

# Uncapacitated Exam Timetabling

Δήμητσας Άγγελος



# Το πρόβλημα

- i. Να μοιράσουμε τις εξετάσεις σε εξεταστικές περιόδους (Περιορισμός στον αριθμό των περιόδων).
- ii. ΚΑΝΕΝΑΣ εξεταζόμενος να μην δίνει δύο εξετάσεις την ίδια εξεταστική περίοδο.  
(Hard Constraints)
- iii. Όσο γίνεται οι εξεταζόμενοι να μην έχουν εξετάσεις σε κοντινές εξεταστικές περιόδους.  
(Soft Constraints)

# Τι θα δούμε

- i. Δημιουργία αρχικής λύσης
- ii. Βελτίωση λύσης
- iii. Δημιουργία νέων λύσεων

Βρίσκοντας  
μια αρχική  
λύση

## **BRUTE FORCE**

Θα δοκιμάσουμε όλα τα πιθανά  
ενδεχόμενα μέχρι κάποιος να δώσει λύση.

# Μιλώντας με νούμερα

## BRUTE FORCE

Αριθμός εξετάσεων: 5

Αριθμός περιόδων: 3

$(ΕΞΕΤΑΣΗ_1) = \{1,2,3\}$

$(ΕΞΕΤΑΣΗ_2) = \{1,2,3\}$

$(ΕΞΕΤΑΣΗ_3) = \{1,2,3\}$

$(ΕΞΕΤΑΣΗ_4) = \{1,2,3\}$

$(ΕΞΕΤΑΣΗ_5) = \{1,2,3\}$

Πιθανές καταστάσεις:  $3^5 = 243$

Μιλώντας με  
νούμερα

## BRUTE FORCE

Το μικρότερο πρόβλημα : sta-f-83.stu

Αριθμός εξετάσεων: 139

Αριθμός περιόδων: 13

$(EΞΕΤΑΣΗ_1) = \{1, 2..13\}$

$(EΞΕΤΑΣΗ_2) = \{1, 2..13\}$

...

$(EΞΕΤΑΣΗ_{139}) = \{1, 2..13\}$

Πιθανές καταστάσεις :  $= 10^{184}$   
7.2309E+27

# Μιλώντας με νούμερα

## BRUTE FORCE

Το μεγαλύτερο πρόβλημα : `pur-s-93.stu`

Αριθμός εξετάσεων: 2419

Αριθμός περιόδων: 42

$(EΞΕΤΑΣΗ_1) = \{1, 2..42\}$

$(EΞΕΤΑΣΗ_2) = \{1, 2..42\}$

...

$(EΞΕΤΑΣΗ_{2419}) = \{1, 2..42\}$

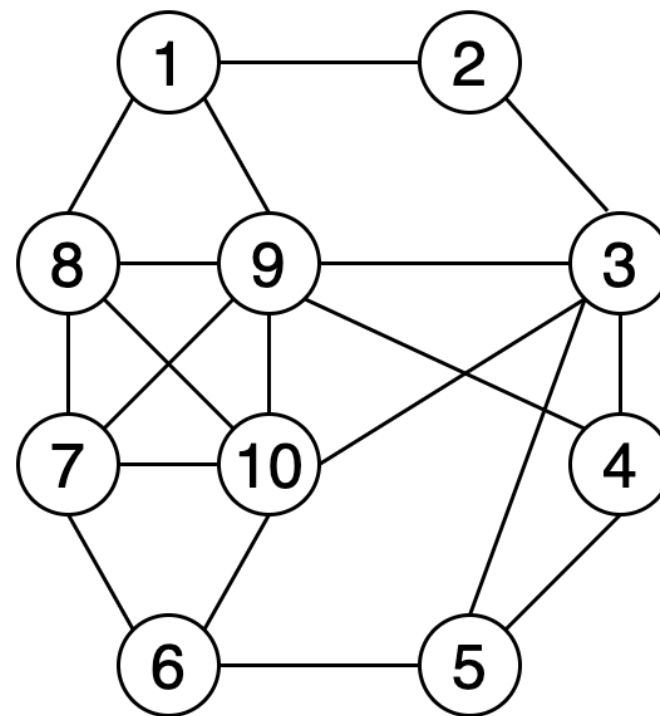
Πιθανές καταστάσεις :  $42^{2419} = 4.36549477264800840315e+3926$

Βρίσκοντας  
μια αρχική  
λύση

## Greedy Graph Coloring

Θα μετατρέψουμε το πρόβλημα σε  
χρωματισμό γραφήματος.

Κάθε χρώμα είναι μία εξεταστική περίοδος!





# Βρίσκοντας μια αρχική λύση

**Γιατί το Greedy Graph Coloring δεν  
αρκεί για εμάς.**

- Ο αλγόριθμος του Greedy Graph Coloring δεν εγγυάται ότι θα χρησιμοποιήσουμε τα λιγότερα δυνατά χρώματα !
  - Δεν ξεχνώ πως κάθε χρώμα είναι μία εξεταστική περίοδος!
- Μπορεί τα χρώματα που χρησιμοποιήσαμε να είναι περισσότερα από τα χρώματα που έχουμε στη διάθεση μας.
  - Δηλαδή έχουμε χρησιμοποιήσει περισσότερες εξεταστικές περιόδους από αυτές που έχουμε στη διάθεση μας.

# Βρίσκοντας μια αρχική λύση

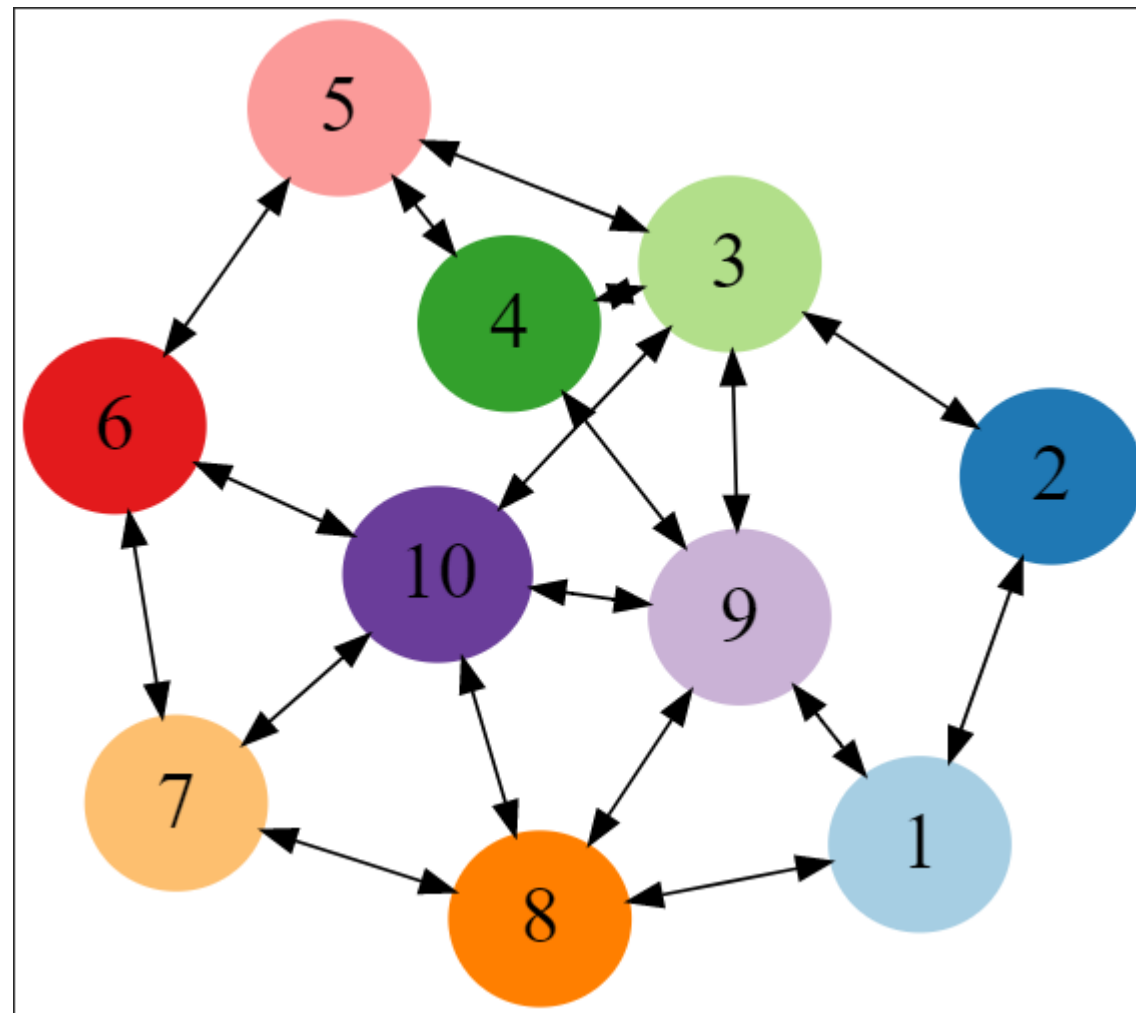
Πώς θα χρωματίσουμε το γράφημα με  
ίσα ή λιγότερα χρώματα από τα  
διαθέσιμα.

