

# Programación en FORTRAN

## Nivel Básico - Sesión 1

---

Martin Josemaría Vuelta Rojas

8 de diciembre de 2017

SoftButterfly

1. Preliminares
2. Introducción histórica

# Preliminares

---

El curso básico de Fortran tiene por objetivo

El curso básico de Fortran tiene por objetivo

- Que obtengan un conocimiento sólido de las características de Fortran.

## El curso básico de Fortran tiene por objetivo

- Que obtengan un conocimiento sólido de las características de Fortran.
- Que se familiarizen con el flujo de información en los programas desarrollados en Fortran.

## El curso básico de Fortran tiene por objetivo

- Que obtengan un conocimiento sólido de las características de Fortran.
- Que se familiarizen con el flujo de información en los programas desarrollados en Fortran.
- Que tengan una introducción nuevas características de los estándares recientes Fortran 2003 y 2008.

## El curso básico de Fortran tiene por objetivo

- Que obtengan un conocimiento sólido de las características de Fortran.
- Que se familiaricen con el flujo de información en los programas desarrollados en Fortran.
- Que tengan una introducción a nuevas características de los estándares recientes Fortran 2003 y 2008.
- Que puedan contruir su entorno de desarrollo para programar en Fortran cómodamente.



Metodología

## Metodología

- Exposiciones dialogadas utilizando, apuntes, elementos de proyección fija.

## Metodología

- Exposiciones dialogadas utilizando, apuntes, elementos de proyección fija.
- Talleres prácticos grupales en la elaboración de programas.

## Metodología

- Exposiciones dialogadas utilizando, apuntes, elementos de proyección fija.
- Talleres prácticos grupales en la elaboración de programas.
- Las clases serán eminentemente prácticas en una relación 70% práctica - 30% teórica.

## Referencias

## Referencias

- I. Chivers, J. Sleightholme, *Introduction to Programming with Fortran. With Coverage of Fortran 90, 95, 2003, 2008 and 77*, Springer-Verlag London, 2012.

## Referencias

- I. Chivers, J. Sleightholme, *Introduction to Programming with Fortran. With Coverage of Fortran 90, 95, 2003, 2008 and 77*, Springer-Verlag London, 2012.
- Michael Metcalf, John Reid, Malcolm Cohen, *Modern Fortran Explained*, Oxford University Press, USA 2011

## Referencias

- I. Chivers, J. Sleightholme, *Introduction to Programming with Fortran. With Coverage of Fortran 90, 95, 2003, 2008 and 77*, Springer-Verlag London, 2012.
- Michael Metcalf, John Reid, Malcolm Cohen, *Modern Fortran Explained*, Oxford University Press, USA 2011
- Morten Hjorth-Jensen, *Computational Physics, Lecture Notes Fall 2015*, Department of Physics, University of Oslo. 2015.



Materiales

## Materiales

Repositorio del curso

## Materiales

Repositorio del curso

 <https://github.com/zodiacfireworks/course-fortran-basic>

## Materiales

### Repositorio del curso

- 🐙 <https://github.com/zodiacfireworks/course-fortran-basic>
- 🐙 <https://github.com/zodiacfireworks/course-fortran-intermediate>

## Materiales

Repositorio del curso

- 🐙 <https://github.com/zodiacfireworks/course-fortran-basic>
- 🐙 <https://github.com/zodiacfireworks/course-fortran-intermediate>

¡Importante!

## Materiales

Repositorio del curso

- 🐙 <https://github.com/zodiacfireworks/course-fortran-basic>
- 🐙 <https://github.com/zodiacfireworks/course-fortran-intermediate>

## ¡Importante!

Cada alumno debera tener una cuenta en GitHub

## Materiales

Repositorio del curso

- 🔗 <https://github.com/zodiacfireworks/course-fortran-basic>
- 🔗 <https://github.com/zodiacfireworks/course-fortran-intermediate>

## ¡Importante!

Cada alumno debera tener una cuenta en GitHub

- 🔗 <https://github.com>

Martín Josemaría Vuelta Rojas



Martín Josemaría Vuelta Rojas

- Software Developer

Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX
  - En web:

