ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

6° Εξάμηνο

- Ασκηση Πράξης 2 -

Αναδρομικός Προγραμματισμός

Δημοσθένης Σταμάτης
http:/www.iee.ihu.gr/~demos
Τμήμα Μηχανικών Πληροφορικής & Ηλεκτρονικών Συστημάτων

Η ΑΝΑΔΡΟΜΗ ως Μεθοδολογία Προγραμματισμού

- Για να επιλύσουμε ένα πρόβλημα με **αναδρομικό** τρόπο **πρέπει να δεχτούμε ότι το πρόβλημα αυτό είναι λυμένο ήδη** για οποιοδήποτε σύνολο δεδομένων εισόδου μικρότερο από το αρχικό.

- Παραδείγματος χάριν αν στο αρχικό πρόβλημα έχουμε να αντιμετωπίσουμε ένα σύνολο από **Ν** δεδομένα, θεωρούμε ότι το πρόβλημα μας είναι λυμένο για **Ν-1** δεδομένα ή **Ν/2** δεδομένα

Πρέπει να κάνουμε 2 πράγματα:

- (α) Να εκφράσουμε το αρχικό μας πρόβλημα σε σχέση με (το λυμένο) μικρότερο της ίδιας μορφής.
- (β) Να αναζητήσουμε τις οριακές περιπτώσεις (συνθήκες) του προβλήματος, οι οποίες αποτελούν βασική προϋπόθεση για να λειτουργήσει η αναδρομή

Αναδρομικός ορισμός του N παραγοντικό

```
factorial(0,1).  % οριακή περίπτωση
factorial(N, NFactorial):-  % αναδρομικός κανόνας
        N>0,
        N1 is N-1,
        factorial(N1,N1Factorial),  % αναδρομική κλήση
        NFactorial is N*N1Factorial.
```

```
?- factorial(5,F).
   F = 120

?- factorial(10,F).
   F = 3628800.
```


Φυσικοί αριθμοί (αναδρομικός ορισμός)

```
natural(1).  % οριακή περίπτωση
natural(N):-  % αναδρομικός κανόνας
        N >1,
        N1 is N - 1,
        natural(N1).  % αναδρομική κλήση
```

```
?- natural(100).
true .

?- natural(43.5).
false.
```

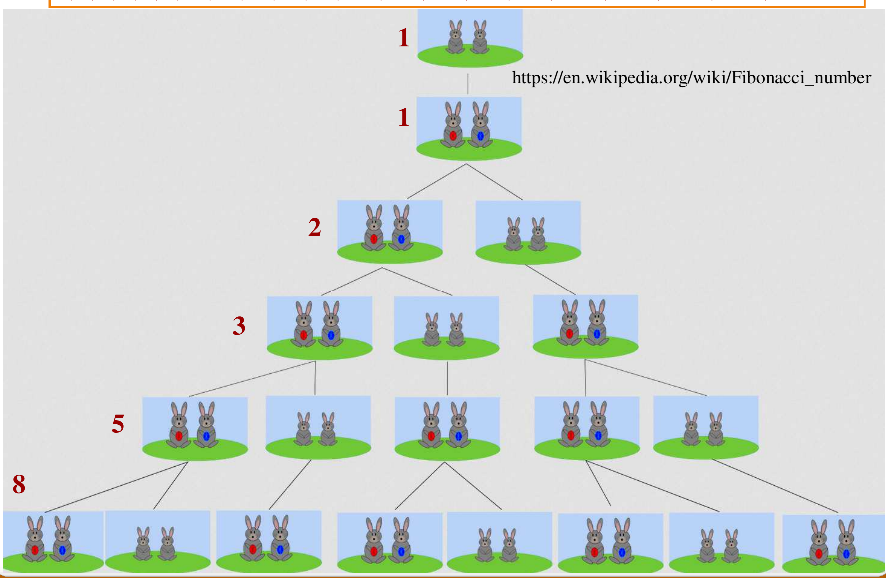
Ύψωση σε Δύναμη (αναδρομικός ορισμός)

```
power(X,0,1).        % οριακή περίπτωση 1
power(X,1,X).        % οριακή περίπτωση 2
power(X,N,P) :-      % αναδρομικός κανόνας
        N > 1,
        N1 is N-1,
        power(X,N1,P1),   % αναδρομική κλήση
        P is X*P1.
```