- Η σειρά των κόμβων με την οποία κάνουμε τον χρωματισμό αλλάζει τον τελικό αριθμό των χρωμάτων.
- Αν χρησιμοποιήσουμε τυχαίες σειρές κόμβων. Οι πιθανές μεταθέσεις (permutations) είναι : Αριθμός Εξετάσεων!
  - Για τις λιγότερες: hec-s-92.stu  
 $81! = 5.79712602074736798588e+120$
  - Για τις περισσότερες: pur-s-93.stu  
 $2419! = 3.52969088157697013226e+7136$
  - Χρειαζόμαστε Αλγόριθμους Degree Ordering

Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering

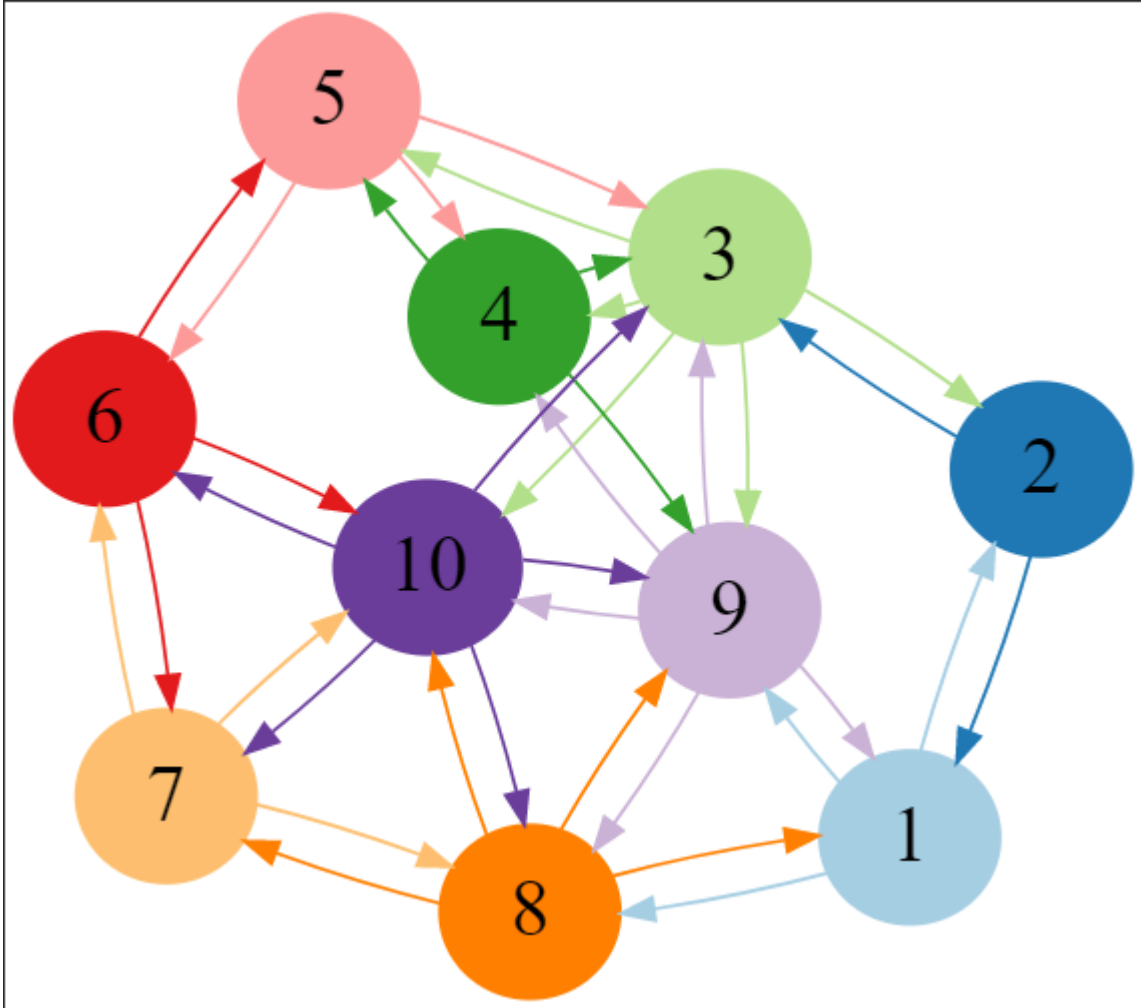
1: Largest First



# Βρίσκοντας μια αρχική λύση

# Αλγόριθμοι Degree Ordering

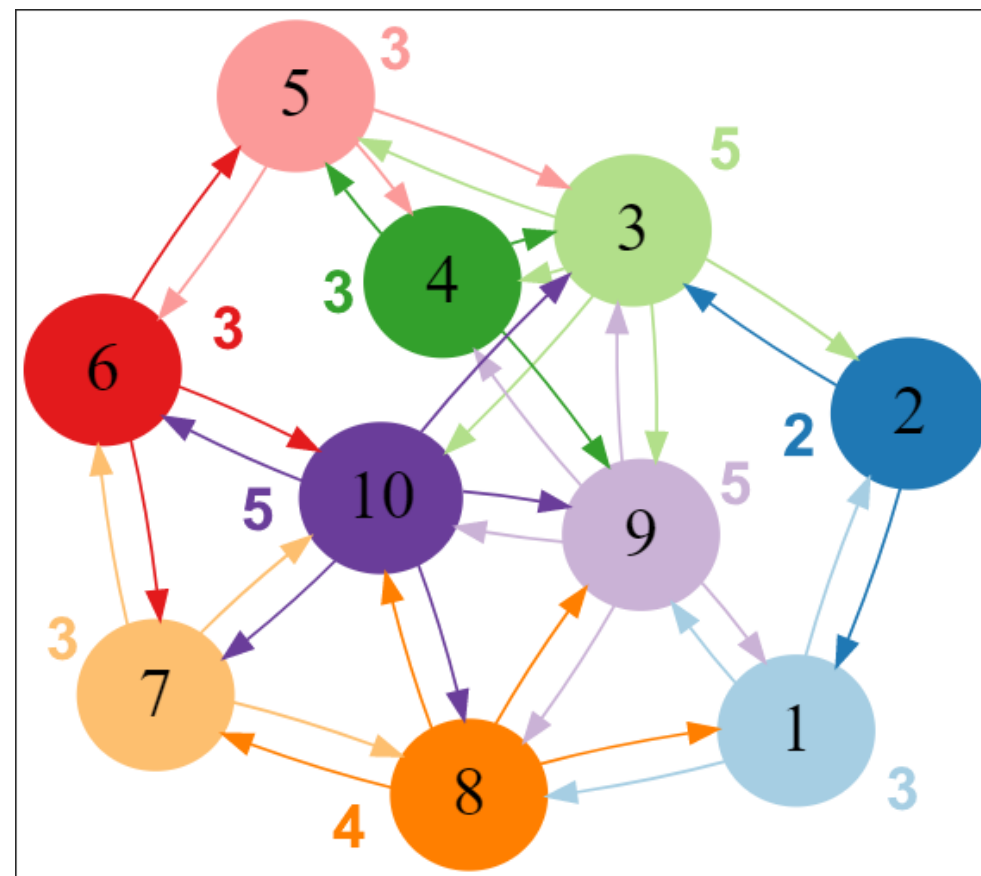
# 1: Largest First



Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering

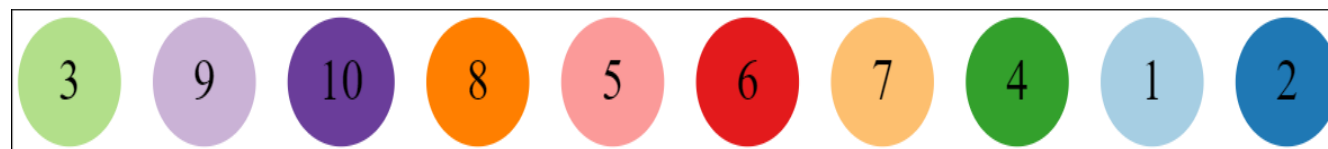
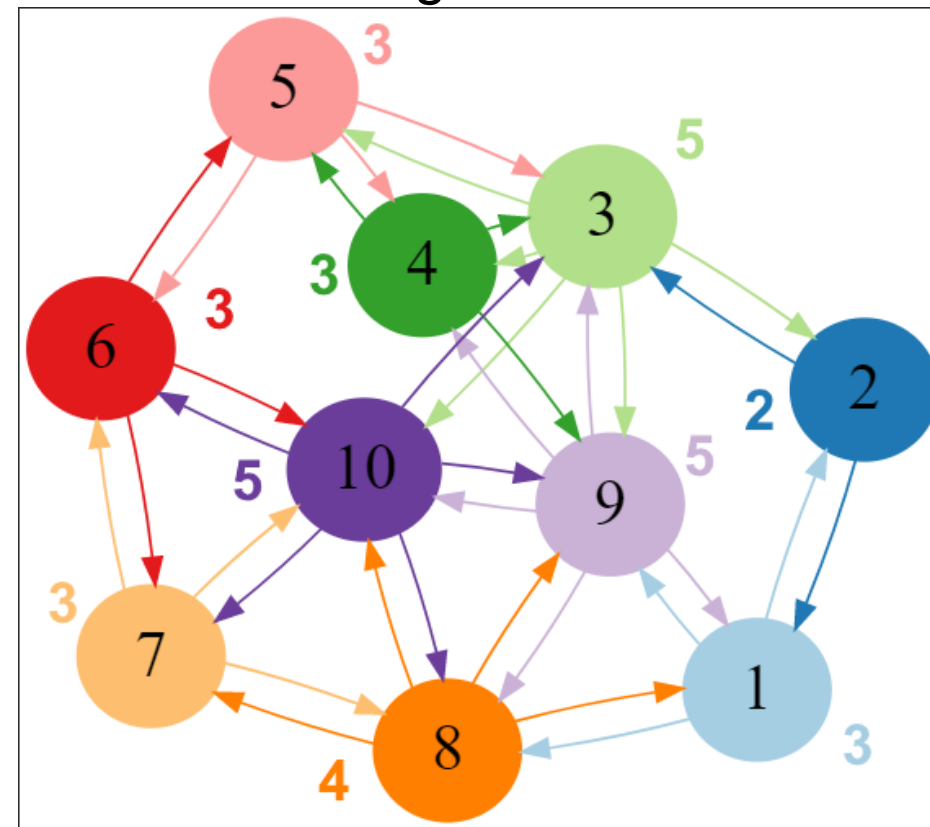
1: Largest First



Brooks' theorem: Μπορώ σίγουρα να χρωματίσω το γράφημα με 5 χρώματα! Εκτός από κυκλικά γραφήματα περιττού μήκους και complete graph που θα χρειαστώ 6!

