Home - Matchmaker Festi

matchmakerireland.com

+353 65 707 4005 | hydrohotel@eircom.net

Festival MATCHMAKING LISDOONVARNA 2016

Home Events Gallery History Meet Willie News Media/PR FAQ Stay Location

Welcome to the Lisdoonvarna Matchmaking Festival – Europe's biggest singles festival!

Days	Hours	Minutes	Seconds
172	18	11	18

Given *n* men and *n* women, can we find a good (heterosexual) matching?

Bob
Dave
Frank
Harry
Joe

Alice
Carol
Elizabeth
Gloria
Isobel

Everybody ranks every possible partner from 1 to *n*

A	C	E	G	I	
1	2	3	4	5	Bob
3	1	5	2	4	Dave
5	4	1	2	3	Frank
1	5	3	4	2	Harry
5	2	1	3	4	Joe

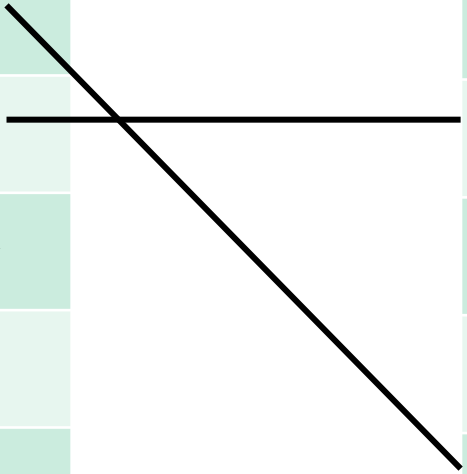
	B	D	F	H	J
Alice	3	4	5	1	2
Carol	1	5	2	4	3
Elizabeth	5	1	4	3	2
Gloria	2	1	3	5	4
Isobel	5	2	1	4	3

A *stable* matching is one where no two people have an incentive to elope with each other.

i.e. we never have P matched with X but prefers Y, at the same time as Y matched with Q but prefers P

A	C	E	G	I	
1	2	3	4	5	Bob
3	1	5	2	4	Dave
5	4	1	2	3	Frank
1	5	3	4	2	Harry
5	2	1	3	4	Joe

	B	D	F	H	J
Alice	3	4	5	1	2
Carol	1	5	2	4	3
Elizabeth	5	1	4	3	2
Gloria	2	1	3	5	4
Isobel	5	2	1	4	3



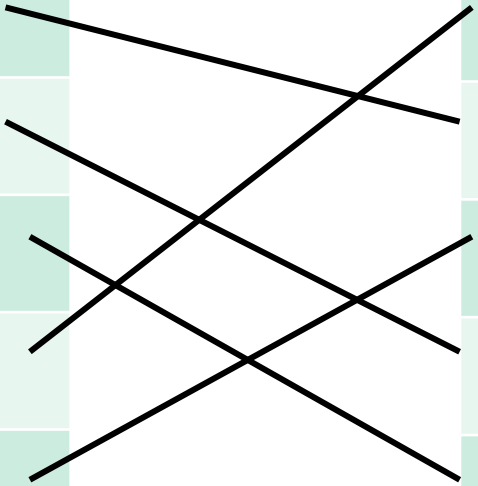
Unstable

because Bob would rather be with Carol than Isobel,
and Carol would rather be with Bob than Dave.

A *stable* matching is one where no two people have an incentive to elope with each other.

i.e. we never have P matched with X but prefers Y, at the same time as Y matched with Q but prefers P

A	C	E	G	I	
1	2	3	4	5	Bob
3	1	5	2	4	Dave
5	4	1	2	3	Frank
1	5	3	4	2	Harry
5	2	1	3	4	Joe



	B	D	F	H	J
Alice	3	4	5	1	2
Carol	1	5	2	4	3
Elizabeth	5	1	4	3	2
Gloria	2	1	3	5	4
Isobel	5	2	1	4	3

Stable

Questions:

Is there always a stable matching?

Can we write an algorithm which is guaranteed to find a stable matching?

How efficient is the algorithm?

Are all stable matchings equal in quality, or are some better than others?

What happens if there are ties (i.e if person X has no strict preference between P and Q)?

What happens if a person does not rank all possible partners?

What happens if we allow m-to-1 matchings instead of 1-1?

