EXERCISE 1

We have 4 teachers a, b, c and d that must give 10 lessons. Each lesson can be held by a single teacher. For each lesson we know the starting time and duration Di = 2 hours. Also, we know that no teacher can hold two consecutive classes and even lessons overlapping temporally. Model the problem as a constraint satisfaction problem by defining the variables, the domains of the variables and constraints between these variables. Also assume that the lessons have the following starting times:

- S1 = 7
- S2 = 8
- S3 = 9
- S4 = 10
- S5 = 11
- S6 = 12
- S7 = 13
- S8 = 14
- S9 = 15
- S10 = 16

Model the problem as a CSP and solve it using the forward checking.

SOLUTION: MODELING

Variables: Hours of class X1 ... X10
Domains: Teachers [a, b, c, d]

Constraints:
Unary: X1, … X10 ::[a,b,c,d]
Binary:

- Each lesson can be held by a single teacher: each variable is instantiated with a single value.
- Each teacher cannot hold two consecutive hours of lessons:
  $Si + Di = Sj \rightarrow Xi \neq Xj$
- Each teacher can not keep overlapping two hours of lessons:
  $Si + Di > Sj$ and $Si + Di \leq Sj + Dj \rightarrow Xi \neq Xj$.

SOLUTION: SEARCH

Initially X1 ... X10 :: [a, b, c, d]

- X1 = a
  - X2 X3 :: [b, c, d], X4 ... X10 :: [a, b, c, d]
- X2 = b
  - X3 :: [c, d], X4 :: [a, c, d], X5 ... X10 :: [a, b, c, d]
- X3 = c
  - X4 :: [a, d], X5 :: [a, b, d], X6 ... X10 :: [a, b, c, d]
- X4 = a
  - X5 :: [b, d], X6 :: [b, c, d], X7 ... X10 :: [a, b, c, d]
- X5 = b
  - X6 :: [c, d], X7 :: [a, c, d], X8 ... X10 :: [a, b, c, d]
- X6 = c
  - X7 :: [a, d], X8 :: [a, b, d], ... X9 X10 :: [a, b, c, d]
- X7 = a
  - X8 :: [b, d], X9 :: [b, c, d], X10 :: [a, b, c, d]
- X8 = b
  - X9 :: [c, d], X10 :: [a, c, d]
- X9 = c
  - X10 :: [a, d]
- X10 = a
  - solution

OBSERVATION

- It should be noted that there is another possible representation of the problem which associates teachers a variable whose domain initially contains all lessons.
- The constraints could eliminate those lessons from the domains that are incompatible with the constraints on the teacher.

- However, this representation does not match the perspective of constraint satisfaction problems as in a solution each variable should be assigned to one and only one value and not to a set of values. In fact, in the latter case, a possible solution would be

- Xa :: [1,4,7]

which should mean that the teacher a can keep in classes 1, 4 and 7 while the unary constraints has an ex-or semantics. You might think of a domain of sets but complicates the treatment.

EXERCISE 2

Given a 4 x 4 checkerboard and 4 colors [r, b, g, y], a color must be placed in each cell of the board so that each row, each column and the two diagonals of the board main contain different colors.

Model the problem as a CSP, and will be solve it <u>until the first solution</u> via the forward checking technique with first-fail heuristic (also known as Minimum Remaining Values MRV)

EXERCISE 2

Given a 4 x 4 checkerboard and 4 colors [r, b, g, y], a color must be placed in each cell of the board so that each row, each column and the two diagonals of the board main contain different colors.

