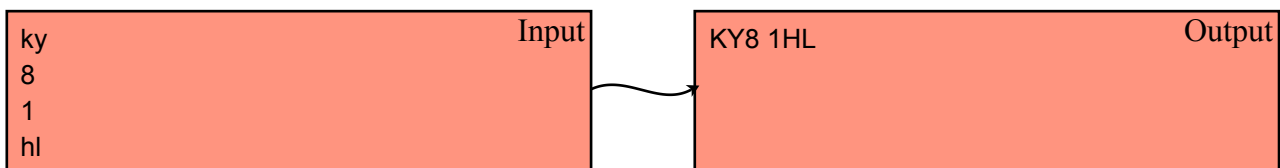
### 48. Advice Please (Part 4) Upper Case

**N5** Good programs should allow for a variety of different inputs. For example, if the user of Advice Please (program 33) enters 'y' rather than 'Y' the program will not give them advice. Using the upper case function, convert the users input into upper case letters to ensure they are given advice, even if they enter 'y'.



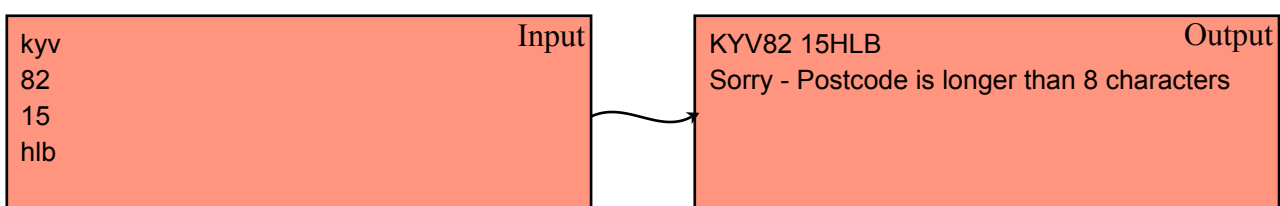
### 49. Postcode Formatter (Part 2) Upper Case

**N5** Edit program 4 to ensure that the postcode is always displayed in capital letters. Even if the user inputs the letters in lower case.



### 50. Postcode Formatter (Part 3) Length

**N5** UK Postcodes are a maximum of 8 characters long (7 letters and numbers + 1 space). Make changes to program 49 to ensure that the user is given an error message if they enter a postcode longer than 8 characters.





## 51. Left Over Paint

**Modulus & Round**

**N5**

A painter and decorator requires a small program that will calculate the number of pots of paint that will be required to complete a job. The program should also include how many metres squared could be painted with the left over paint.

Input

```
Enter the area in m2 to be painted.
178
Enter the area (m2) that a single pot covers.
25
```

Output

```
You will need 10 pots of paint.
You can paint 22 m2 with the left over paint.
```

## 52. The ASCII Game

**Random & Ord/Chr**

**N5**

A teacher wishes to demonstrate the relationship between ASCII codes and letters of the alphabet. Write a program that will generate a random number between 97 and 122 (the ASCII values for lower case a to z). The program should display the generated number and ask the user to guess what letter of the alphabet the number represents. The user should be prompted, as shown below, if they get the answer wrong and be asked to enter another guess.

```
Guess which letter of the alphabet is
represented in ASCII by the number - 113
f
Sorry f would be 102
r
Sorry r would be 114
q
Correct, q = 113
```

Input & Output

## 53. Guess the Number (Part 3)

**Round**

**N5**

The guess the number game currently allows the user to enter a real number as their guess. Adapt program 46 to include input validation, ensuring that the user can only enter an integer as their guess.

```
Guess the hidden number between 1 and 100.
Enter your guess.
45
Your guess is too low. Try again.
86.7
Not a whole number. Please enter again
86
Your guess is too high. Try again.
67
Correct! Well Done.
```

Input & Output

## 54. Scrabble Letters

**Random & Chr**

**N5**

Create a program that will generate 7 random scrabble letters, displaying the score of each letter.

Output

```
Your letters are:
F - 4
Z - 10
E - 1
S - 1
U - 2
C - 4
H - 4
```