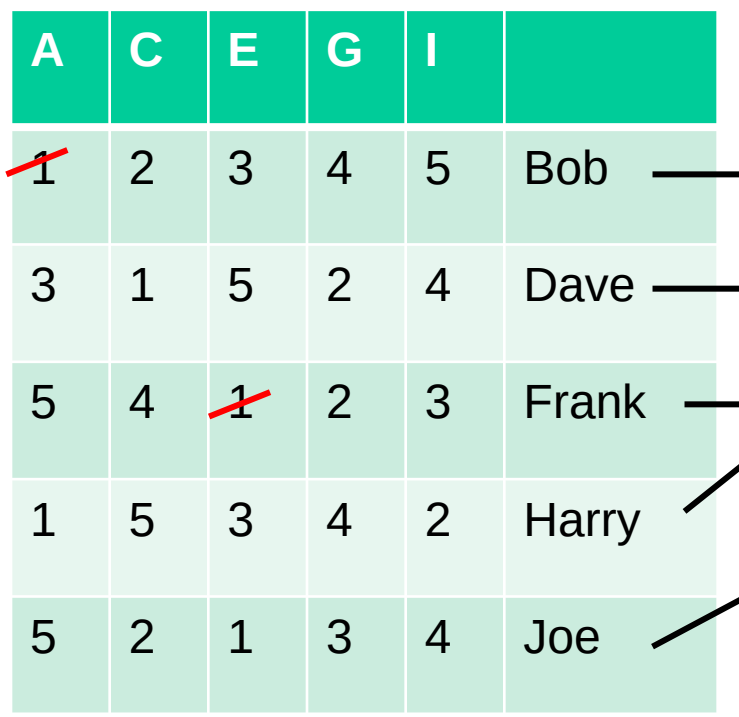
The algorithm:

While there are unmatched men
    select an unmatched man
    if there are no women left on his list
        remove the man
    else the man proposes to the top-ranked woman still on his list
        if the woman is not engaged
            reply with 'OK' to the man and both become engaged
        else if the woman thinks the man is not better than her current partner
            reply with 'no' to the man, who remains unmatched
            the unmatched man removes the woman from his list
        else (and so the woman thinks this man is better than her current partner)
            reply with OK to the man and both become engaged
            send a 'Dear John' letter to current partner, who is now unmatched
            the unmatched man removes the woman from his list

Run of the algorithm

A	C	E	G	I	
1	2	3	4	5	Bob
3	1	5	2	4	Dave
5	4	1	2	3	Frank
1	5	3	4	2	Harry
5	2	1	3	4	Joe

	B	D	F	H	J
Alice	3	4	5	1	2
Carol	1	5	2	4	3
Elizabeth	5	1	4	3	2
Gloria	2	1	3	5	4
Isobel	5	2	1	4	3



Bob, Dave, Frank, Harry, Joe Bob Frank

Run of the algorithm

A	C	E	G	I	
~~1~~	2	3	4	5	Bob
3	~~1~~	5	2	4	Dave
5	4	~~1~~	~~2~~	3	Frank
1	5	3	4	2	Harry
5	2	1	3	4	Joe

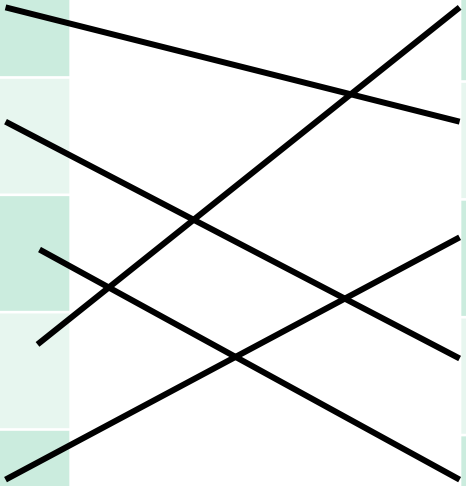
	B	D	F	H	J
Alice	3	4	5	1	2
Carol	1	5	2	4	3
Elizabeth	5	1	4	3	2
Gloria	2	1	3	5	4
Isobel	5	2	1	4	3

~~Bob~~, ~~Frank~~ ~~Dave~~ ~~Frank~~

Final solution

A	C	E	G	I	
~~1~~	2	3	4	5	Bob
3	~~1~~	5	2	4	Dave
5	4	~~1~~	~~2~~	3	Frank
1	5	3	4	2	Harry
5	2	1	3	4	Joe

	B	D	F	H	J
Alice	3	4	5	1	2
Carol	1	5	2	4	3
Elizabeth	5	1	4	3	2
Gloria	2	1	3	5	4
Isobel	5	2	1	4	3



Does the algorithm terminate? ☑

every man steps through their list making proposals, and so will eventually find a match or will reach the end of their list.

