



Πανεπιστήμιο Αιγαίου

Εισαγωγή στην Πληροφορική

Ενότητα 5: Εισαγωγή στην Γλώσσα Προγραμματισμού
Fortran 90

Ανδρέας Παπασαλούρος

Τμήμα Μαθηματικών

Σάμος, Μάιος 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αιγαίου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Εισαγωγή

Εισαγωγή

Η Fortran είναι μία από τις παλαιότερες γλώσσες προγραμματισμού υψηλού επιπέδου.

Εισαγωγή

Η Fortran είναι μία από τις παλαιότερες γλώσσες προγραμματισμού υψηλού επιπέδου.

Προτάθηκε το 1954 από τον John Backus και τους συνεργάτες του στην εταιρεία IBM.

Εισαγωγή

Η Fortran είναι μία από τις παλαιότερες γλώσσες προγραμματισμού υψηλού επιπέδου.

Προτάθηκε το 1954 από τον John Backus και τους συνεργάτες του στην εταιρεία IBM.

Το όνομα Fortran αναφέρεται στην δυνατότητα της γλώσσας για αναπαράσταση μαθηματικών τύπων (The IBM Mathematical FORMula TRANslating system), χαρακτηριστικό καινοτόμο στην εποχή του.

Εισαγωγή

Η Fortran είναι μία από τις παλαιότερες γλώσσες προγραμματισμού υψηλού επιπέδου.

Προτάθηκε το 1954 από τον John Backus και τους συνεργάτες του στην εταιρεία IBM.

Το όνομα Fortran αναφέρεται στην δυνατότητα της γλώσσας για αναπαράσταση μαθηματικών τύπων (The IBM Mathematical FORmula TRANslating system), χαρακτηριστικό καινοτόμο στην εποχή του.

Η επίδοση των προγραμμάτων Fortran ήταν ανάλογη αυτών σε συμβολική γλώσσα.

Η Fortran αναπτύχθηκε ως γλώσσα επιστημονικού υπολογισμού.

Ένα απλό πρόγραμμα Fortran:

```
PROGRAM first
```

```
! Ένα απλό πρόγραμμα Fortran
```

```
PRINT*, "Καλημέρα"
```

```
END PROGRAM first
```


Ένα απλό πρόγραμμα Fortran:

```
PROGRAM first
```

```
! Ένα απλό πρόγραμμα Fortran
```

```
PRINT*, "Καλημέρα"
```

```
END PROGRAM first
```

Δεσμευμένες λέξεις και συμβολικά ονόματα

Δεσμευμένες λέξεις και συμβολικά ονόματα

Οι λέξεις PROGRAM και END έχουν ειδική σημασία σε όλα τα προγράμματα της Fortran.

Κανόνες για τα συμβολικά ονόματα

Κανόνες για τα συμβολικά ονόματα

Ένα συμβολικό όνομα στη Fortran πρέπει να πληρεί τους παρακάτω κανόνες:

Συμβολικά ονόματα

Έγκυρα συμβολικά ονόματα

```
x1  
one_for_me
```

Μη έγκυρα συμβολικά ονόματα

```
1forme  
_one_for_you  
one to three  
one$and&other
```

Σχόλια

Η γραμμή

```
! Ένα απλό πρόγραμμα Fortran
```

του παραπάνω προγράμματος αποτελείται από κείμενο το οποίο ακολουθεί ένα θαυμαστικό (!) και είναι ένα σχόλιο (comment).

Σταθερές

Έστω το πρόγραμμα:

```
program second

! Τύπωσε δεδομένα στην οθόνη
print*, 12
print*, 3.14159
print*, 6.023E23
print*, ' My name is John'
print*, 'c'

end program second
```

Σταθερές και τύποι δεδομένων

Σταθερές και τύποι δεδομένων

Τιμές δεδομένων που εμφανίζονται μέσα στον κώδικα ενός προγράμματος, όπως τα 12, 3.14159, κ.λπ., του παραπάνω προγράμματος ονομάζονται σταθερές του προγράμματος.