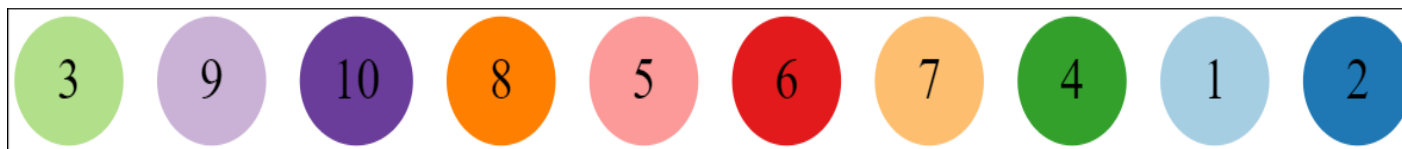
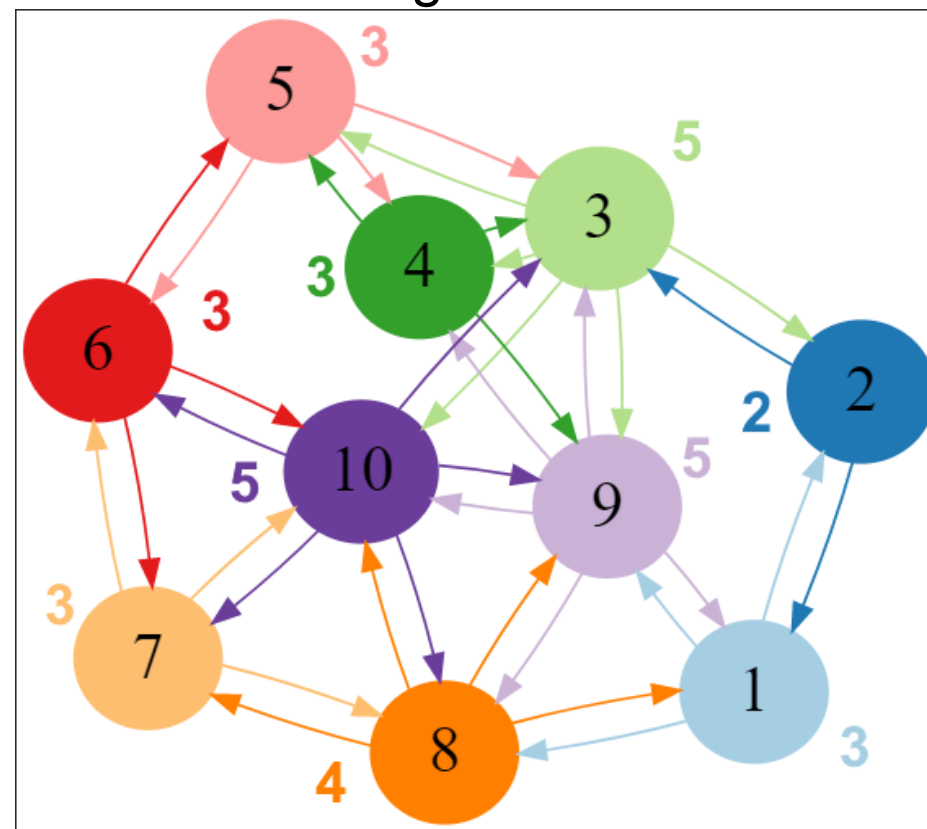
Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering  
1: Largest First



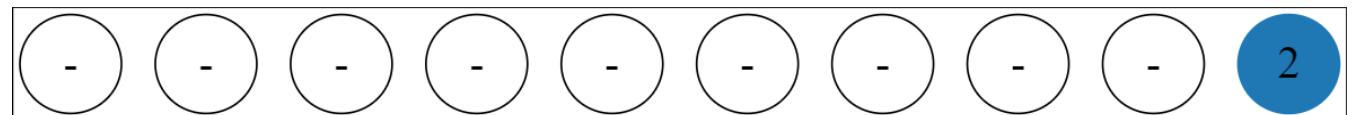
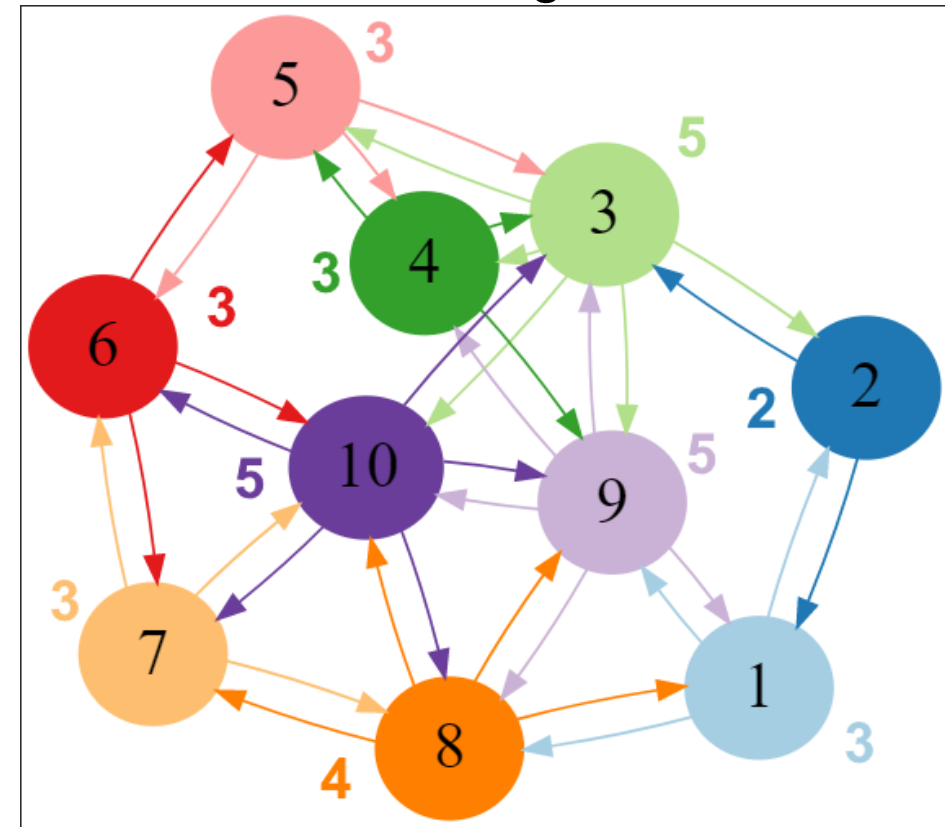
Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering  
1: Largest First



Βρίσκοντας  
μια αρχική  
λύση

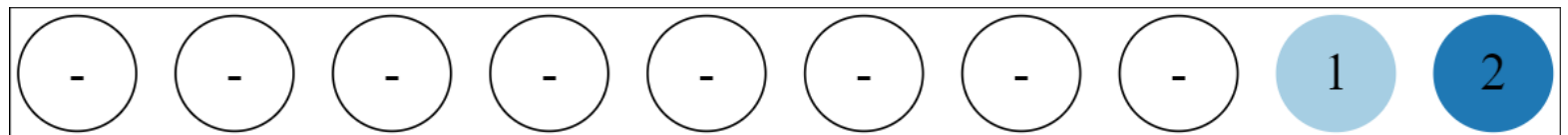
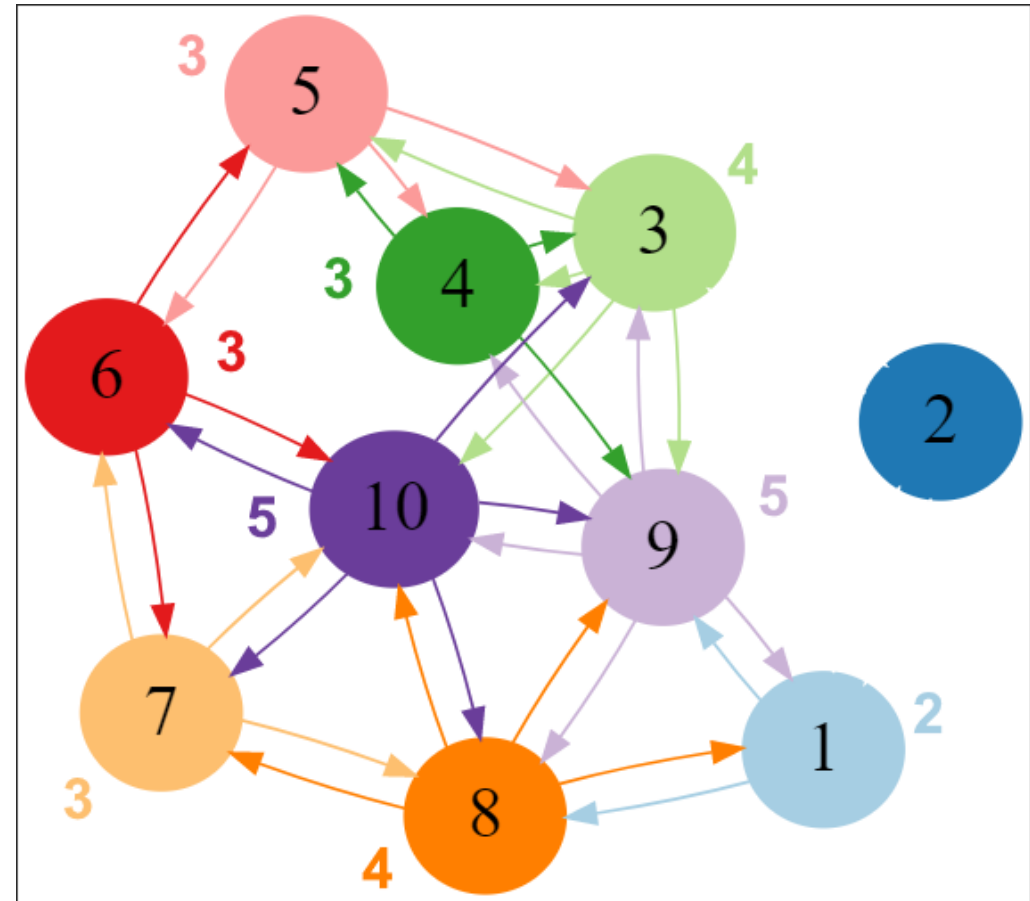
Αλγόριθμοι Degree Ordering  
2:Smallest-Degree-Last





