INTRODUCIÓN A LA PROGRAMACIÓN
PRÁTICO – 5

(En la letra de cada ejercicio, se debe declarar todas las variables necesarias y adecuadas para la resolución del mismo.)

1 – a ) Escribir un programa fortran que convierta un rango de temperaturas en grados Celsius a otro de temperatura en grados Fahrenheit según la fórmula:
Tf = (9/5)*Tc + 32. y lo despliegue en pantalla.

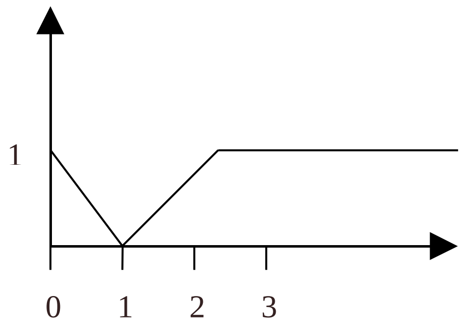
b) Escribir un programa fortran que calcule el factorial de un número dado y que contemple que factorial de 0 es 1. Se deben contemplar factoriales de números mayores a 3e10.

c) Escribir un programa para calcular la suma: $1+x+x^2+x^3+\ldots+x^n$ con x y n dados.

d) Idem la parte d) pero con la siguiente suma: $a_0 + a_1*x + a_2*x^2 + \ldots + a_n*x^n$ siendo $a_n=(2*n+1)/(3*n+1)$

2 – a) Escribir un programa que imprima todos los números enteros en el rango de 1 a 30 salvo los que son múltiplos de 3 (sugerencia: usar la función mod que da el resto de una división entera).

b) Escribir un programa que evalúe e imprima la siguiente función:




f(x) = 1-x , si x pertenece a [0,1)
-1+x , si x pertenece a [1,2)
1 , si x es mayor o igual a 2

La función debe ser evaluada a intervalos de h=0.1 desde 0 hasta 3

c) Escribir un programa que calcule las combinaciones de “n” en “k” según la fórmula: combinaciones(n,k) = (n!)/(k!*(n-k)!). Escribirlo con un solo loop.

d) La derivada de una función f(x), se define como el límite para h->0 de :

(f(x+h)-f(x))/h

Una aproximación a la derivada viene de tomar h finito, cuanto más pequeño, mejor la aproximación. Si f(x)=sin(x) y x=0, escribir un programa que devuelva la diferencia entre cos(0) (o sea 1) y las aproximaciones de la derivada de f(x) en x=0 con h desde 0.1 hasta 0.01 y variando de a intervalos de 0.01.

e) Escribir un programa para saber si un entero positivo dado es primo o no (usar la función intrínseca mod(n,m), el resto de una división entera).

f) La sucesión de Fibonacci se define por la secuencia de números enteros positivos dada por: fib(n) = fib(n-1)+fib(n-2) , fib(0)=1 y fib(1)=1, para un n dado. Escribir un programa que imprima la sucesión de Fibonacci para n dado.

3) Este ejercicio pantea el uso de varios loops encadenados.

a) Escribir un programa que devuelva el valor del binomio de Newton en base a su desarrollo:

$$(a+b)^n = comb(n,0)a^n b^0 + comb(n,1)a^{n-1}b^1 + \ldots + comb(n+k,k)a^{n-k}b^k + comb(n,n)a^0 b^n$$

comb(n,k) es la combinaciones de n tomadas de k. (Usar las parte 2c)

b) Escribir un programa que permita saber cual es el máximo número n de precisión simple para el cual el factorial de n no da overflow. (Eventualmente el programa da overflowpor lo que se debe ir imprimiendo el nro n).

c) Escribir un programa que imprima N lineas en la consola. La línea nro. K debe imprimir todos los números de 1 hasta k. La salida del programa debe ser la siguiente:

```
1
1 2
1 2 3
…
1 2 3 4 5…N
```

d) Escribir un programa que imprima todas las ternas de números naturales tal que la suma de cada terna es menor o igual a un número N dado. (Probar con N=5).

e) Dada la función de dos variables f(x,y)=(x^2+y^2), imprimir los valores de la función, para todos los pares de (x,y) pertenecientes a ([0,1],[0.1]) variando las variables x con incrementos de h=0.1 y la y con incrementos de k=0.1. La primera fila impresa debe contener los valores de x en [0.1] para y=0, la segunda los valores de x en [0,1] para y=0.1, etc.