Σταθερές και τύποι δεδομένων

Τιμές δεδομένων που εμφανίζονται μέσα στον κώδικα ενός προγράμματος, όπως τα 12, 3.14159, κ.λπ., του παραπάνω προγράμματος ονομάζονται σταθερές του προγράμματος. Οι σταθερές είναι δυνατόν να ανήκουν σε διαφορετικούς τύπους δεδομένων, για παράδειγμα, ακεραίους ή πραγματικούς αριθμούς. Για τους κυριότερους τύπους στη Fortran θα γίνει αναφορά στη συνέχεια.

Μεταβλητές

Στα προγράμματα χρησιμοποιούνται *συμβολικά ονόματα* για το χειρισμό των δεδομένων. Έστω το ακόλουθο πρόγραμμα:

```
program third

! Δήλωση μιας μεταβλητής με όνομα  apotelesma
integer:: apotelesma

apotelesma = 12

print*, apotelesma

end program third
```

Μεταβλητές

Στα προγράμματα χρησιμοποιούνται *συμβολικά ονόματα* για το χειρισμό των δεδομένων. Έστω το ακόλουθο πρόγραμμα:

```
program third

! Δήλωση μιας μεταβλητής με όνομα  apotelesma
integer:: apotelesma

apotelesma = 12

print*, apotelesma

end program third
```

Μεταβλητές

Στα προγράμματα χρησιμοποιούνται *συμβολικά ονόματα* για το χειρισμό των δεδομένων. Έστω το ακόλουθο πρόγραμμα:

```
program third

! Δήλωση μιας μεταβλητής με όνομα  apotelesma
integer:: apotelesma

apotelesma = 12

print*, apotelesma

end program third
```

Μεταβλητές

Στο παραπάνω παράδειγμα, το `apotelesma` είναι ένα *συμβολικό όνομα* δηλώνεται από τον προγραμματιστή.

Μεταβλητές

Στο παραπάνω παράδειγμα, το `apotelesma` είναι ένα *συμβολικό όνομα* δηλώνεται από τον προγραμματιστή.
Στο όνομα αυτό θα ανατίθενται *ακέραιες* τιμές.

Μεταβλητές

Στο παραπάνω παράδειγμα, το `apotelesma` είναι ένα *συμβολικό όνομα* δηλώνεται από τον προγραμματιστή.

Στο όνομα αυτό θα ανατίθενται *ακέραιες* τιμές.

Το συμβολικό όνομα `apotelesma` είναι το όνομα μιας *μεταβλητής* τύπου *ακεραίου*

```
apotelesma = 12
```

Χαρακτηριστικά μεταβλητών

Τα κυριότερα χαρακτηριστικά μιας μεταβλητής είναι:

Χαρακτηριστικά μεταβλητών

Τα κυριότερα χαρακτηριστικά μιας μεταβλητής είναι:

Το όνομα Τα ονόματα των μεταβλητών ακολουθούν τους κανόνες για τα συμβολικά ονόματα στη Fortran οι οποίοι παρουσιάστηκαν προηγουμένως.

Χαρακτηριστικά μεταβλητών

Τα κυριότερα χαρακτηριστικά μιας μεταβλητής είναι:

Το όνομα Τα ονόματα των μεταβλητών ακολουθούν τους κανόνες για τα συμβολικά ονόματα στη Fortran οι οποίοι παρουσιάστηκαν προηγουμένως. Αποτελεί καλή πρακτική τα ονόματα των μεταβλητών να αποδίδουν τη σημασία τους.

Χαρακτηριστικά μεταβλητών

Τα κυριότερα χαρακτηριστικά μιας μεταβλητής είναι:

- Το όνομα** Τα ονόματα των μεταβλητών ακολουθούν τους κανόνες για τα συμβολικά ονόματα στη Fortran οι οποίοι παρουσιάστηκαν προηγουμένως. Αποτελεί καλή πρακτική τα ονόματα των μεταβλητών να αποδίδουν τη σημασία τους.
- Ο τύπος** Είναι το είδος των δεδομένων στα οποία αναφέρεται μια μεταβλητή, π.χ. ακέραιος, πραγματικός αριθμός, ή χαρακτήρας.

Χαρακτηριστικά μεταβλητών

Τα κυριότερα χαρακτηριστικά μιας μεταβλητής είναι:

- Το όνομα** Τα ονόματα των μεταβλητών ακολουθούν τους κανόνες για τα συμβολικά ονόματα στη Fortran οι οποίοι παρουσιάστηκαν προηγουμένως. Αποτελεί καλή πρακτική τα ονόματα των μεταβλητών να αποδίδουν τη σημασία τους.
- Ο τύπος** Είναι το είδος των δεδομένων στα οποία αναφέρεται μια μεταβλητή, π.χ. ακέραιος, πραγματικός αριθμός, ή χαρακτήρας.
- Η τιμή** Αναφέρεται στα δεδομένα μιας μεταβλητής.