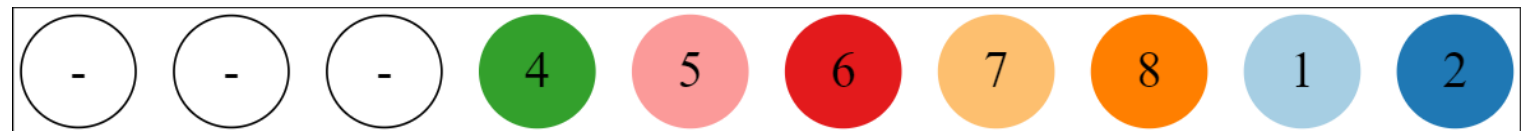
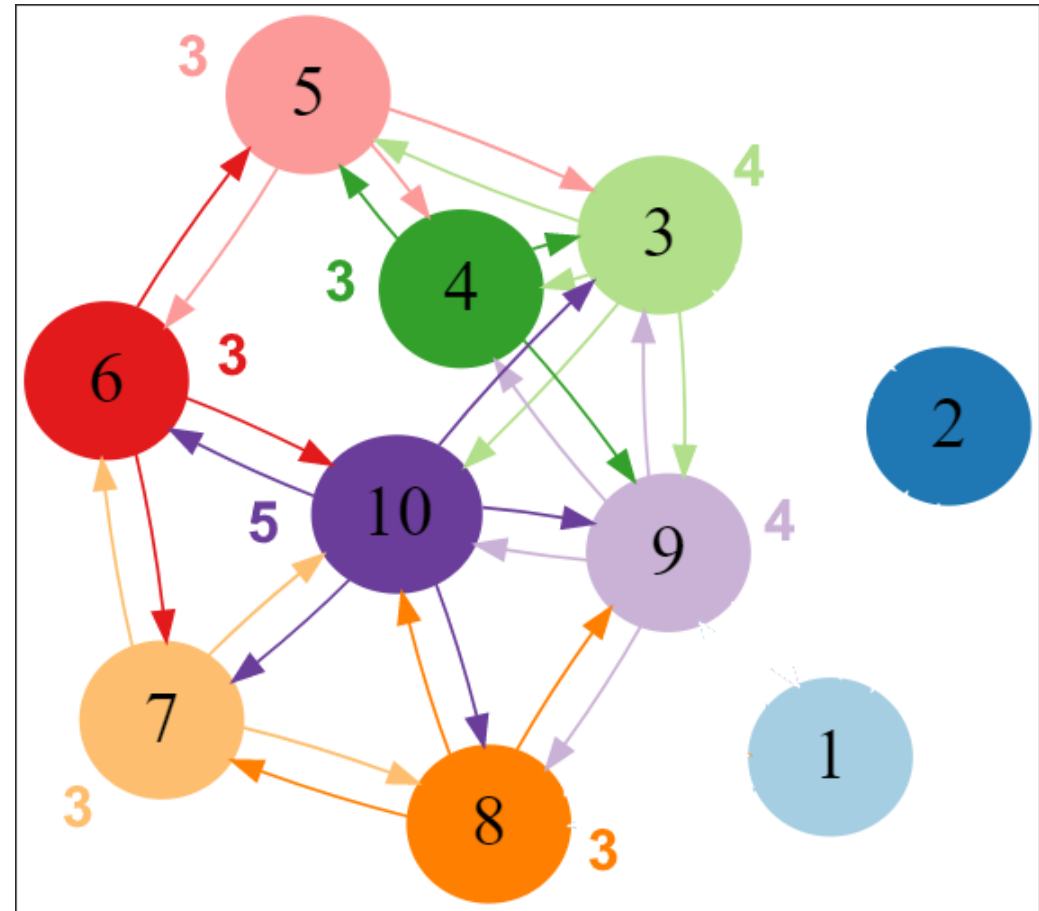
Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering  
2:Smallest-Degree-Last



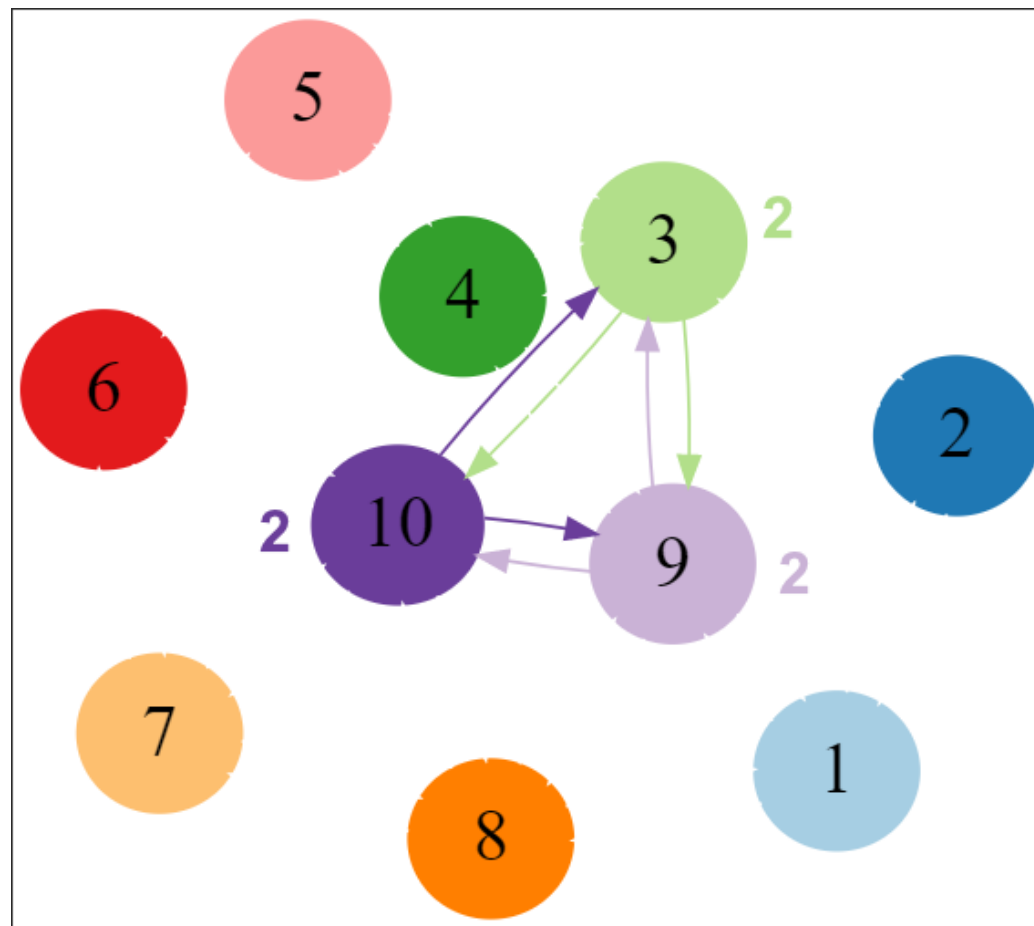
Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering  
2:Smallest-Degree-Last

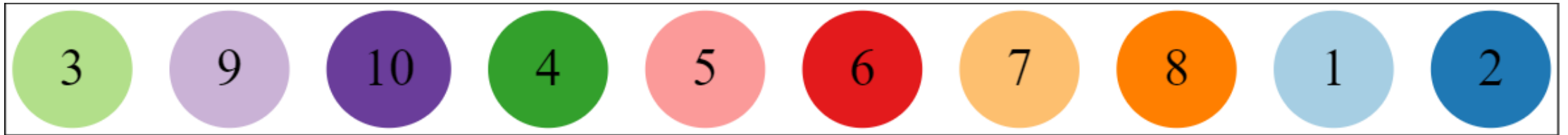


Βρίσκοντας  
μια αρχική  
λύση

Αλγόριθμοι Degree Ordering  
2:Smallest-Degree-Last



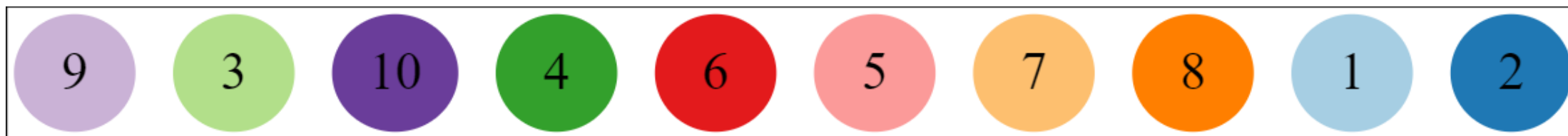
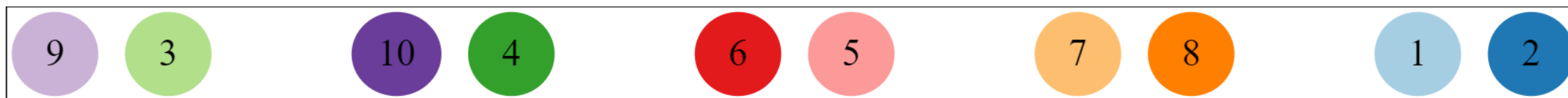
# Βρίσκοντας μια αρχική λύση



- Η αλληλουχία που βρήκαμε με την μέθοδο του Smallest-Degree-Last ίσως να μην χρησιμοποιεί λιγότερες ή ίσες εξεταστικές περιόδους από τις διαθέσιμες.
- Για να σπάσουμε τον ντετερμινισμό θα την ανακατέψουμε!