ΤΜΗΜΑ ΜΗΧ.ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΔΙΠΑΕ

Η Αναδρομική ακολουθία Fibonacci

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946



ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ 6ο ΕΞΑΜΗΝΟ

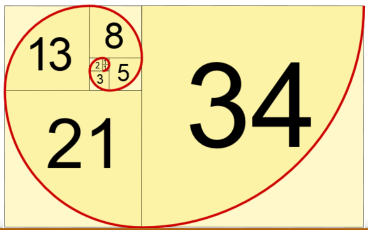
Η Αναδρομική ακολουθία Fibonacci

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946

- fibonacci(1,1). *% οριακή περίπτωση 1*
- fibonacci(2,1). *% οριακή περίπτωση 2*
- fibonacci(N,Result) :- *% αναδρομικός κανόνας*

```
        N > 2,
        N1 is N-1,
        N2 is N-2,
        fibonacci(N1,Y1),  % 1η αναδρομική κλήση
        fibonacci(N2,Y2),  % 2η αναδρομική κλήση
        Result is Y1+Y2.
```

?- fibonacci(9,A).
A = 34



Ο υπολογισμός του Μέγιστου Κοινού Διαιρέτη αναδρομικά

- Για να βρούμε το μέγιστο κοινό διαιρέτη δύο αριθμών Μ και Ν, (Μ>Ν) αρκεί να βρούμε:
- το μέγιστο κοινό διαιρέτη του Ν με το υπόλοιπο της διαίρεσης Μ:Ν

```
ΜΚΔ(N,M) = ΜΚΔ(M,N), αν N<M
ΜΚΔ(N,M) = ΜΚΔ(M,mod(N,M)), αν N≥M
ΜΚΔ(N,0) = N
```

Α/Α	Ν	Μ
1	8	22
2	22	8
3	8	6
4	6	2
5	**2**	0

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ 6ο ΕΞΑΜΗΝΟ

Ο υπολογισμός του Μέγιστου Κοινού Διαιρέτη αναδρομικά

```
mkd(N,0,N).  % οριακή περίπτωση

mkd(N,M,X) :-
        N>=M,
        M1 is mod(N,M),
        mkd(M,M1,X). % αναδρομική κλήση

mkd(N,M,X) :-
        N<M,
        mkd(M,N,X).  % εναλλαγή παραμέτρων

?- mkd(1024,24,R).
   R = 8
```


ΤΜΗΜΑ ΜΗΧ.ΠΛΗΡΟΦΟΡΙΚΗΣ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΔΙΠΑΕ

Τυχαία αναδρομή vs αναδρομή ουράς

ΑΝΑΔΡΟΜΗ	ΑΝΑΔΡΟΜΗ ΟΥΡΑΣ
factorial(0,1).	natural(1).
factorial(N, NFactorial):-	natural(N):-
N>0,	N >1,
N1 is N-1,	N1 is N - 1,
factorial(N1,N1Factorial),	natural(N1).
NFactorial is N*N1Factorial.	

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ 6ο ΕΞΑΜΗΝΟ

Τυχαία αναδρομή vs αναδρομή ουράς

ΑΝΑΔΡΟΜΗ	ΑΝΑΔΡΟΜΗ ΟΥΡΑΣ
fibonacci(1,1). fibonacci(2,1). fibonacci(N,Result) :- N > 2, N1 is N-1, N2 is N-2, fibonacci(N1,Y1), fibonacci(N2,Y2), Result is Y1+Y2	fibonacci(N, Result) :- i_fibonacci(N, 1, 1, Result). i_fibonacci(1, Result, _, Result). i_fibonacci(N, FN1, FN2, F) :- N>1, N1 is N - 1, Sofar is FN1 + FN2, i_fibonacci (N1, FN2, Sofar, F).

(*) Η 2η και 3η παράμετρος του κατηγορήματος του i_fibonacci ονομάζονται παράμετροι συσσώρευσης. Περισσότερα για την τεχνική αυτή αργότερα . . .