Τύποι στη Fortran

Οι κυριότεροι τύποι δεδομένων στη γλώσσα Fortran είναι οι ακόλουθοι:

Τύποι στη Fortran

Οι κυριότεροι τύποι δεδομένων στη γλώσσα Fortran είναι οι ακόλουθοι:

- ακέραιοι

Τύποι στη Fortran

Οι κυριότεροι τύποι δεδομένων στη γλώσσα Fortran είναι οι ακόλουθοι:

- ακέραιοι
- πραγματικοί

Τύποι στη Fortran

Οι κυριότεροι τύποι δεδομένων στη γλώσσα Fortran είναι οι ακόλουθοι:

- ακέραιοι
- πραγματικοί
- χαρακτήρες

Τύποι στη Fortran

Οι κυριότεροι τύποι δεδομένων στη γλώσσα Fortran είναι οι ακόλουθοι:

- ακέραιοι
- πραγματικοί
- χαρακτήρες
- λογικοί

Ακέραιοι: Ο τύπος `integer`

Τύπος ακεραίων αριθμών, θετικών και αρνητικών.

Παραδείγματα ακεραίων σταθερών αποτελούν τα παρακάτω:

1

3

-343

32343

+233

Οι παρακάτω γραμμές αποτελούν δηλώσεις ακεραίων μεταβλητών:

```
integer:: aNumber
```

```
integer:: amount, total
```

Ακέραιοι

Ακέραιοι

Οι αριθμοί του τύπου `integer` παρίστανται εσωτερικά στον υπολογιστή σε μορφή *συμπληρώματος ως προς 2*

Ακέραιοι

Οι αριθμοί του τύπου `integer` παρίστανται εσωτερικά στον υπολογιστή σε μορφή *συμπληρώματος ως προς 2*

Το μήκος (αριθμός bit) της παράστασης εξαρτάται από το μήκος της λέξης του υπολογιστή, δηλαδή τον μικρότερο αριθμό bit για την αναπαράσταση μιας μονάδας δεδομένων από τον υπολογιστή.

Είναι δυνατός ο ρητός ορισμός των bytes της αναπαράστασης του αριθμού όπως φαίνεται στα παρακάτω παραδείγματα

```
integer(1) :: x ! Αναπαράσταση ακέραιου με 8 bit
```

Ακέριοι

Οι αριθμοί του τύπου `integer` παρίστανται εσωτερικά στον υπολογιστή σε μορφή *συμπληρώματος ως προς 2*

Το μήκος (αριθμός bit) της παράστασης εξαρτάται από το μήκος της λέξης του υπολογιστή, δηλαδή τον μικρότερο αριθμό bit για την αναπαράσταση μιας μονάδας δεδομένων από τον υπολογιστή.

Είναι δυνατός ο ρητός ορισμός των bytes της αναπαράστασης του αριθμού όπως φαίνεται στα παρακάτω παραδείγματα

```
integer(1) :: x ! Αναπαράσταση ακέραιου με 8 bit  
integer(2) :: y ! Αναπαράσταση ακέραιου με 16 bit
```

Ακέριοι

Οι αριθμοί του τύπου `integer` παρίστανται εσωτερικά στον υπολογιστή σε μορφή *συμπληρώματος ως προς 2*

Το μήκος (αριθμός bit) της παράστασης εξαρτάται από το μήκος της λέξης του υπολογιστή, δηλαδή τον μικρότερο αριθμό bit για την αναπαράσταση μιας μονάδας δεδομένων από τον υπολογιστή.

Είναι δυνατός ο ρητός ορισμός των bytes της αναπαράστασης του αριθμού όπως φαίνεται στα παρακάτω παραδείγματα

```
integer(1) :: x ! Αναπαράσταση ακέραιου με 8 bit
integer(2) :: y ! Αναπαράσταση ακέραιου με 16 bit
integer(4) :: z ! Αναπαράσταση ακέραιου με 32 bit
```

Παρατηρήστε ότι ο αριθμός των bytes είναι δύναμη του 2.