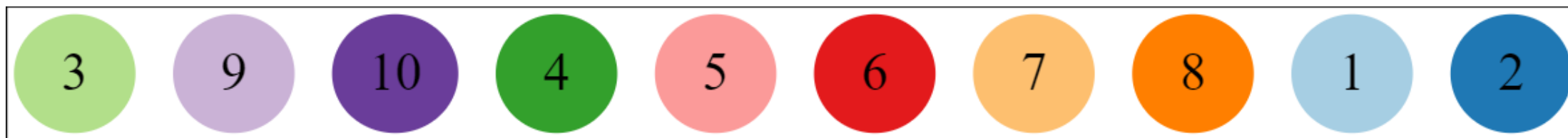
# Βρίσκοντας μια αρχική λύση

Ανακάτεμα ανά δύο:



# Βρίσκοντας μια αρχική λύση

Ανακάτεμα ανά τρία:



# Benchmarks Benchmarks Benchmarks Benchmarks

## CPU

Intel Core i7 3632QM

4 Cores, 8 Threads

## RAM

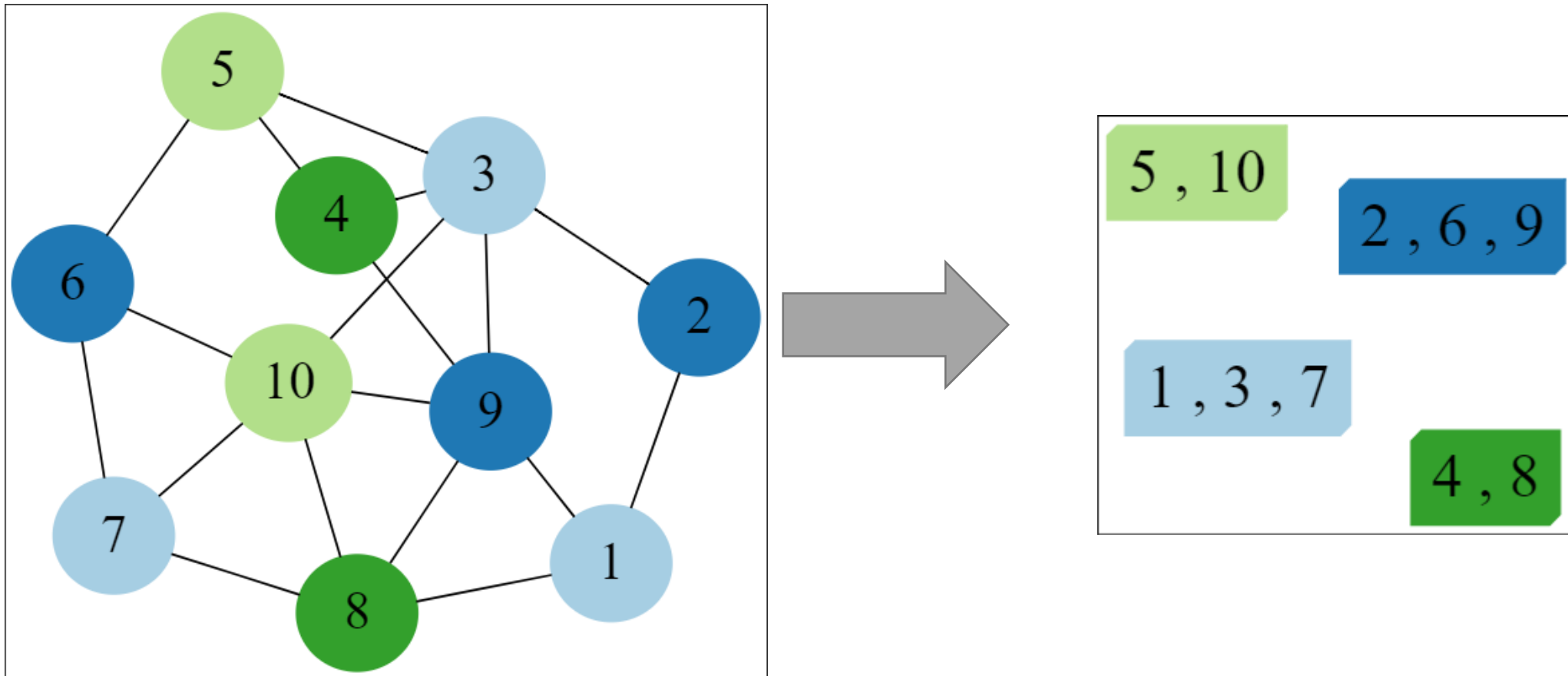
8 GB DDR3 1600 MHZ

Δείγμα: 1000

Αρχείο Δεδομένων	Εξετάσεις	Περίοδοι	Πυκνότητα	Μέσος Χρόνος
car-f-92.stu	543	32	0.14	339 ms
car-s-91.stu	682	35	0.13	566 ms
ear-f-83.stu	190	24	0.27	43 ms
hec-s-92.stu	81	18	0.42	23 s
kfu-s-93.stu	461	20	0.06	185 ms
lse-f-91.stu	381	18	0.06	100 ms
pur-s-93.stu	2419	42	0.03	11 s
rye-s-93.stu	486	23	0.07	248 ms
sta-f-83.stu	139	13	0.14	53 ms
tre-s-92.stu	261	23	0.18	102 ms
uta-s-92.stu	622	35	0.13	614 ms
ute-s-92.stu	184	10	0.08	44 ms
yor-f-83.stu	181	21	0.29	63 ms

# Βελτιώνοντας τις λύσεις μου.

Έχω την έγκυρη λύση μου την ομαδοποιώ σε ανεξάρτητα σετ.





## Βελτιώνοντας τις λύσεις μου.

1 , 3 , 7	2 , 6 , 9	4 , 8	5 , 10
-----------	-----------	-------	--------

Από τον πίνακα συγκρούσεων βρίσκω πόσοι εξεταζόμενοι από την μία περίοδο έχουν εξέταση και στις άλλες περιόδους.

Πχ για τις περιόδους {4,8} και {5,10} θα είναι το άθροισμα των κοινών εξεταζόμενων: 4 με 5, 4 με 10, 8 με 5 και 8 με 10 κοκ.

Δημιουργώ έναν πίνακα (θα επιταχύνει την διαδικασία αν τον έχω αποθηκεύσει στην RAM και δεν τον δημιουργώ κάθε φορά).

# Βελτιώνοντας τις λύσεις μου.

1 , 3 , 7	2 , 6 , 9	4 , 8	5 , 10
-----------	-----------	-------	--------

Έστω ότι έχω πλέον αυτόν τον πίνακα

COMMON	{1,3,7}	{2,6,9}	{4,8}	{5,10}
{1,3,7}	-	1	5	20
{2,6,9}	-	-	2	5
{4,8}	-	-	-	10
{5,10}	-	-	-	-

# Βελτιώνοντας τις λύσεις μου.

1 , 3 , 7	2 , 6 , 9	4 , 8	5 , 10
-----------	-----------	-------	--------

COMMON	{1,3,7}	{2,6,9}	{4,8}	{5,10}
{1,3,7}	-	1	5	20
{2,6,9}	-	-	2	5
{4,8}	-	-	-	10
{5,10}	-	-	-	-

Η βαθμολογία της λύσης μου θα είναι

$$1 * 16 + (2 * 16 + 5 * 8) + (10 * 16 + 5 * 8 + 20 * 4) = 368$$

Παρατηρώ ότι αν αλλάξω την σειρά των ανεξάρτητων σετ θα έχω διαφορετική βαθμολογία.