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX
  - En web:
    - HTML, CSS, JavaScript, Python



## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX
  - En web:
    - HTML, CSS, JavaScript, Python
  - En mobile:

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX
  - En web:
    - HTML, CSS, JavaScript, Python
  - En mobile:
    - Kotlin, Java, C++

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX
  - En web:
    - HTML, CSS, JavaScript, Python
  - En mobile:
    - Kotlin, Java, C++
  - Hobbie:

## Martín Josemaría Vuelta Rojas

- Software Developer
- Web Developer
- Investigador
- Programador
  - En investigación:
    - C, C++, Fortran, Python, R, Julia, Mathematica, Matlab, LaTeX
  - En web:
    - HTML, CSS, JavaScript, Python
  - En mobile:
    - Kotlin, Java, C++
  - Hobbie:
    - Scala, Pixie, Clojure, Elixir, Haskell, Oz, Kotlin, ...

Martín Josemaría Vuelta Rojas

Martín Josemaría Vuelta Rojas

- SoftButterfly

Martín Josemaría Vuelta Rojas

- SoftButterfly
- HackSpace Perú

Martín Josemaría Vuelta Rojas

- SoftButterfly
- HackSpace Perú
- Jupyter Notebook



Martín Josemaría Vuelta Rojas

- SoftButterfly
- HackSpace Perú
- Jupyter Notebook
- Fedora

## Martín Josemaría Vuelta Rojas

- SoftButterfly
- HackSpace Perú
- Jupyter Notebook
- Fedora
- GNOME

## Martín Josemaría Vuelta Rojas

- SoftButterfly
- HackSpace Perú
- Jupyter Notebook
- Fedora
- GNOME
- UNMSM

Martín Josemaría Vuelta Rojas

Martín Josemaría Vuelta Rojas

📱 +51 982 042 088

Martín Josemaría Vuelta Rojas

📱 +51 982 042 088

✉ [martin.vuelta@gmail.com](mailto:martin.vuelta@gmail.com)

Martín Josemaría Vuelta Rojas

📞 +51 982 042 088

✉️ martin.vuelta@gmail.com

**in** martinvuelta

Martín Josemaría Vuelta Rojas

📱 +51 982 042 088

✉️ martin.vuelta@gmail.com

**in** martinvuelta

🌀 zodiacfireworks



## Introducción histórica

---

En el origen ...

En el origen ...

- Código máquina en notación octal

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

# Orígenes

En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

A inicios de los 50s

# Orígenes

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler
- Menos laborioso que el código máquina

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler
- Menos laborioso que el código máquina
- Conocimiento detallado del hardware



## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler
- Menos laborioso que el código máquina
- Conocimiento detallado del hardware

## El panorama general ...

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler
- Menos laborioso que el código máquina
- Conocimiento detallado del hardware

## El panorama general ...

- Conocimiento del hardware

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler
- Menos laborioso que el código máquina
- Conocimiento detallado del hardware

## El panorama general ...

- Conocimiento del hardware
- Facilidad de cometer errores

## En el origen ...

- Código máquina en notación octal
- Conocimiento muy detallado del hardware

## A inicios de los 50s

- Assembler
- Menos laborioso que el código máquina
- Conocimiento detallado del hardware

## El panorama general ...

- Conocimiento del hardware
- Facilidad de cometer errores
- Encontrar los errores en los programas era bastante difícil

Génesis 1953 ...

## Génesis 1953 ...

John Backus envía una carta a su jefe en IBM pidiendo permiso para investigar una *mejor forma* de programar las computadoras. La carta contenía un esbozo de proyecto con un tiempo de desarrollo de 6 meses.

## Génesis 1953 ...

John Backus envía una carta a su jefe en IBM pidiendo permiso para investigar una *mejor forma* de programar las computadoras. La carta contenía un esbozo de proyecto con un tiempo de desarrollo de 6 meses.

Así empezó el proyecto que daría origen a Fortran

*“The project completion was always six months away!”*  
*John Backus*



1957

1957

En febrero FORTRAN, el primer lenguaje de programación de alto nivel, fue anunciado al mundo por John Backus y su equipo de IBM en la Western Joint Computer Conference celebrada en Los Ángeles.