Πραγματικοί: Ο τύπος `real`

Οι αριθμοί του τύπου `real` παριστάνονται εσωτερικά στον υπολογιστή σύμφωνα με την παράσταση κινητής υποδιαστολής απλής ακρίβειας (με 32 bit)

Πραγματικοί: Ο τύπος `real`

Οι αριθμοί του τύπου `real` παριστάνονται εσωτερικά στον υπολογιστή σύμφωνα με την παράσταση κινητής υποδιαστολής απλής ακρίβειας (με 32 bit)

Σταθερές `real` είναι:

`-1.23e-12`

`1.`

`.23`

`-0.1`

`1e-1`

`3.141592653`

Πραγματικοί: Ο τύπος `real`

Οι αριθμοί του τύπου `real` παριστάνονται εσωτερικά στον υπολογιστή σύμφωνα με την παράσταση κινητής υποδιαστολής απλής ακρίβειας (με 32 bit)

Σταθερές `real` είναι:

`-1.23e-12`

`1.`

`.23`

`-0.1`

`1e-1`

`3.141592653`

Μεταβλητές απλής ακρίβειας:

```
real :: pressure
```

```
real :: amount, total
```

Πραγματικοί διπλής ακρίβειας

Ο τύπος διπλής ακρίβειας χρησιμοποιείται όταν υπάρχει απαίτηση για μέγιστη δυνατή ακρίβεια στην αναπαράσταση ενός πραγματικού αριθμού. Οι αριθμοί διπλής ακρίβειας παρίστανται εσωτερικά με την παράσταση κινητής υποδιαστολής διπλής ακρίβειας η οποία χρησιμοποιεί 64 bit ή 8 byte.

Σταθερές διπλής υποδιαστολής είναι οι ακόλουθες:

-1.23D-12

343.012_8

.23_8

-0.1_8

3._8

1E-1_8

Οι παραπάνω σταθερές παρατηρήστε ότι διαφέρουν από τις σταθερές απλής ακρίβειας στα παρακάτω: Είτε έχουν την

Μιγαδικοί: Ο τύπος complex

Ένας μιγαδικός παριστάνεται ως
(πραγματικό μέρος, φανταστικό μέρος).

Μιγαδικοί: Ο τύπος complex

Ένας μιγαδικός παριστάνεται ως

(πραγματικό μέρος, φανταστικό μέρος).

Ο $2.5 + 3.4i$ παριστάνεται ως $(2.5, 3.4)$.

Παραδείγματα σταθερών μιγαδικού τύπου είναι:

$(1., 3.4)$! 0 αριθμός $1 + 3.4i$

$(2.5, .99e-2)$! 0 αριθμός $2.5 + 0.0099i$

Οι παρακάτω γραμμές αναφέρονται στη δήλωση μιας μεταβλητής, c , τύπου μιγαδικού αριθμού και στην ανάθεση της τιμής $1 + 2i$ στη μεταβλητή αυτή:

```
COMPLEX :: c
```

```
c = (1,2)
```

Χαρακτήρες: Ο τύπος character

Ο τύπος χαρακτήρα χρησιμοποιείται για το χειρισμό μεμονωμένων χαρακτήρων όσο και για το χειρισμό σειρών χαρακτήρων (συμβολοσειρών). Οι σταθερές αυτού του τύπου, μεμονωμένοι χαρακτήρες και συμβολοσειρές, περικλείονται από απλά ή διπλά εισαγωγικά.

Παραδείγματα μεμονωμένων χαρακτήρων είναι:

'A'

"1"

Παραδείγματα συμβολοσειρών:

"Maria"

'Good Morning'

Δηλώσεις χαρακτήρων

Δηλώσεις μεταβλητών μεμονωμένων χαρακτήρων:

```
character      :: middle_name  
character(1)   :: m_name
```

Δηλώσεις συμβολοσειρών μεγέθους n:

```
character(16)  :: first_name  
character(32)  :: last_name
```

Στο παραπάνω παράδειγμα, η μεταβλητή `first_name` είναι μια συμβολοσειρά 16 χαρακτήρων, ενώ η `last_name` είναι μια συμβολοσειρά 32 χαρακτήρων.