Does the algorithm compute a match for everybody? ☑

no woman is ever matched to 2 men at the same time;
no man is ever matched to 2 women at the same time;
once a woman is proposed to, she never becomes unmatched (she only 'trades up').
Suppose some man is never matched. Then some woman is also never matched, since there are equal numbers. That means she is never proposed to. But since all men rank all women, the unmatched man must have proposed to her. Contradiction.

Is the matching stable? ☑

suppose not. Then there are a man M and woman W such that M is matched with X, but prefers W, and W is matched with K, but prefers M. M must have proposed to W before X, and W must have rejected him for a more preferred match. But women never change to a lower ranked match. So W must prefer K to M. Contradiction.

Questions:

Is there always a stable matching?

Can we write an algorithm which is guaranteed to find a stable matching?

We have an algorithm that always terminates, always finds a match for every man and every woman, and always produces a stable matching. So the answer to the 2nd question is 'Yes'.

That means the answer to the 1st question is 'Yes' – there is always a stable matching.

```
While there are unmatched men
    select an unmatched man
    if there are no women left on his list
        remove the man
    else the man proposes to the top-ranked woman still on his list
        if the woman is not engaged
            reply with 'OK' to the man and both become engaged
        else if the woman thinks the man is not better than her current partner
            reply with 'no' to the man, who remains unmatched
            the unmatched man removes the woman from his list
        else (and so the woman thinks this man is better than her current partner)
            reply with OK to the man and both become engaged
            send a 'Dear John' letter to current partner, who is now unmatched
            the unmatched man removes the woman from his list
```

How should we maintain the data in the algorithm?

Unmatched men: we process in some order : so use a queue?

Processing the rankings:

Men: need to move through their list in order, and never add: a list?

A dictionary of lists: key = man, value = ranked list

Women: need to find the ranking for an arbitrary man, never changes: a map (Python dictionary)?

A dictionary of dictionaries: key = woman, value = dictionary of ranks

```
def stablemarriage(m_lists, w_lists):
    m_match = {}   #will contain the final stable matches
    w_match = {}
    singlemen = QueueA()    #a queue of single men
    for m in m_lists:
        singlemen.enqueue(m)
    men_next = {m:0 for m in m_lists}   #next woman to ask
    w_maps = {}       #for each woman, map men to rank
    for w in w_lists:
        w_maps[w] = {}
        for i in range(len(w_lists[w])):
            w_maps[w][w_lists[w][i]] = i
    while not singlemen.is_empty(): #while still unmatched men
        m = singlemen.dequeue()     #get the next single man
        w = m_lists[m][men_next[m]] #get man's next woman
        if w not in w_match:        #if she is not engaged
            w_match[w] = m              #engage them
            m_match[m] = w  #elif prefers man to current partner
        elif w_maps[w][m] < w_maps[w][w_match[w]]:
            singlemen.enqueue(w_match[w])   #jilt current partner
            w_match[w] = m                  #engage man and woman
            m_match[m] = w
        else:
            singlemen.enqueue(m)     #else man unmatched again
        men_next[m] += 1            #get ready for the next woman
    return m_match, w_match
```

Complexity:

O(n) steps to build the initial singleman queue and man_next indices
$O(n^2)$ to build the women rank dictionaries

Each man makes at most n proposals, so $O(n^2)$ proposals in total
each proposal requires no more than a constant number of dequeuing, enqueuing, list lookups, dictionary lookups and dictionary assignments, so total cost of proposals is still $O(n^2)$

Total time complexity is therefore $O(n^2)$

The algorithm as presented here generates the 'man optimal' matching

- every man receives their highest possible ranked woman
  - where 'possible' means could appear in a stable matching

Unfortunately, the same matching is also the 'woman pessimal' matching

- every woman receives their lowest possible ranked man

Switching the order of the input arguments produces 'woman optimal' and 'man pessimal' matchings.

Note:
- incomplete lists and ties still allows stable matchings, for revised definitions of stable
- m-to-1 matchings are also guaranteed to have a stable solution
- matchings where all participants are from the same set have no guaranteed stable solution

Lloyd Shapley and Alvin Roth were awarded the Nobel Prize for Economics in 2012, partly for this algorithm (Gale-Shapley), its extensions, and its associated theory.