1957

En febrero FORTRAN, el primer lenguaje de programación de alto nivel, fue anunciado al mundo por John Backus y su equipo de IBM en la Western Joint Computer Conference celebrada en Los Ángeles.

A mediados de abril de 1957 tuvo lugar la primera entrega del compilador de FORTRAN para IBM 704 a Westinghouse Bettis para su uso en el diseño de reactores nucleares.

Las versiones de FORTRAN

## Las versiones de FORTRAN

1957 FORTRAN I

## Las versiones de FORTRAN

1957 FORTRAN I

1958 FORTRAN II

## Las versiones de FORTRAN

1957 FORTRAN I

1958 FORTRAN II

1958 FORTRAN III (No disponible al público)

## Las versiones de FORTRAN

**1957** FORTRAN I

**1958** FORTRAN II

**1958** FORTRAN III (No disponible al público)

**1961** FORTRAN IV (Una versión mejorada de FORTRAN II)



## Algoritmo TPK en FORTRAN 0

```
1      DIMENSION A(11)
2      READ A
3      2      DO 3,8,11 J=1,11
4      3      I=11-J
5      Y=SQRT(ABS(A(I+1)))+5*A(I+1)**3
6      IF (400>=Y) 8,4
7      4      PRINT I,999.
8      GOTO 2
9      8      PRINT I,Y
10     11     STOP
```

## Algoritmo TPK en FORTRAN I

```
1  C      THE TPK ALGORITHM
2  C      FORTRAN I STYLE
3          FUNF(T)=SQRTF(ABSF(T))+5.0*T**3
4          DIMENSION A(11)
5  1      FORMAT(6F12.4)
6          READ 1,A
7          DO 10 J=1,11
8              I=11-J
9              Y=FUNF(A(I+1))
10         IF(400.0-Y)4,8,8
11  4      PRINT 5,I
12  5      FORMAT(I10,10H TOO LARGE)
13         GOTO 10
14  8      PRINT 9,I,Y
15  9      FORMAT(I10,F12.7)
16 10     CONTINUE
17     STOP 52525
```

# La estandarización

1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.

# La estandarización

- 1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.
- 1966 Publicación del ANSI X3.91966 (FORTRAN 66), el primer estándar.

# La estandarización

- 1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.
- 1966 Publicación del ANSI X3.91966 (FORTRAN 66), el primer estándar.
- 1978 Publicación del ANSI X3.91978 (FORTRAN 77), también publicado como ISO 1539:1980.

# La estandarización

- 1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.
- 1966 Publicación del ANSI X3.91966 (FORTRAN 66), el primer estándar.
- 1978 Publicación del ANSI X3.91978 (FORTRAN 77), también publicado como ISO 1539:1980.
- 1991 ISO/IEC 1539:1991 (Fortran 90)

# La estandarización

- 1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.
- 1966 Publicación del ANSI X3.91966 (FORTRAN 66), el primer estándar.
- 1978 Publicación del ANSI X3.91978 (FORTRAN 77), también publicado como ISO 1539:1980.
- 1991 ISO/IEC 1539:1991 (Fortran 90)
- 1997 ISO/IEC 15391:1997 (Fortran 95)



# La estandarización

- 1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.
- 1966 Publicación del ANSI X3.9:1966 (FORTRAN 66), el primer estándar.
- 1978 Publicación del ANSI X3.9:1978 (FORTRAN 77), también publicado como ISO 1539:1980.
- 1991 ISO/IEC 1539:1991 (Fortran 90)
- 1997 ISO/IEC 1539:1997 (Fortran 95)
- 2004 ISO/IEC 1539:2004 (Fortran 2003)

# La estandarización

- 1962 El primer comité de estandarización de la ASA (Ahora ANSI) se reúne.
- 1966 Publicación del ANSI X3.9:1966 (FORTRAN 66), el primer estándar.
- 1978 Publicación del ANSI X3.9:1978 (FORTRAN 77), también publicado como ISO 1539:1980.
- 1991 ISO/IEC 1539:1991 (Fortran 90)
- 1997 ISO/IEC 1539:1997 (Fortran 95)
- 2004 ISO/IEC 1539:2004 (Fortran 2003)
- 2010 ISO/IEC 1539:2010 (Fortran 2008)

