Computer Programming
Bachelor in Biomedical Engineering
Bachelor in Applied Mathematics and Computing
Course 2020 / 2021

Exercise Sheet 5 - SOLUTIONS

Iteration Statements: While

Content Table

Exercise 1

Write a program that asks the user to introduce numbers and stores them in a vector one after the other. After introducing a number the program asks the user if he/she wants to introduce more (Y/N). When the user decides not to introduce more numbers, the program prints the content of the vector on screen.

Example of execution:

```
Introduce a number: 3
Do you want to introduce more values (Y/N)? Y
Introduce a number: 7
Do you want to introduce more values (Y/N)? Y
Introduce a number: 2
Do you want to introduce more values (Y/N)? N
The numbers in the vector are:
 3 7 2
```

What will happen if the user answers the question with something different from Y or N? The program will stop asking for numbers.

SOLUTION

```
clear;
vector = [];
index   = 0;
cContinue = 'Y';

while (cContinue == 'Y')
    number = input('Introduce a number: ');
    index = index + 1;
    vector(index) = number;
    cContinue = input('Do you want to introduce more values (Y/N)?
','s');
end

disp ('The numbers in the vector are:');
for value = vector
    fprintf('%d ', value);
end
```

Exercise 2

Modify the previous program so that if the user answers with something different from Y or N the program repeats the question.
Tip: You need to use a second while loop inside the first while loop that repeats the question again and again until the answer is Y or N.

Example of execution:

```
Introduce a number: 3
Do you want to introduce more values (Y/N)? X
Sorry, I don't understand.
Do you want to introduce more values (Y/N)? T
Sorry, I don't understand.
Do you want to introduce more values (Y/N)? Y
Introduce a number: 7
Do you want to introduce more values (Y/N)? N
The numbers in the vector are:
 3 7
```

SOLUTION

```
clear;
vector = [];
index   = 0;
cContinue = 'Y';
while (cContinue == 'Y')
    number = input('Introduce a number: ');
    index = index + 1;
    vector(index) = number;
    cContinue = input('Do you want to introduce more values (Y/N)?
','s');
    while ((cContinue ~= 'N') && (cContinue ~= 'Y'))
        disp('Sorry, I don''t understand');
        cContinue = input('Do you want to introduce more values (Y/N)?
','s');
    end
end

disp ('The numbers in the vector are:');
for value = vector
    fprintf('%d ', value);
end
```

Exercise 3

Write a program that asks the user to introduce 3 consonants. If the character is a vowel the program displays an error message and asks for a consonant again. At the end, it displays the 3 consonants introduced one after the other.
Note: you can assume that the user will only introduce consonants or vowels (no punctuation characters), and always in lowercase.

Example of execution:

```
Introduce a consonant: d
Introduce a consonant: o
Wrong character
Introduce a consonant: g
Introduce a consonant: c
The consonants introduced are:
d g c
```

SOLUTION

```
clear;
vect = [];
cont = 0;
while (cont < 3)
    letter = input('Introduce a consonant: ','s');
    switch letter
        case {'a','e','i','o','u'}
            disp('Wrong character');
        otherwise
            cont = cont + 1;
            vect(cont) = letter;
    end
end
disp('The consonants introduced are: ');
for val = vect
    fprintf('%c ', val);
end
```

Exercise 4

Write a program that asks the user to introduce numbers. If the number is dividable by 2, 3 or 5, it prints the result of the corresponding divisions. Otherwise, if the number cannot be divided by any of those 3 numbers, the program finishes.

Example of execution:

```
Introduce a number: 30
        30 divided by 2 is 15
        30 divided by 3 is 10
        30 divided by 5 is 6
Introduce a number: 15
        15 divided by 3 is 5
        15 divided by 5 is 3
Introduce a number: 7
      Not divisible by 2, 3 or 5
```

SOLUTION

```
clear;
number = input('Introduce a number: ');
while ((rem(number,2) == 0) || (rem(number,3) == 0) || (rem(number,5) ==
0))
    if (rem(number,2) == 0)
        fprintf('%d divided by 2 is %d\n', number, number/2);
    end
    if (rem(number,3) == 0)
        fprintf('%d divided by 3 is %d\n', number, number/3);
    end
    if (rem(number,5) == 0)
        fprintf('%d divided by 5 is %d\n', number, number/5);
    end
    number = input('Introduce a number: ');
end
disp('Not divisible by 2, 3 or 5');
```

