

## Solutions to Exercise Series N°: 05

### Exercise1

Write an algorithm that asks for a starting number A and displays the next n numbers (n entered by the user). For example, if the user enters the number  $a=35$ , and  $n=12$  the program will display the numbers from 35 to 46

| <b>While loop</b>  | <b>For loop</b>  |
|--|--|
| <pre> <b>Algorithm</b> Display_numbers <b>Var</b>     a,n,i : integer <b>Begin</b>     Write (" Enter the starting number: ");     Read(a)     Write (" Enter n : ");     Read(n)     i←0     while (i&lt;n) do         write (a+i)         i←i+1     endwhile <b>End</b>                 </pre> | <pre> <b>Algorithm</b> Display_numbers <b>Var</b>     a,n,i : integer <b>Begin</b>     Write (" Enter the starting number: ");     Read(a)     Write (" Enter n : ");     Read(n)     i←0     for i←a to a+n-1 step 1 do         write (i )     endFor <b>End</b>                 </pre> |

**Exercise2.** Write an algorithm that calculates the smallest common multiplier PPMC of two positive integers (without calculating the largest common divisor PGCD)

|   |  |
|---|--|
| <pre> <b>Algorithm</b> calcul_PPMC <b>Var</b>     a,b,i,p : integer <b>Begin</b>     Write (" Enter two integers: ");     Read(a,b)     <b>If</b> (a&gt;b) <b>then</b>         i←a     <b>Else</b>         i←b     <b>endIf</b>     while (i mod a ≠0 and i mod b ≠0) do         i←i+1     endwhile     Write (" PPMC=",i); <b>End</b>                 </pre> | <pre>                 // exit after finding the first common multiple                 </pre> |
|---|--|

**Exercise3.** Write an algorithm that calculates the number of digits of an integer n.

|   |
|---|
| <pre> <b>Algorithm</b> calcul_PPMC <b>Var</b>     n,c,a : integer <b>Begin</b>     Write (" Enter an integer number: ");     Read(n)     c←0     a←n     while (a ≠0) do         a←a div 10         c←c+1     endwhile     Write (" the number of digits of ",n , " is ", c); <b>End</b>                 </pre> |
|---|

**Exercise 4.** Write an algorithm that allows you to enter positive and not zero, check if this integer is prime,.

| Algo1   | Algo2 (optimized)   |
|---|---|
| <pre> <b>Algorithm</b> Is_prime <b>Var</b>     n,i,c : integer <b>Begin</b>     <b>Do</b>         Write (" Enter an integer: ");         Read(n)     <b>While</b>(n≤0)         c←0         a←1         <b>while</b> (a ≤n) <b>do</b>             <b>if</b> (n mod a=0) <b>then</b>                 c←c+1 // calculate the number of divisors             <b>endif</b>             a←a+1         <b>endWhile</b>         <b>if</b>(c=2) <b>then</b>             write (n, " is prime ")         <b>else</b>             write (n, " is not prime ")         <b>endif</b>     <b>End</b> </pre> | <pre> <b>Algorithm</b> Is_prime <b>Var</b>     n,i,c : integer <b>Begin</b>     <b>Do</b>         Write (" Enter an integer: ");         Read(n)     <b>While</b>(n≤0)         c←0         a←2         // if we find a divisor we stop         <b>while</b> (a ≤n/2 and c=0) <b>do</b>             <b>if</b> (n mod a=0) <b>then</b>                 c←1             <b>endif</b>             a←a+1         <b>endWhile</b>         <b>if</b>(c=0) <b>then</b>             write (n, " is prime ")         <b>else</b>             write (n, " is not prime ")         <b>endif</b>     <b>End</b> </pre> |

**Exercise 5.** Write a C program that calculates the sum  $S = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} \dots \pm \frac{1}{n}$  (n entered by the user)

| Algorithm   | C Program  |
|---|--|
| <pre> <b>Algorithm</b> Sum <b>Var</b>     n,i,j : integer     s:real <b>Begin</b>     Write (" Enter an integer: ");     Read(n)     j←-1     i←1     s←0     <b>for</b> i←1 <b>a</b> n <b>do</b>         s ←s +j*1/i         j ←-j     <b>endFor</b>     write ( " som= ",s) <b>End</b> </pre> | <pre> #include&lt;stdio.h&gt; int main ( ) {     int n, i, j ;     float s;     printf( "enter an integer : " );     scanf( "%d", &amp;n) ;     j=i=1;     s=0;     <b>for</b> (i=1;i&lt;=n; i++)         { s =s +j*1./i; //one is real to have real division           j =-j;         }     Printf ("som= %.2f ",s); } </pre> |

**Exercise6.** Write an algorithm that displays the number of divisors of each integer less than n

|   |
|---|
| <pre> <b>Algorithm</b> Number_of_divisors <b>Var</b> n,i,j,nbrdiv : integer <b>Begin</b> Write (" Enter an integer: ");         Read(n)         <b>for</b> i←1 to n <b>do</b>             nbrdiv ←0             <b>for</b> j←1 to i <b>do</b>                 <b>if</b> (i mod j=0) <b>then</b>                     nbrdiv ← nbrdiv +1 // calculate the number of divisors                 <b>endif</b>             <b>endFor</b>             printf ( " number of divisor of %d is : %d\n ",i,nbrdiv);         <b>endFor</b> <b>End</b> </pre> |
|---|

## Practical work

**Ex1.** Write a program that reads two integers and then displays the quotient of the first divided by the second, without using the division operator (/ or div). (Division is repeated subtraction).

```
#include <stdio.h>
int main()
{
    int x, y, q, r ;
    printf("enter 2 nbrs \n") ;
    scanf("%d%d", &x, &y) ;
    q=0 ;
    r=x ;
    while ( r>y )
    {
        r-=y ;
        q++ ;
    }
    printf("the quotient of %d over %d is %d the remainder is
%d\n"),x, y, q, r);
}
```

**Ex2.** A program that asks for a number between 10 and 20, until the answer matches. In the event of a response greater than 20, a message will appear: "Smaller! ", and conversely, "Bigger!" » if the number is less than 10.

```
#include<stdio.h>
int main ( )
{
    int n;
    do { printf ( "enter an integer:  " ) ;
        scanf ( "%d" , &n) ;
        if (n>20)
            printf("smaller \n  ");
        else if (n<10)
            printf("Bigger \n  ");
        else printf("the number is between 10 &nd 20 .... \n ");
    }
    while (n<10 && n>20);
}
```

**Ex3:** program that reads a positive integer n then calculates the sum:  $S = \frac{1}{2} + \frac{2}{3} + \frac{3}{4} + \frac{4}{5} + \dots + \frac{n-2}{n-1} + \frac{n-1}{n}$

```
#include<stdio.h>
int main ( )
{
    int n, i, j ;
    float s;
    printf ( "enter an integer :  " ) ;
    scanf ( "%d" , &n) ;
    i=1;
    s=0;
    for (i=1;i<=n; i++)
        s =s +(float)i/(i+1);
    Printf ("som= %.2f  ",s);
}
```

**Ex4.** Write an algorithm that prints a right isosceles triangle with right side length n (n entered by user)

```
#include <stdio.h>
int main()
{
    int n, i, j ;
    printf("enter no. of lines") ;
    scanf("%d", &n) ;
    for ( i=1 ;i<=n ;i++ )
    {
        for ( j=1 ;j<=i ;j++ )
            printf("*") ;
        printf("\n") ;
    }
}
```