# La estandarización

## Algoritmo TPK en FORTRAN 77

```
1      PROGRAM TPK
2      C      THE TPK ALGORITHM
3      C      FORTRAN 77 STYLE
4      REAL A(0:10)
5      READ (5,*) A
6      DO 10 I = 10, 0, -1
7          Y = FUN(A(I))
8          IF ( Y . LT. 400) THEN
9              WRITE(6,9) I,Y
10             FORMAT(I10, F12.6)
11             ELSE
12                 WRITE (6,5) I
13                 FORMAT(I10, ' TOO LARGE')
14             ENDIF
15 10     CONTINUE
16     END
17
18     REAL FUNCTION FUN(T)
19     REAL T
20     FUN = SQRT(ABS(T)) + 5.0*T**3
21     END
```

# La estandarización

## Algoritmo TPK en FORTRAN 90

```
1      PROGRAM TPK
2      !      The TPK Algorithm
3      !      Fortran 90 style
4      IMPLICIT NONE
5      INTEGER          :: I
6      REAL             :: Y
7      REAL, DIMENSION(0:10) :: A
8      READ (*,*) A
9      DO I = 10, 0, -1      ! Backwards
10         Y = FUN(A(I))
11         IF ( Y < 400.0 ) THEN
12             WRITE(*,*) I, Y
13         ELSE
14             WRITE(*,*) I, ' Too large'
15         END IF
16     END DO
17     CONTAINS                ! Local function
18     FUNCTION FUN(T)
19     REAL :: FUN
20     REAL, INTENT(IN) :: T
21     FUN = SQRT(ABS(T)) + 5.0*T**3
22     END FUNCTION FUN
23 END PROGRAM TPK
```

# La estandarización

## Algoritmo TPK en FORTRAN 90

```
1      module Functions
2      public :: fun
3      contains
4          function fun(t) result (r)
5              real, intent(in) :: t
6              real :: r
7              r = sqrt(abs(t)) + 5.0*t**3
8          end function fun
9      end module Functions
10
11      program TPK
12      ! The TPK Algorithm
13      ! F95 style
14      use Functions
15      integer      :: i
16      real          :: y
17      real, dimension(0:10) :: a
18      read *, a
19      do i = 10, 0, -1      ! Backwards
20          y = fun(a(i))
21          if ( y < 400.0 ) then
22              print *, i, y
23          else
24              print *, i, " Too large"
```

*"I don't know what the programming language of the year 2000 will look like, but I know it will be called FORTRAN."*

*Charles Anthony Richard Hoare*

*Circa 1982*

Aplicaciones

## Aplicaciones

- Predicción del clima



## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo

## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo
- Análisis financiero

## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo
- Análisis financiero
- Simulación de choques vehiculares

## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo
- Análisis financiero
- Simulación de choques vehiculares
- Análisis de datos de sondas espaciales

## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo
- Análisis financiero
- Simulación de choques vehiculares
- Análisis de datos de sondas espaciales
- Modelación de armas nucleares

## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo
- Análisis financiero
- Simulación de choques vehiculares
- Análisis de datos de sondas espaciales
- Modelación de armas nucleares
- Dinámica de fluidos computacionales

## Aplicaciones

- Predicción del clima
- Análisis de datos de sísmicos para la exploración de depósitos de gas y petróleo
- Análisis financiero
- Simulación de choques vehiculares
- Análisis de datos de sondas espaciales
- Modelación de armas nucleares
- Dinámica de fluidos computacionales
- “Numerical Wind Tunnel”

En la actualidad ...



- Intel

- Intel
- IBM

- Intel
- IBM
- NVIDIA



- WG5: <https://wg5-fortran.org/>

- WG5: <https://wg5-fortran.org/>
- Fortran Wiki: <http://fortranwiki.org/>

- WG5: <https://wg5-fortran.org/>
- Fortran Wiki: <http://fortranwiki.org/>
- WG5: <https://wg5-fortran.org/>