ANOTHER SOLUTION

We can use a variable containing a Boolean value for the condition of the while loop. Initially we set the value to true (1) and we want to stop the loop when we set it to false (0).

```
clear;
bContinue = 1;
while (bContinue == 1)
    number = input('Introduce a number: ');
    if ((rem(number,2) ~= 0) && (rem(number,3) ~= 0) && (rem(number,5)
~= 0))
        disp('Not divisible by 2, 3 or 5');
        bContinue = 0;
    else
        if (rem(number,2) == 0)
            fprintf('%d divided by 2 is %d\n', number, number/2);
        end
        if (rem(number,3) == 0)
            fprintf('%d divided by 3 is %d\n', number, number/3);
        end
        if (rem(number,5) == 0)
            fprintf('%d divided by 5 is %d\n', number, number/5);
        end
    end
end
```

Exercise 5

Modify Exercise 1, so that after the user finishes introducing numbers the program asks him/her to introduce a value. The program then verifies and prints whether this value is in the vector or not.

SOLUTION

This is the 'search algorithm'. We compare the values in the vector one by one with the value the user introduced. As soon as we find this value in the vector, we stop looping through the vector since it is not necessary to continue on comparing.

```
clear;
vector = [];
index   = 0;
cContinue = 'Y';

while (cContinue == 'Y')
    number = input('Introduce a number: ');
    index = index + 1;
    vector(index) = number;
    cContinue = input('Do you want to introduce more values (Y/N)?
','s');
end

index   = 1;
bFound = 0;
number = input ('Introduce a number: ');
while ((index <= length(vector)) && (bFound == 0))
    if (number == vector(index))
        bFound = 1;
    else
        index = index + 1;
```

```
    end
end

if (bFound == 1)
    disp ('The number is in the vector');
else
    disp ('The number is NOT in the vector');
end
```

Exercise 6

Modify the previous program so that it now asks the user to introduce two values. The program then tells the user if at least one of them is in the vector, or if none of the two have been found.

SOLUTION

```
clear;
vector = [];
index   = 0;
cContinue = 'Y';

while (cContinue == 'Y')
    number = input('Introduce a number: ');
    index = index + 1;
    vector(index) = number;
    cContinue = input('Do you want to introduce more values (Y/N)?
','s');
end

index   = 1;
bFound = 0;
number1 = input ('Introduce a number: ');
number2 = input ('Introduce another number: ');
while ((index <= length(vector)) && (bFound == 0))
    if ((number1 == vector(index)) || (number2 == vector(index)))
        bFound = 1;
    else
        index = index + 1;
    end
end

if (bFound == 1)
    disp ('At least one of the two numbers appears in the vector');
else
    disp ('The vector does not contain any of the numbers');
end
```

Exercise 7

Write a program that asks the user to introduce numbers until he/she introduces a 0. It then prints the biggest number.

Example of execution:

```
Introduce a number: 10
The number is 10
Introduce a number: 24
The number is 24
Introduce a number: 9
The number is 9
Introduce a number: 20
The number is 20
Introduce a number: 0
The biggest number is 24
```

SOLUTION

```
clear;
maxNum  = 0;
numberRead = input('Introduce number: ');
while (numberRead ~= 0)
    fprintf('The number is %d\n', numberRead);
    if (numberRead > maxNum)
        maxNum = numberRead;
    end
    numberRead = input('Introduce number: ');
end
fprintf('The biggest number is %d\n', maxNum);
```

Exercise 8

Write a program that displays a list of products available in a supermarket. Then, the program asks the user how much money they want to spend. The program then asks the user to introduce the code of the product they wish to buy.

- If the product code is not valid, the program will display an error message
- If the user does not have enough money to buy a product, the program will also display an error message
- If the user can buy the product, the program will show the user how much money they have left after buying the product

This continues until the user runs out or money or chooses to exit. When the user finishes buying products, the program will display the total number of products they bought and how much money the user has left.

