

Examination ProgMod

October 1st, 2015

You may write your answers in Dutch or English. No documents or calculator.

1 Exercise 1

Write a simple program which

1. (0.5 pt) asks the user to input on the keyboard two real numbers (x, y) and two integers (n, m) ;
2. (0.5 pt) computes $z = x^m + y^n$ and prints z it on the screen;
3. (0.5 pt) prints 'too big' if z exceeds 1000

2 Exercise 2

1. (1 pt) Write a function which takes as argument an integer n and returns $n!$ ("factorial n ")
2. (0.5 pt) Write a small program which makes use of this function

3 Exercise 3

1. (1pt) Write a subroutine which takes as arguments two integers m, n and a real x and computes $\sqrt{x^n + x^m}$
2. (1pt) Write a small program which makes use of this subroutine

4 Exercise 4

Write a program according to the following specifications:

1. (0.5 pt) We wish to store in the arrays `xcoordinates`, `ycoordinates` the x, y coordinates of a yet unknown number of points. Declare these arrays.
2. (0.5 pt) Have the user choose the size N (i.e. the user enters this number with the keyboard) of the arrays and fill them with numbers so that all points have different coordinates.
3. (1pt) Open a file (name it as you wish) and use a do-loop to write the coordinates of these points on two columns
4. (1pt) compute the following quantities:

$$\alpha = \sum_{i=1}^N (|x_i| + |y_i|) \quad \beta = \frac{1}{N} \sqrt{\sum_{i=1}^N (x_i^2 + y_i^2)} \quad \gamma = \min_{i=1, N} y_i$$

5 Exercise 5

Euler discovered the remarkable quadratic formula: $n^2 + n + 41$. It turns out that the formula will produce 40 primes for the consecutive values $n = 0$ to 39.

1. (0.5 pt) Write a program using this formula which writes in a file the 40 primes generated.
2. (0.5 pt) how would you verify that the number you obtain for $n=40$ is not a prime number ? (do not write a code, rather explain to me the algorithm you would employ).

6 Exercise 6

1. (0.5 pt) Write a function *myweirdfunc* which takes an integer n as argument and returns: $n/2$ if n is even, $3n+1$ if n is odd
2. (1 pt) write a program which declares an integer array *mytab* of length 15 and place the value 11 in *mytab(1)*. Fill *mytab(2)* with the value you obtain by calling the function on *mytab(1)*. Fill *mytab(3)* with the value you obtain by calling the function on *mytab(2)*. Fill *mytab(4)* with the value you obtain by calling the function on *mytab(3)*, etc ... Automatize this process so as to fill the whole array this way.

What value does *mytab* contain in the end ? (the idea is to run the code in your head or on paper)



Recommendations:

- Every single used variable has to be defined.
- Is it an integer ? a real ? a static array ? an allocatable array ? etc ...
- Comment your code appropriately.
- Points will be deducted for unclear/unreadable statements.
- Every single used variable has to be defined (I insist).