Extending the algorithm to m-to-1 matchings allows applications in

- matching medical graduates to hospitals, in US, Canada, Scotland, Norway, ...
  - 30000 graduates at a time
- matching internet users to available servers
- matching passengers to drivers in ride-sharing systems
- matching kidneys to recipients in international organ transplant schemes

Menu

www.nrmp.org

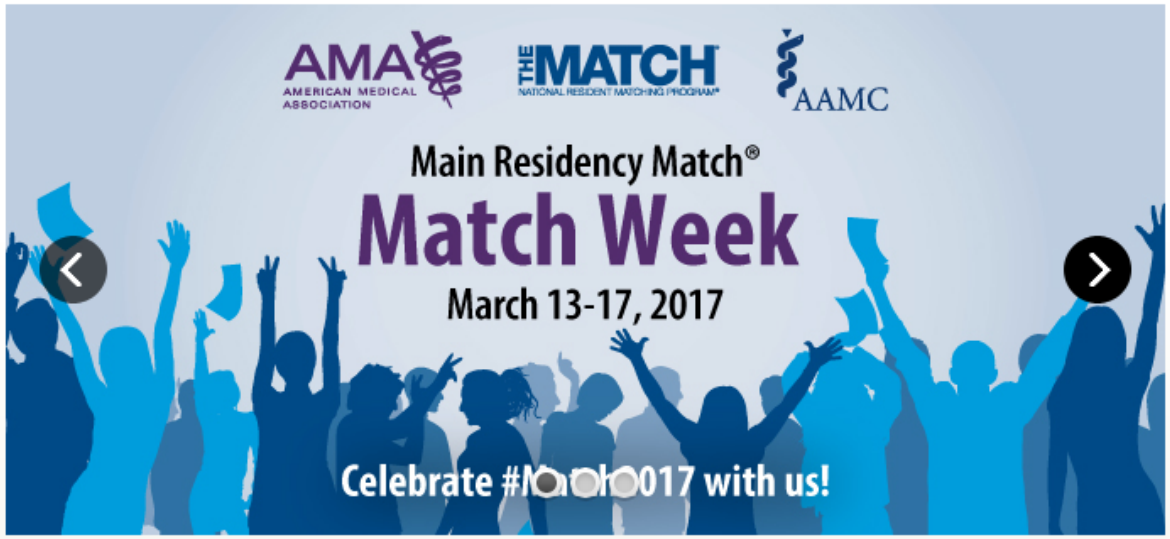
www.nrmp.org

THE MATCH
NATIONAL RESIDENT MATCHING PROGRAM®

ABOUT | NEWS & ANNOUNCEMENTS | THE MATCH, A TO Z | CONTACT US | NRMP

KEYWORD

RESIDENCY | FELLOWSHIP | HOW A MATCH WORKS | POLICIES | MATCH DATA



REGISTER /LOGIN FOR A MATCH

START HERE

REGISTER /LOGIN HELP

RESIDENCY OVERVIEW

EMAIL THIS PAGE

FELLOWSHIP OVERVIEW

The Match provides unparalleled medical matching services in the United States. It's 100% objective, 100% accurate, and 100% committed to a fair and transparent process. With its internationally recognized algorithm, comprehensive data reports, and advanced technology, The Match is helping applicants achieve their dreams.

Getting it right since 1952.

NRMP has released a statement on the Executive Order on immigration.

POLICY CHANGES IMPLEMENTED BY NRMP BOARD OF DIRECTORS The NRMP Board of Directors has implemented changes to NRMP policy for all Matches opening after June 30, 2016. **LEARN WHAT YOU NEED TO KNOW** about applicant-program communication, length of couples' rank order lists, the

BY
ALEX HERN

POLITICS 15 OCTOBER 2012

Trading kidneys, repugnant markets and stable marriages win the Nobel Prize in Economics

Roth and Shapley charted a course for economists to go beyond simply arguing for markets in everything.