# Βελτιώνοντας τις λύσεις μου.

1 , 3 , 7    2 , 6 , 9    4 , 8    5 , 10

Η βαθμολογία της λύσης μου θα είναι

$$1 * 16 + (2 * 16 + 5 * 8) + (10 * 16 + 5 * 8 + 20 * 4) = 368$$

COMMON	{1,3,7}	{2,6,9}	{4,8}	{5,10}
{1,3,7}	-	1	5	20
{2,6,9}	-	-	2	5
{4,8}	-	-	-	10
{5,10}	-	-	-	-

5 , 10    4 , 8    2 , 6 , 9    1 , 3 , 7

Η βαθμολογία της λύσης μου θα είναι

$$10 * 16 + (2 * 16 + 5 * 8) + (1 * 16 + 5 * 8 + 20 * 4) = 368$$

**Παρατηρώ ότι δύο παλίνδρομες λύσεις θα έχουν πάντα το ίδιο αποτέλεσμα.**

# Μιλώντας με νούμερα

## Παλίνδρομα

1,2,3,4 **4,3,2,1**

1,2,4,3 **3,4,2,1**

1,3,2,4 **4,2,3,1**

1,3,4,2 **2,4,3,1**

1,4,2,3 **3,2,4,1**

1,4,3,2 **2,3,4,1**

2,1,3,4 **4,3,1,2**

2,1,4,3 **3,4,1,2**

2,3,1,4 **4,1,3,2**

2,4,1,3 **3,1,4,2**

3,2,1,4 **4,1,2,3**

3,1,2,4 **4,2,1,3**

Μιλώντας με  
νούμερα

Παλίνδρομα

1,2,3 **3,2,1**

2,1,3 **3,1,2**

1,3,2 **2,3,1**

# Βελτιώνοντας τις λύσεις μου.

- Έστω η σειρά:



- Έστω αλλάζω την 8 με την 22



- Θα επηρεαστούν το πολύ σε απόσταση  $\{-5, +5\}$



- Επειδή κοιτάω μόνο προς τα πίσω στην βαθμολογία μου



# Βελτιώνοντας τις λύσεις μου.

- Έστω η σειρά:



- Έστω αλλάζω την 12 με την 14



- Θα επηρεαστούν το πολύ σε απόσταση  $\{-5, +5\}$



- Επειδή κοιτάω μόνο προς τα πίσω στην βαθμολογία μου





# Μιλώντας με νούμερα

## PERMUTATIONS

- Αν χρησιμοποιήσουμε τυχαίες σειρές περιόδων. Οι πιθανές μεταθέσεις (permutations) είναι :  
ΑριθμόςΠεριοδών!
- Για τις λιγότερες: ute-s-92.stu  
 $10! = 3628800$
- Για τις περισσότερες: pur-s-93.stu  
 $42! = 1.40500611775287989854e51$
- Χρειαζόμαστε Αλγόριθμους Approximation

# Approximation

## Προσεγγιστικοί αλγόριθμοι

- Δεν θα βρουν απαραίτητα την καλύτερη λύση όπως θα έκανε το brute force ή το branch and bound
- Θα βρουν όμως καλύτερες λύσεις γρήγορα.
- Χρησιμοποιούν διάφορες παραμέτρους οπότε η αποτελεσματικότητά τους αλλάζει αναλόγως το είδος του.

Δημιουργία  
νέων λύσεων  
από  
υφιστάμενες.

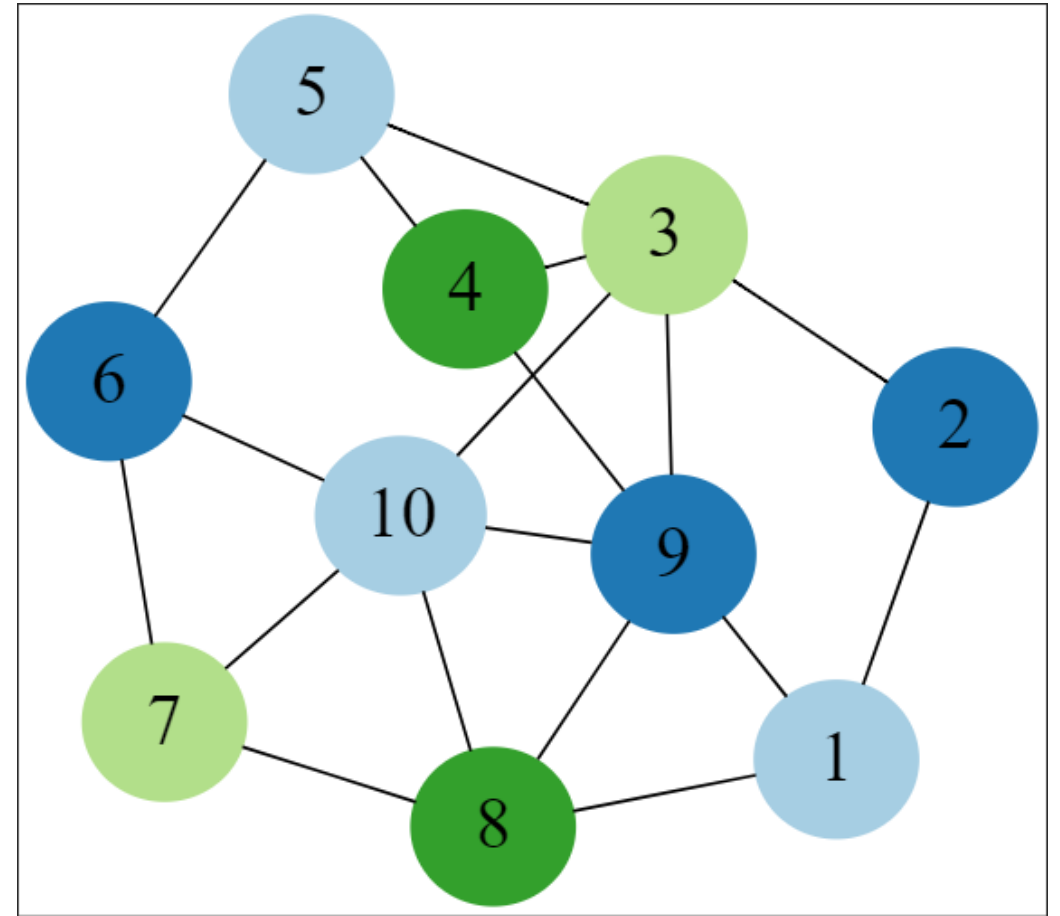
Η καλύτερη λύση ή μία καλύτερη λύση  
από αυτήν που έχω δεν είναι  
απαραίτητο να είναι στον τρόπο που  
έχω χωρίσει τις εξετάσεις σε  
περιόδους.

Χρειάζομαι έναν τρόπο να δημιουργώ  
γρήγορα νέα ανεξάρτητα σετ  
εξετάσεων!

Δημιουργία  
νέων λύσεων  
από  
υφιστάμενες.

Αλυσίδες Kempe

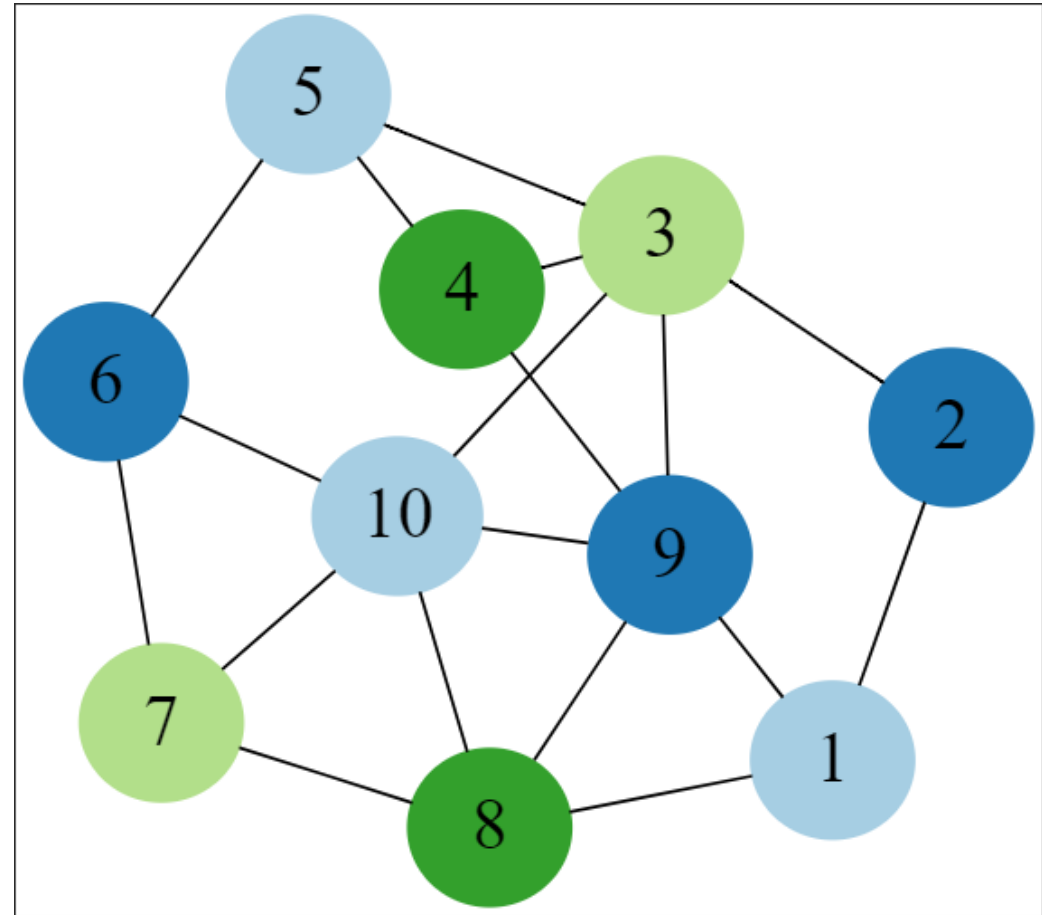
Έστω ότι έχω φτάσει στον παρακάτω  
χρωματισμό.



# Δημιουργία νέων λύσεων από υφιστάμενες.

## Αλυσίδες Kempe

Θα προσπαθήσω να αλλάξω χρώματα σε κάποιους κόμβους. Δηλαδή να αλλάξω εξετάσεις ανάμεσα σε εξεταστικές περιόδους.



