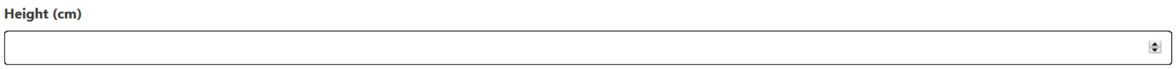
Assignment 2

Equivalence partitioning analysis

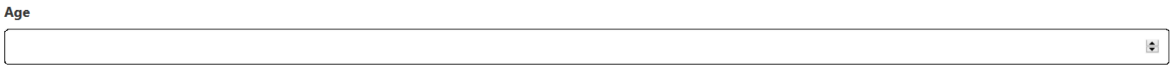
We used the black box technique (we did not look at the code) equivalence partitioning to divide our data into partitions that we will consider as the same. We applied this on the input fields where we expect range limits. The valid partitions are those which would produce the expected result – a match happening.

The values were chosen to represent the real-world situations. One exception is the “name” partitioning where due to lack of time we didn’t implement regular expression for the field. Because of this, we ended up with no invalid partitions. The expected scenario would be to exclude numbers and special characters.



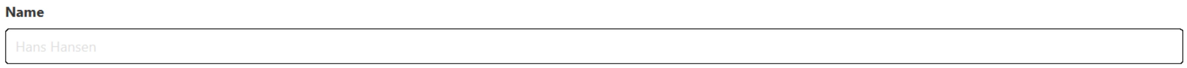
Height (cm)

Equivalence Partitioning (Height)		
Invalid	Valid	Invalid
< 50	50 - 300	300 <



Age

Equivalence Partitioning (Age)		
Invalid	Valid	Invalid
< 18	18 - 120	120 <



Name

Hans Hansen

Equivalence Partitioning (Name)	
Valid	Invalid
Alphabet characters Special characters Numbers	-

Boundary value analysis

We have made a webpage called Keamatch that we are going to make some boundary value analysis on. We do not look at the code in this instance.

we are going to look at the boundary values for height and age. The reason we are going to look at them is because they have a maximum and minimum value.

The minimum boundary values for height are 49, 50, 51. 50 is chosen because that is around the lowest an adult human has been recorded.

The maximum boundary values for height are 299, 300, 301. 300 is chosen because the trend is that the humanity’s height keeps increasing.

The minimum boundary values for age are 17, 18, 19. 18 is chosen because we did not want anyone under the age of 18.

The maximum boundary values for age are 119, 120, 121. 120 is chosen because we believe it is not very probable for people this age to be interested in dating.

Creation of a decision table

Our application has two different algorithms, Basic and Advanced one. The former matches based on age and hobbies, while the latter on gender, height and zodiac sign. The relevant tables bellow specify the actions (outcomes) depending on given conditions.

Basic algorithm

Conditions	Rule 1	Rule 2	Rule 3	Rule 4
Age	F	F	T	T
Hobbies	F	T	F	T
Outcome	0%	0%	50%	100%

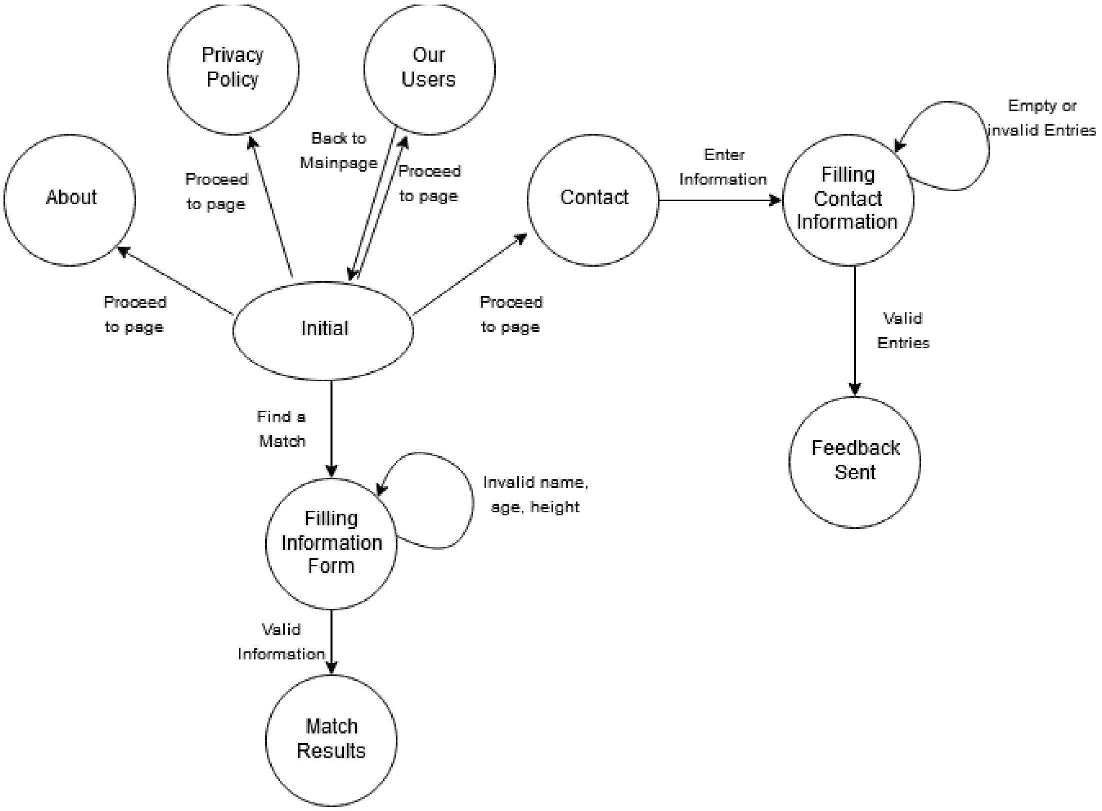
Advanced algorithm

Cond.	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
Gender	F	F	F	F	T	T	T	T
Height	F	F	T	T	F	F	T	T
Zodiac	F	T	F	T	F	T	F	T
Outcome	0%	0%	0%	0%	50%	50%	50%	100%

Legend
F: User and Seeker don’t match this value
T: User and Seeker match this value

A state transition diagram (see pic underneath)

We start in the initial state, where we have the option to choose different kind of actions which will trigger a state transition. Two guards are implemented on two states, that if they are passed (true), they enable for an event to cause a transition to another state.



A state transition table (see pic underneath)

In the state transition table, we show the new states of the application after a certain action has been taken and corresponding event has been activated.

State	Action	Event	Next State
Initial	Matching Form Filled	Check Form	Verify Form
Verify Form		Verification Failed	Initial
Verify Form		Verification Passed	Show Matches
Initial	Contact Button Pressed		Contact
Contact	Contact Form Filled	Check Form	Verify Form
Verify Form		Verification Failed	Contact
Verify Form		Verification Passed	Confirmation
Initial	Our Users Button Pressed		Our Users
Our Users	Back to Main Page Button Pressed		Initial
Initial	Privacy Button Pressed		Privacy
Initial	About Button Pressed		About