The 2012 Nobel Prize in Economics – technically the Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel, but nobody cares – has been awarded to two American Economists, Alvin Roth and Lloyd Shapley "for the theory of stable allocations and the practice of market design". The Nobel

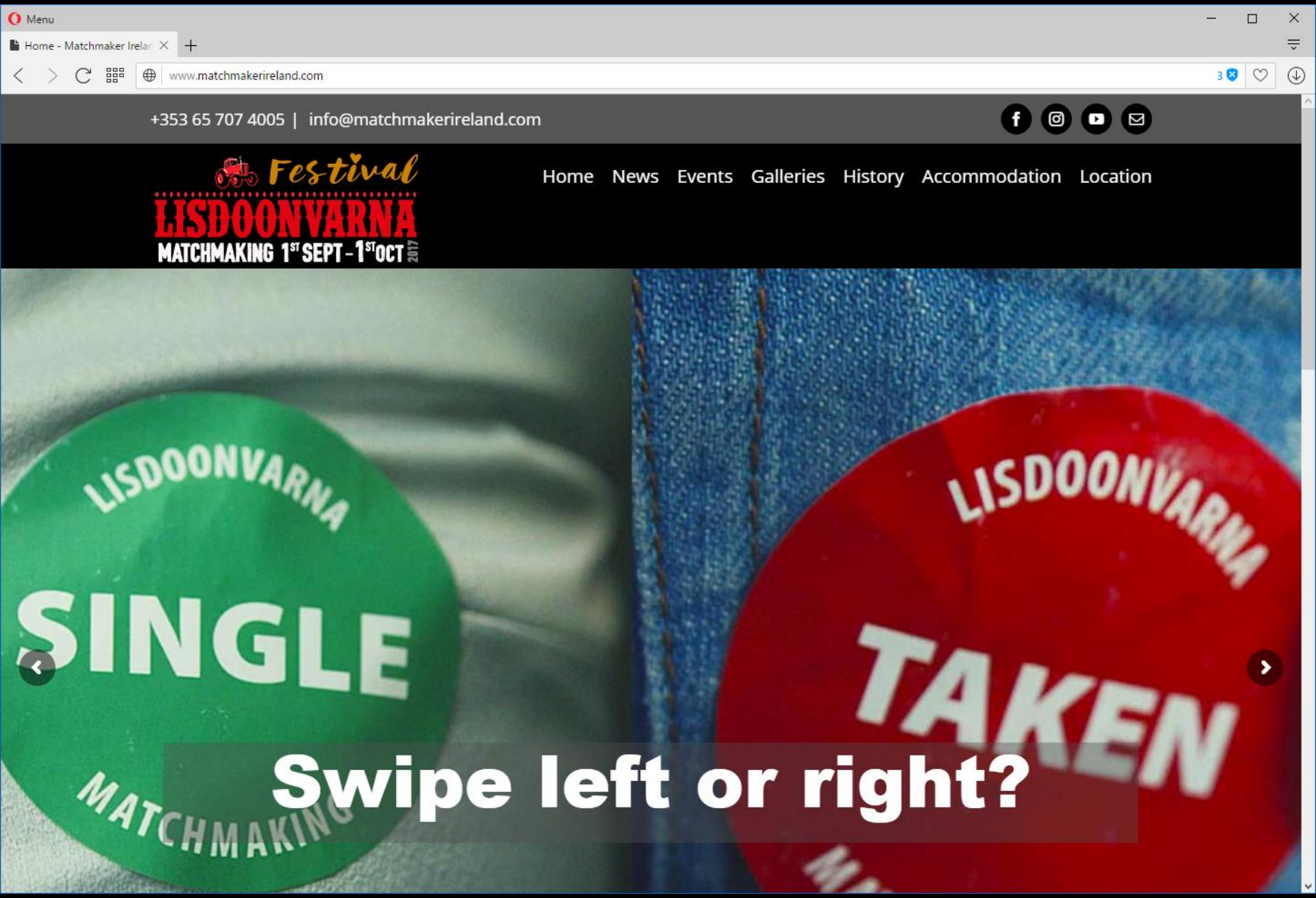
ns what that means:

MOST POPULAR

1 
The American berserk
BY SIMON HEFFER

2 
The thinking behind John McDonnell's new fiscal credibility rule
BY PAUL MASON

3 Helen's story of abuse in The


Menu
Home - Matchmaker Irelar
www.matchmakerireland.com
+353 65 707 4005 | info@matchmakerireland.com
Festival
LISDOONVARNA
MATCHMAKING 1ST SEPT - 1ST OCT 2017
Home News Events Galleries History Accommodation Location
LISDOONVARNA
SINGLE
MATCHMAKING
LISDOONVARNA
TAKEN
Swipe left or right?

END OF MODULE

Next lecture:

Revision and sample exam questions