Το κενό ' ' είναι ένας ειδικός χαρακτήρας. Έτσι, η συμβολοσειρά 'Το παρόν' είναι διαφορετική από τη συμβολοσειρά 'Τοπαρόν'.

Λογικές τιμές: Ο τύπος `logical`

Ορίζονται δύο βασικές τιμές για τον λογικό τύπο: `.true.` και `.false.`

Μεταβλητές του τύπου `logical` είναι δυνατόν να πάρουν μόνο μια από τις δύο παραπάνω τιμές. Παραδείγματα δηλώσεων λογικών μεταβλητών είναι τα ακόλουθα:

```
logical :: is_married, has_children  
is_married = .true.  
has_children = .false.
```

Δηλώσεις μεταβλητών

- Η δήλωση μιας μεταβλητής είναι μια ειδική εντολή η οποία αντιστοιχίζει το όνομα της μεταβλητής στον τύπο της εκάστοτε μεταβλητής.
- Η δήλωση μιας μεταβλητής δεσμεύει κατάλληλο χώρο στο μνήμη του υπολογιστή για την αποθήκευση των δεδομένων που αντιστοιχούν στην τιμή της μεταβλητής. Η δήλωση μιας μεταβλητής έχει τη μορφή:

τύπος μεταβλητής :: όνομα μεταβλητής

- Η δήλωση μιας μεταβλητής γίνεται πριν τη χρήση της.

Ρητές δηλώσεις μεταβλητών

- Στη Fortran είναι δυνατή η χρήση μιας μεταβλητής χωρίς την προηγούμενη δήλωση του τύπου της.

Ρητές δηλώσεις μεταβλητών

- Στη Fortran είναι δυνατή η χρήση μιας μεταβλητής χωρίς την προηγούμενη δήλωση του τύπου της.
- Στην περίπτωση αυτή, ο τύπος της μεταβλητής καθορίζεται από το αρχικό γράμμα του ονόματος της μεταβλητής.

Ρητές δηλώσεις μεταβλητών

- Στη Fortran είναι δυνατή η χρήση μιας μεταβλητής χωρίς την προηγούμενη δήλωση του τύπου της.
- Στην περίπτωση αυτή, ο τύπος της μεταβλητής καθορίζεται από το αρχικό γράμμα του ονόματος της μεταβλητής.
- Αν αυτό είναι ένα από τα i, j, k, l, m, n τότε η μεταβλητή είναι ακέραιου τύπου, ενώ
- αν το αρχικό γράμμα είναι ένα από τα υπόλοιπα γράμματα του λατινικού αλφαβήτου η μεταβλητή είναι τύπου πραγματικού.

Ρητές δηλώσεις μεταβλητών

- Στη Fortran είναι δυνατή η χρήση μιας μεταβλητής χωρίς την προηγούμενη δήλωση του τύπου της.
- Στην περίπτωση αυτή, ο τύπος της μεταβλητής καθορίζεται από το αρχικό γράμμα του ονόματος της μεταβλητής.
- Αν αυτό είναι ένα από τα i, j, k, l, m, n τότε η μεταβλητή είναι ακέραιου τύπου, ενώ
- αν το αρχικό γράμμα είναι ένα από τα υπόλοιπα γράμματα του λατινικού αλφαβήτου η μεταβλητή είναι τύπου πραγματικού.
- Ο τύπος των μεταβλητών που χρησιμοποιούνται στα προγράμματα με αυτό τον τρόπο ονομάζονται *υπονοούμενος* (implicit).
- Μια χρήση μεταβλητών όπως η παραπάνω είναι ενδεχόμενο να οδηγήσει σε λάθη και πρέπει να αποφεύγεται.

Η δήλωση IMPLICIT NONE

H

δήλωση

IMPLICIT NONE

στην αρχή του προγράμματος 'απαγορεύει' τη χρήση μιας μεταβλητής πριν αυτή δηλωθεί και συνιστάται η χρήση της.

Επώνυμες σταθερές

Μέχρι τώρα έγινε λόγος για σταθερές τιμές δεδομένων στις οποίες αναφερόμαστε ρητά στην τιμή τους, π.χ. 123. 1.2E-4, κ.λπ.

Σε αρκετές περιπτώσεις θέλουμε να δώσουμε σε μια σταθερά ένα *συμβολικό όνομα*, όπως και στην περίπτωση των μεταβλητών.