# Δημιουργία νέων λύσεων από υφιστάμενες.

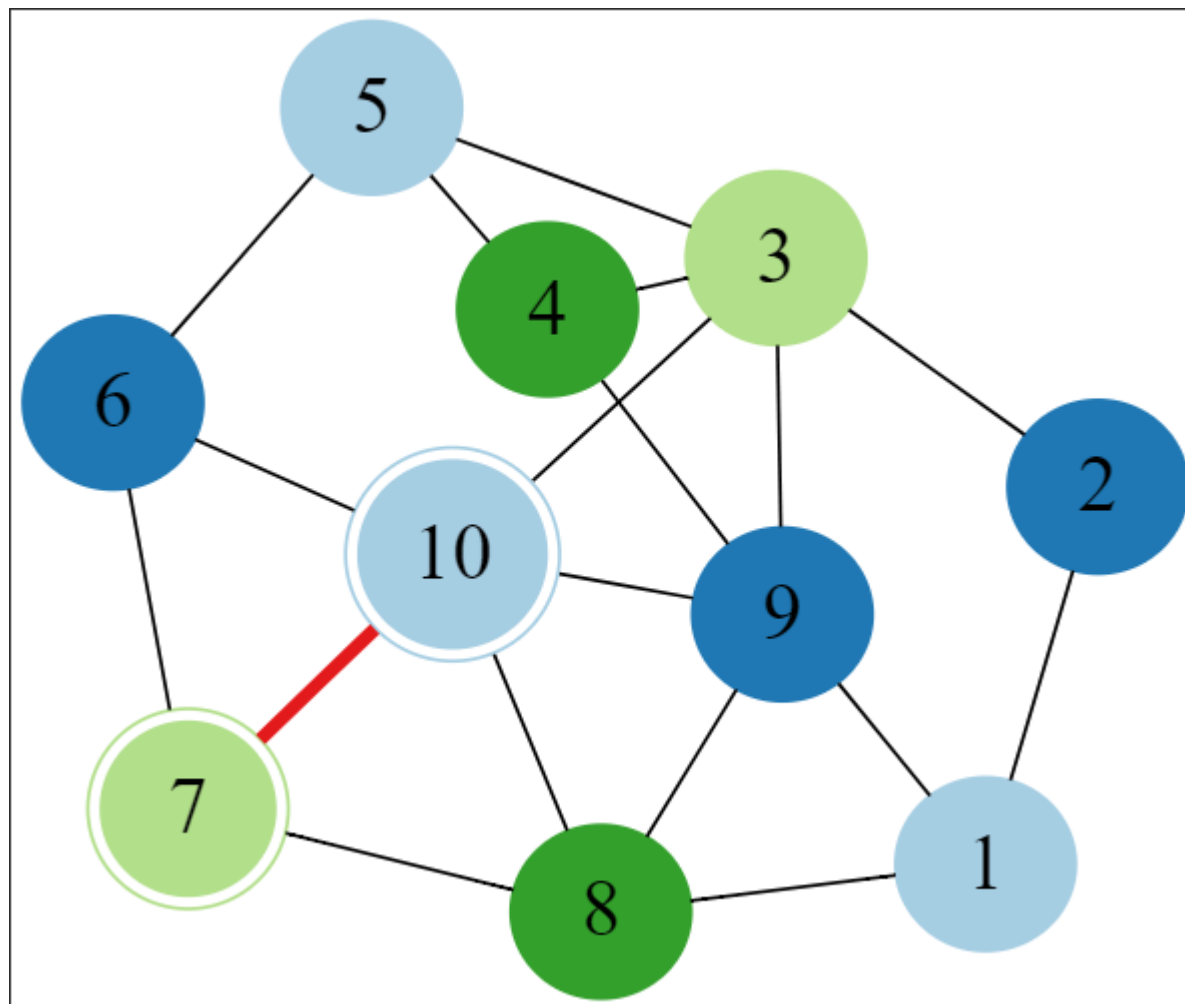
Αλυσίδες Kempe

Αρχικά επιλέγω δύο περιόδους στην τύχη.

Έστω την γαλάζια και την ανοιχτή πράσινη.

Πάλι στην τύχη επιλέγω δύο εξετάσεις από αυτές.

Έστω το 7 και το 10.

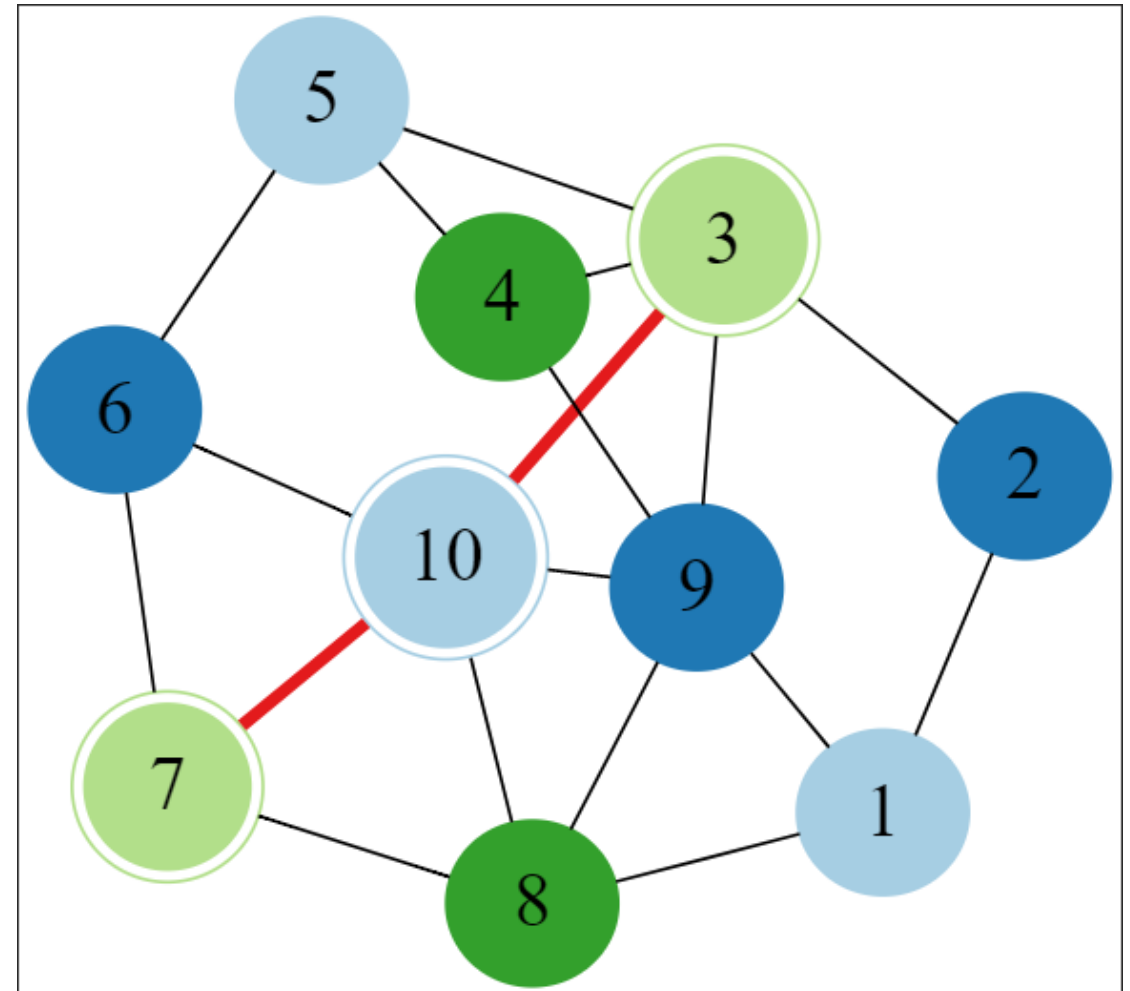


# Δημιουργία νέων λύσεων από υφιστάμενες.

## Αλυσίδες Kempe

Επιλέγω τις εξετάσεις που ενώνονται με το 7 και έχουν χρώμα γαλάζιο (καμία).

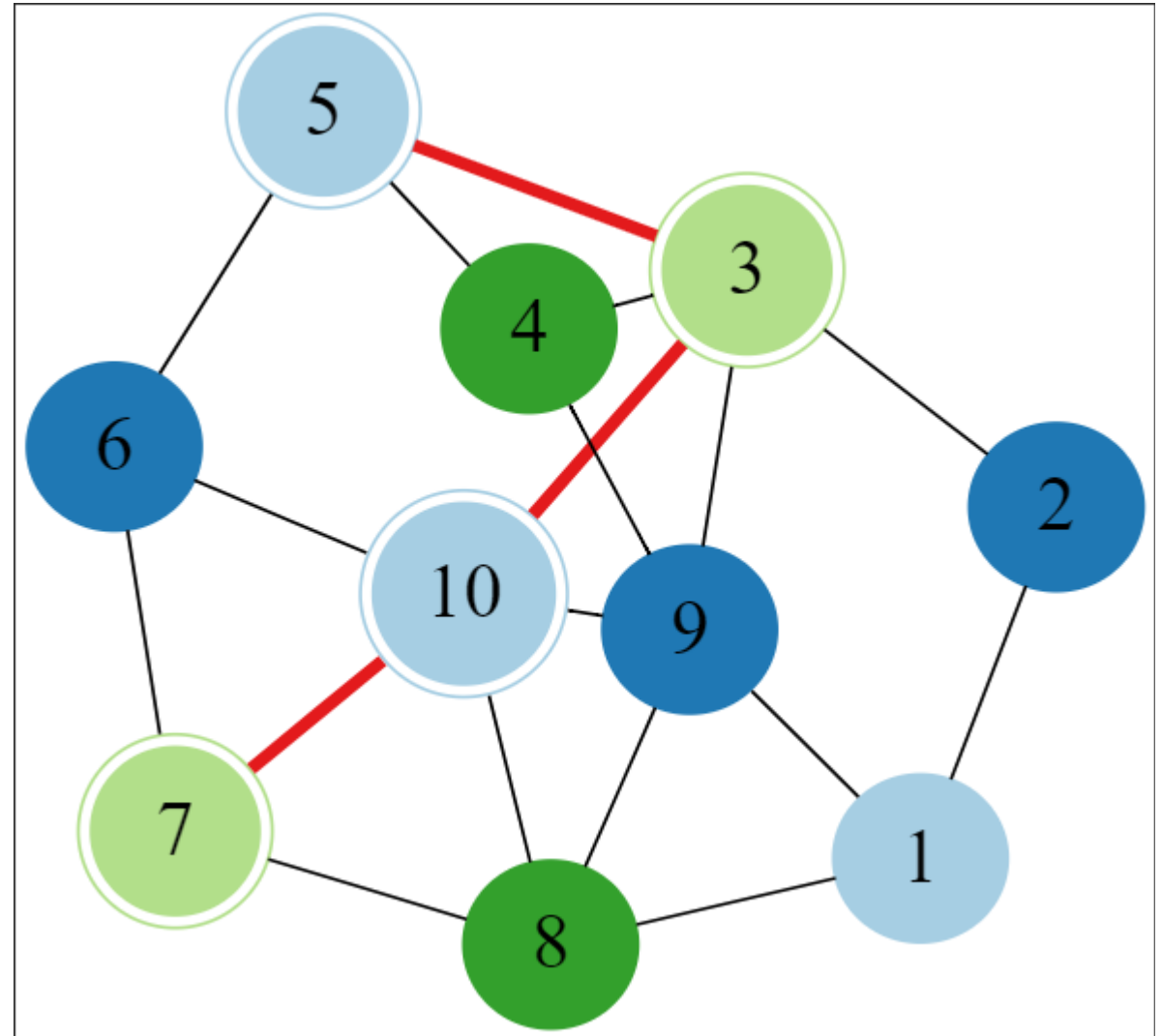
Επιλέγω τις εξετάσεις που ενώνονται με το 7 και έχουν χρώμα ανοιχτό πράσινο (3).