Example of execution:

```
Available products:
(a)pple: 2
(b)read: 3
(c)heese: 8
(d)ill: 1
(e)xit
```

```
How much money would you like to spend? 10
Introduce the code of the product you would like to buy: a
You have 8 money left
Introduce the code of the product you would like to buy: b
You have 5 money left
Introduce the code of the product you would like to buy: c
You do not have enough money to buy that product.
Please enter another code.
Introduce the code of the product you would like to buy: f
Sorry, we do not have that product.
Please enter another code
Introduce the code of the product you would like to buy: d
You have 4 money left
Introduce the code of the product you would like to buy: e
You have bought 3 products
You have 4 money left
```

SOLUTION

```
clear;
disp('Available products:');
disp('(a)pple: 2');
disp('(b)read: 3');
disp('(c)heese: 8');
disp('(d)ill: 1');
disp('(e)exit');

budget = input('How much money would you like to spend? ');
bought = 0;
product = 'a';
while((budget > 0) && (product ~= 'e'))
    product = input('Introduce the code of the product you would like to
buy: ', 's');
    switch product
        case 'a'
            price = 2;
        case 'b'
            price = 3;
        case 'c'
            price = 8;
        case 'd'
            price = 1;
        case 'e'
            price = 0;
        otherwise
            price = -1;
    end
    if(price > 0)
        if(budget >= price)
            budget = budget - price;
            bought = bought + 1;
            if(budget > 0)
                fprintf('You have %d money left\n', budget);
            end
        else
            disp('You do not have enough money to buy that product.
Please enter another code.');
        end
    elseif(price < 0)
```

```
        disp ('Sorry, we do not have that product. Please enter another
code');
    end
end

fprintf('You have bought %d products\n', bought);
if(budget > 0)
    fprintf('You have %d money left\n', budget);
else
    disp ('You have spent all your money');
end
```

Exercise 9

Stanislaw Ulam stated that starting from any integer number, if you follow these steps you'll reach the number 1:

- If the number is even divide it by 2
- If the number is odd multiply it by 3 and add 1

Write a program to prove Ulam's conjecture. The program asks the user to introduce a number and prints on screen all the numbers obtained until it reaches 1.

SOLUTION

```
clear;
numberVal = input('Introduce a number: ');
while (numberVal ~= 1)
    if (rem(numberVal,2) == 0)
        numberVal = numberVal / 2;
    else
        numberVal = (numberVal * 3) + 1;
    end
    fprintf('%d\n', numberVal);
end
```

Exercise 10

Write a program that asks the user to introduce a number and indicates whether the number is prime or not.

Example 1:

Introduce a number: 7
The number is prime

Example 2:

Introduce a number: 24
The number is NOT prime

Tip:

Given a number, how do I know if it is prime?

"A prime number is a positive integer that has no positive integer divisors other than 1 and itself"

Example: 7 cannot be divided by 2, 3, 4, 5, 6… therefore is prime
Example: 9 cannot be divided by 2, but it can be divided by 3, therefore is not prime

SOLUTION

```
clear;
number = input('Introduce a number: ');
isPrime = 'Y';
i = 2;
while((i <= number/2) && (isPrime == 'Y'))
    if(rem(number,i) == 0)
        isPrime = 'N';
    else
        i = i + 1;
    end
end
```

We check the all the numbers between 2 and number/2. We finish the loop either when we reach number/2 or when we find a divisor.

```
if(isPrime=='Y')
    disp('The number is prime');
else
    disp('The number is not prime');
end
```

At this point in the program the while loop has finished, but we don't know which of the two conditions of the while loop finished it (was isPrime set to 'N' or did we reach number/2?). So, we need to check one of them to get the answer. In this case, we chose to check whether isPrime = 'Y' or 'N' (alternatively, you could also check whether i <= number/2 or not).

ANOTHER SOLUTION

```
clear;
number = input('Introduce a number: ');
i = 2;
while ((i <= number/2) && (rem(number,i) ~= 0))
    i = i + 1;
end

if (i <= number/2)
    disp('The number is not prime');
else
    disp('The number is prime');
end
```

Another solution, without the use of a Boolean value in the while loop condition.