Δήλωση σταθεράς με τη δεσμευμένη λέξη `PARAMETER`.

Η δήλωση αυτή γίνεται στην αρχή του προγράμματος, μαζί με τις δηλώσεις των μεταβλητών.

```
real:: x  
real, parameter:: e = 2.7182817
```

```
x = 12.5  
print*, x * e ! Εκτύπωση του γινομένου e * x
```


Αριθμητικές εκφράσεις

Ένα πρόγραμμα είναι δυνατόν να περιέχει *αριθμητικούς υπολογισμούς*. Οι αριθμητικοί υπολογισμοί ορίζονται μέσα σε ένα πρόγραμμα Fortran με τη μορφή *αριθμητικών εκφράσεων ή παραστάσεων*. Παραδείγματα τέτοιων εκφράσεων είναι τα ακόλουθα:

$$1 + 2$$

$$12 * 3$$

$$(4.05 - 123) * 2.05e5$$

Αριθμητικές εκφράσεις

Ένα πρόγραμμα είναι δυνατόν να περιέχει *αριθμητικούς υπολογισμούς*. Οι αριθμητικοί υπολογισμοί ορίζονται μέσα σε ένα πρόγραμμα Fortran με τη μορφή *αριθμητικών εκφράσεων ή παραστάσεων*. Παραδείγματα τέτοιων εκφράσεων είναι τα ακόλουθα:

$$1 + 2$$

$$12 * 3$$

$$(4.05 - 123) * 2.05e5$$

$$12.5 * 34 - 12.1 / 0.44$$

Αριθμητικές εκφράσεις

Ένα πρόγραμμα είναι δυνατόν να περιέχει *αριθμητικούς υπολογισμούς*. Οι αριθμητικοί υπολογισμοί ορίζονται μέσα σε ένα πρόγραμμα Fortran με τη μορφή *αριθμητικών εκφράσεων ή παραστάσεων*. Παραδείγματα τέτοιων εκφράσεων είναι τα ακόλουθα:

$1 + 2$

$12 * 3$

$(4.05 - 123) * 2.05e5$

$12.5 * 34 - 12.1 / 0.44$

`num + 2`

Όροι και αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές στη Fortran είναι οι ακόλουθοι:

Όροι και αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές στη Fortran είναι οι ακόλουθοι:

-
- + Πρόσθεση
 - Αφαίρεση ή μοναδιαία άρνηση
 - * Πολ/μός

Όροι και αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές στη Fortran είναι οι ακόλουθοι:

-
- + Πρόσθεση
 - Αφαίρεση ή μοναδιαία άρνηση
 - * Πολ/μός
 - / Διαίρεση

Όροι και αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές στη Fortran είναι οι ακόλουθοι:

+	Πρόσθεση
-	Αφαίρεση ή μοναδιαία άρνηση
*	Πολ/μός
/	Διαίρεση
**	Ύψωση σε δύναμη

Παραδείγματα του μοναδιαίου - είναι:

-12

-(34.4 + 23.05 * 4)

Προτεραιότητες και παρενθέσεις

Προτεραιότητα τελεστών: που ακολουθεί:

Σύμβολο	Όνομα	Προτεραιότητα
**	Υψωση σε δύναμη	Υψηλότερη
* /	Πολ/μός και διαίρεση	
-	Μοναδιαία άρνηση	
+ -	Πρόσθεση και αφαίρεση	Χαμηλότερη

Στην παράσταση

12.5 - 34 * 12.1 / 0.44 ** 2

Αριθμητικές εκφράσεις και συναρτήσεις

Οι αριθμητικές εκφράσεις είναι δυνατόν να περιέχουν *συναρτήσεις*.

Η Fortran παρέχει ένα σύνολο συναρτήσεων γενικής χρήσης οι οποίες ονομάζονται *εγγενείς (intrinsic) συναρτήσεις*.

Μια συνάρτηση αποτελείται από ένα συμβολικό όνομα ακολουθούμενο από ένα ή περισσότερα *ορίσματα* μέσα σε *παρενθέσεις*. Τα ορίσματα αυτά είναι αριθμητικές εκφράσεις δηλ. σταθερές, μεταβλητές ή πιο σύνθετες παραστάσεις.