Δημιουργία  
νέων λύσεων  
από  
υφιστάμενες.

Αλυσίδες Kempe

Επιλέγω τις εξετάσεις που ενώνονται με το 3  
και έχουν χρώμα γαλάζιο (5).

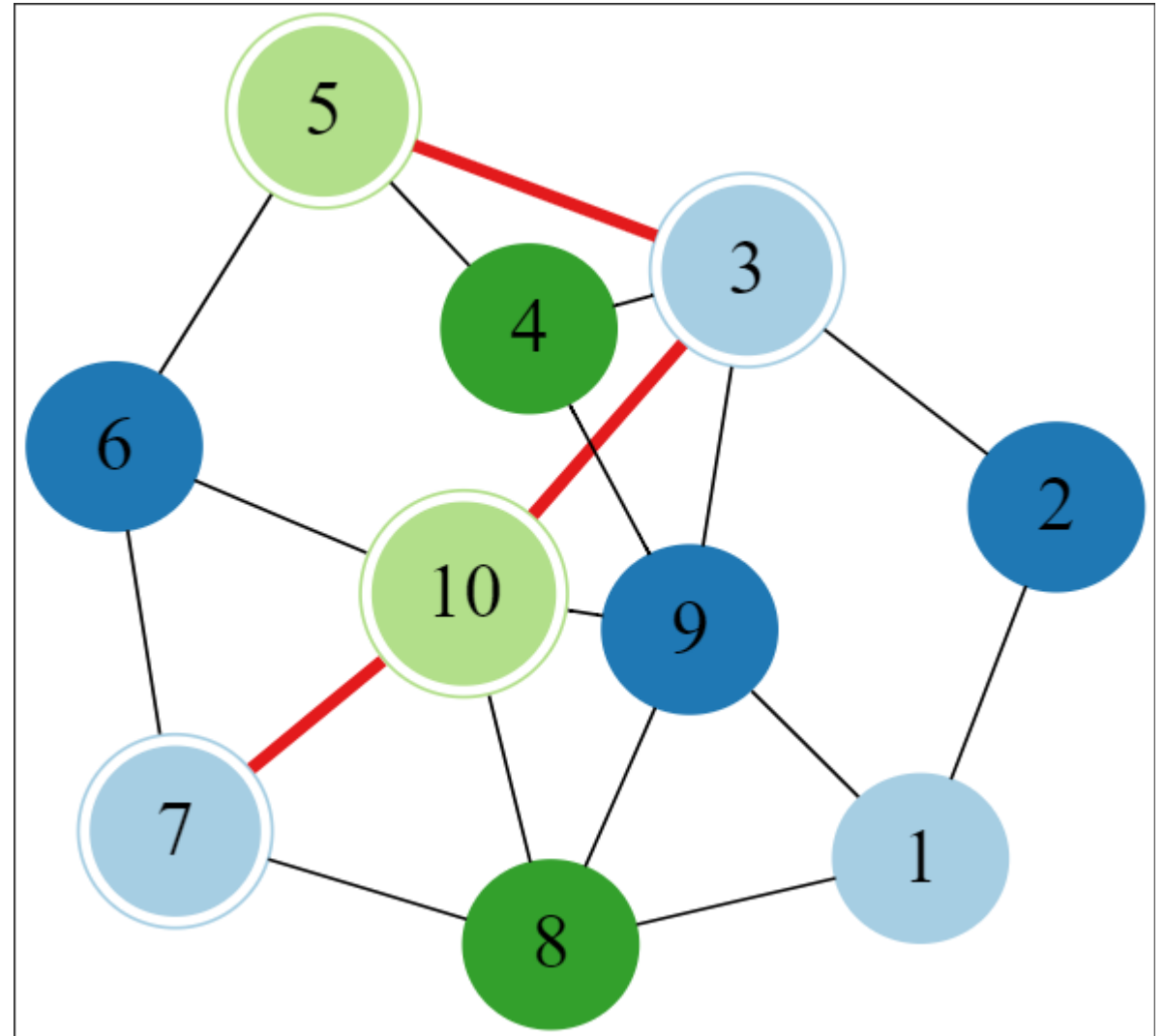




Δημιουργία  
νέων λύσεων  
από  
υφιστάμενες.

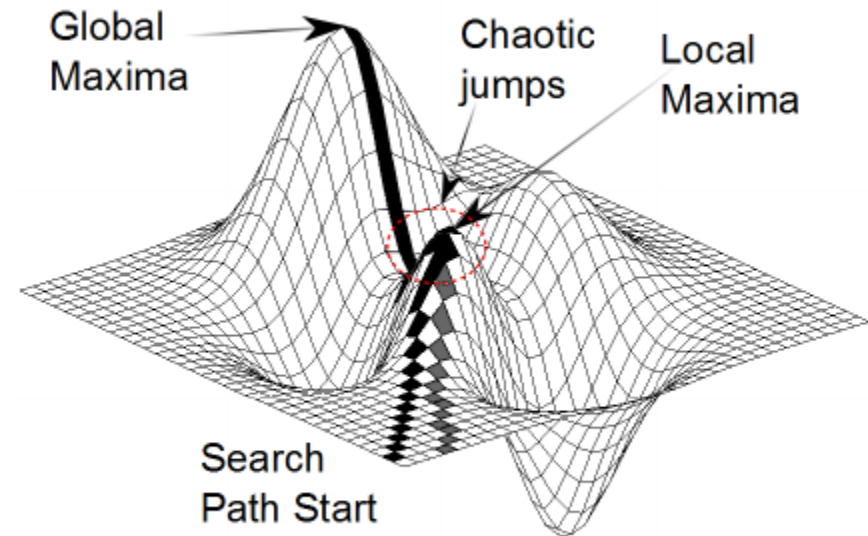
Αλυσίδες Kempe

Στην αλυσίδα Kempe που δημιουργήσα  
ανταλλάσσω τα χρώματα.



# Approximation

## Hill Climbing



# Approximation

## Simulated Annealing



# Η προσέγγιση μου

Φίλτρο  $-\ln \frac{n}{10} e^{(-\frac{x'-x}{x})}$

Απομένων χρόνος	Θερμική Περίοδος ν	Κινήσεις	$-\ln v/_{10}$
600	1	10	2.302585093
540	2	9	1.609437912
480	3	8	1.203972804
420	4	7	0.916290732
360	5	6	0.693147181
300	6	5	0.510825624
240	7	4	0.356674944
180	8	3	0.223143551
120	9	2	0.105360516
60	10	1	0

# Η προσέγγιση μου

```
1 Αρχή
2 Ανάγνωση Αρχείου
3 Υπολογισμός Degree
4  $X = \text{greedyColoring}$ 
5 Διαίρεση σε περιόδους
6
7 Όσο υπάρχει χρόνος
8   Δημιουργία  $X'$ 
9
10   Αν  $E(X') < E(X)$ 
11      $X = X'$ 
12   Αλλιώς  $\Phi\acute{\iota}\lambda\tau\rho\omicron > \text{Τυχαίο } \{0..1\}$ 
13      $X = X'$ 
14
15   Έλεγχος για αλλαγή θερμικής περιόδου
16
17 Τελική λύση  $X$ 
```

# Αποτελέσματα

Dataset	30 min test	Good Solutions	Διαφορά
car-f-92	4.72	3.71	27.22%
car-s-91	5.93	4.93	20.28%
ear-f-83	34.53	32.63	5.82%
hec-s-92	10.64	10.03	6.08%
kfu-s-93	13.83	12.9	7.21%
lse-f-91	11.1	9.82	13.03%
pur-s-93	7.03	4.49	56.57%
rye-s-93	8.97	7.93	13.11%
sta-f-83	157.04	157.03	0.01%
tre-s-92	8.55	7.72	10.75%
uta-s-92	3.87	3.04	27.30%
ute-s-92	25.14	24.77	1.49%
yor-f-83	36.69	34.71	5.70%

ΕΥΧΑΡΙΣΤΩ