Το παρακάτω πρόγραμμα καλεί τη συνάρτηση `sqrt` για να υπολογίζει την τετραγωνική ρίζα του αριθμού 2.

```
PROGRAMM root

PRINT*, " Η τετραγωνική ρίζα του 2 είναι: ", sqrt(2.0)
END PROGRAM root
```



Όνομα	Τιμή	Τύπος ορίσματος	Τύπος αποτελέσματος
ABS(X)	$ x $	R/I	R/I
SQRT(X)	\sqrt{x}	R	R
SIN(X)	$\sin x$	R	R
COS(X)	$\cos x$	R	R
TAN(X)	$\tan x$	R	R
EXP(X)	e^x	R	R
LOG(X)	$\log_e(x)$	R	R
LOG10(X)	$\log_{10}(x)$	R	R
ASIN(X)	$\sin^{-1}(x)$	R	R
ACOS(X)	$\cos^{-1}(x)$	R	R
ATAN(X)	$\tan^{-1}(x)$	R	R
CMPLX(x, y)	$x + iy$	I/R/C	C
INT(X)		R	I
NINT(X)		R	I
REAL(X)		I	R
MOD(A, B)		R/I	R/I
MAX(A, B)		R/I	R/I
MIN(A, B)		R/I	R/I

Εντολές ανάθεσης

Η εντολή

μεταβλητή = έκφραση

αντιγράφει την τιμή της έκφρασης έκφραση ως περιεχόμενο της μεταβλητής μεταβλητή. Αρχικά, υπολογίζεται η τιμή (και ο τύπος) της έκφρασης.

Στο παράδειγμα

```
result = 2.5 ** 2
```

η έκφραση $2.5 ** 2$ αποτιμάται, δηλ. υπολογίζεται η τιμή της και στη συνέχεια το αποτέλεσμα (6.25) καταχωρείται ως τιμή της μεταβλητής result.

Μετατροπή πραγματικού σε ακέραιο

```
real :: a,b
```

```
integer :: c
```

```
a = 1.4; b = 3.3
```

```
c = a + b
```

Η τιμή του c είναι 4.

Μετατροπή ακέραιου σε πραγματικό

```
real ::a  
integer ::c
```

```
c = 2  
a = c
```

Η τιμή του a είναι 2.0.

Μετατροπή πραγματικού σε μιγαδικό

```
real :: a = 1.0  
complex c
```

```
c = a
```

Η τιμή του c είναι $(1.0, 0.0)$.

oooooooooooooooo

PROGRAM VolumeSphere

```
PROGRAM VolumeSphere
```

```
END PROGRAM VolumeSphere
```



```
PROGRAM VolumeSphere
  ! Υποχρεωτική δήλωση μεταβλητών
  IMPLICIT NONE
  ! Δήλωση μεταβλητών
  REAL:: V,r
  ! Δήλωση σταθεράς
  REAL, PARAMETER :: pi = 3.14159
  ! Είσοδος δεδομένων
  PRINT*, ' Ακτίνα='
  READ*, r
  ! Υπολογισμός όγκου

  ! Εκτύπωση αποτελεσμάτων

END PROGRAM VolumeSphere
```

```
PROGRAM VolumeSphere
  ! Υποχρεωτική δήλωση μεταβλητών
  IMPLICIT NONE
  ! Δήλωση μεταβλητών
  REAL :: V,r
  ! Δήλωση σταθεράς
  REAL, PARAMETER :: pi = 3.14159
  ! Είσοδος δεδομένων
  PRINT*, ' Ακτίνα='
  READ*, r
  ! Υπολογισμός όγκου
  V = 4.0 / 3 * pi * r ** 3
  ! Εκτύπωση αποτελεσμάτων

END PROGRAM VolumeSphere
```

```
PROGRAM VolumeSphere
! Τποχρεωτική δήλωση μεταβλητών
IMPLICIT NONE
! Δήλωση μεταβλητών
REAL:: V,r
! Δήλωση σταθεράς
REAL, PARAMETER :: pi = 3.14159
! Είσοδος δεδομένων
PRINT*, ' Ακτίνα='
READ*, r
! Υπολογισμός όγκου
V = 4.0 / 3 * pi * r ** 3
! Εκτύπωση αποτελεσμάτων
PRINT*, ' Όγκος σφαίρας :',V

END PROGRAM VolumeSphere
```