Model the problem as a CSP, and will be solve it until the first solution via the forward checking technique with first-fail heuristic (also known as Minimum Remaining Values MRV)

SOLUTION

- Each cell of the checkerboard is a variable X_{11} ..X_{44}. The initial domains of the variables are composed of the four available colors.
- The constraints are
- for all i $X_{ij} \neq X_{ik}$ for $j \neq k$
- for all i $X_{ji} \neq X_{ki}$ for $j \neq k$
- for each i and j $X_{ii} \neq X_{jj}$ with $i \neq j$
- for each i and j, $X_{i,4-i+1} \neq X_{j,4-j+1}$ with $i \neq j$

EXERCISE 2

r, b y, g	r, b y, g	r, b y, g	r, b y, g
r, b y, g	r, b y, g	r, b y, g	r, b y, g
r, b y, g	r, b y, g	r, b y, g	r, b y, g
r, b y, g	r, b y, g	r, b y, g	r, b y, g

X11 = r

r	b y, g	b y, g	b y, g
b y, g	b y, g	r, b y, g	r, b y, g
b y, g	r, b y, g	b y, g	r, b y, g
b y, g	r, b y, g	r, b y, g	b y, g

X12 = b

r	b	y, g	y, g
b y, g	y, g	r, b y, g	r, b y, g
b y, g	r y, g	b y, g	r, b y, g
b y, g	r y, g	r, b y, g	b y, g

X13 = y

r	b	y	g
b y, g	y, g	r, b g	r, b y, g
b y, g	r y, g	b g	r, b y, g
b y, g	r y, g	r, b g	b y, g

X14 = g

r	b	y	g
b y, g	y, g	r, b	r, b y
b y, g	r y	b g	r, b y
b y	r y, g	r, b g	b y

X22 = y

r	b	y	g
b g	y	r, b	r, b
b y, g	r	b g	r, b y
b y	r g	r, b g	b

X32 = r

r	b	y	g
b g	y	b	r, b
b y, g	r	b g	b y
b y	g	r, b g	b

X23 = b

r	b	y	g
g	y	b	r
b y, g	r	g	b y
y	g	r g	b

X24 = r

r	b	y	g
g	y	b	r
b y, g	r	g	b y
y	g	rg	b

X21 = g

EXERCISE 2

r	b	y	g
g	y	b	r
b y	r	g	b y
y	g	r g	b

X33 = g

r	b	y	g
g	y	b	r
b y	r	g	b y
y	g	r	b

X41 = y

r	b	y	g
g	y	b	r
b	r	g	b y
y	g	r	b

X31 = b

r	b	y	g
g	y	b	r
b	r	g	y
y	g	r	b

EXERCISE 3

- Suppose we have the following constraints:
  $X < Y,\ X \neq K,\ Y + 5 \leq K,\ Y + 7 > Z,\ X \leq Z$
  defined on the variables X, Y, Z, K whose domain of definition is [1..20].

- Solve the problem by applying the strategy of <u>full look ahead</u>.

  THE MODEL IS GIVEN
  $X < Y,\ X \neq K,\ Y + 5 \leq K,\ Y + 7 > Z,\ X \leq Z$
  X, Y, Z, K :: [1..20]

SOLUTION

- X = 1
- full look ahead Propagation:
  - Y :: [2..15], K :: [7..20], Z :: [1..20]
- Y = 2
  - K :: [7..20], Z :: [1..8]
- K = 7
  - Z :: [1..8]
- Z = 1 <u>solution</u>

EXERCISE 4

- You must visit six clients A, B, C, D, E, F during working hours (from 9 to 19). (Implicit duration 1 hour)
  - Two clients cannot be visited simultaneously.
  - We know that customers C and F must be visited before the customer D.
  - A is a customer out of town, while B, C, D, E and F are all in the center. Then, the trip from A to any other customer takes 1 hour, while any trip in the city center is carried out at negligible time (= 0).
  - The client C can only be visited from 15 to 17.

- Model the problem as a CSP. Explore the search tree <u>until the first solution</u> with standard backtracking and forward checking and comment the results.

SOLUTION

- Variables: Customer
- Domains: possible visiting times A, B, C, D, E, F :: [9..18]
- Constraints:
  - Two clients can not be visited at the same time:

    for $\forall$ **X, Y** **X $\neq$ Y**
  - Customers C and F must be visited before the customer's D:

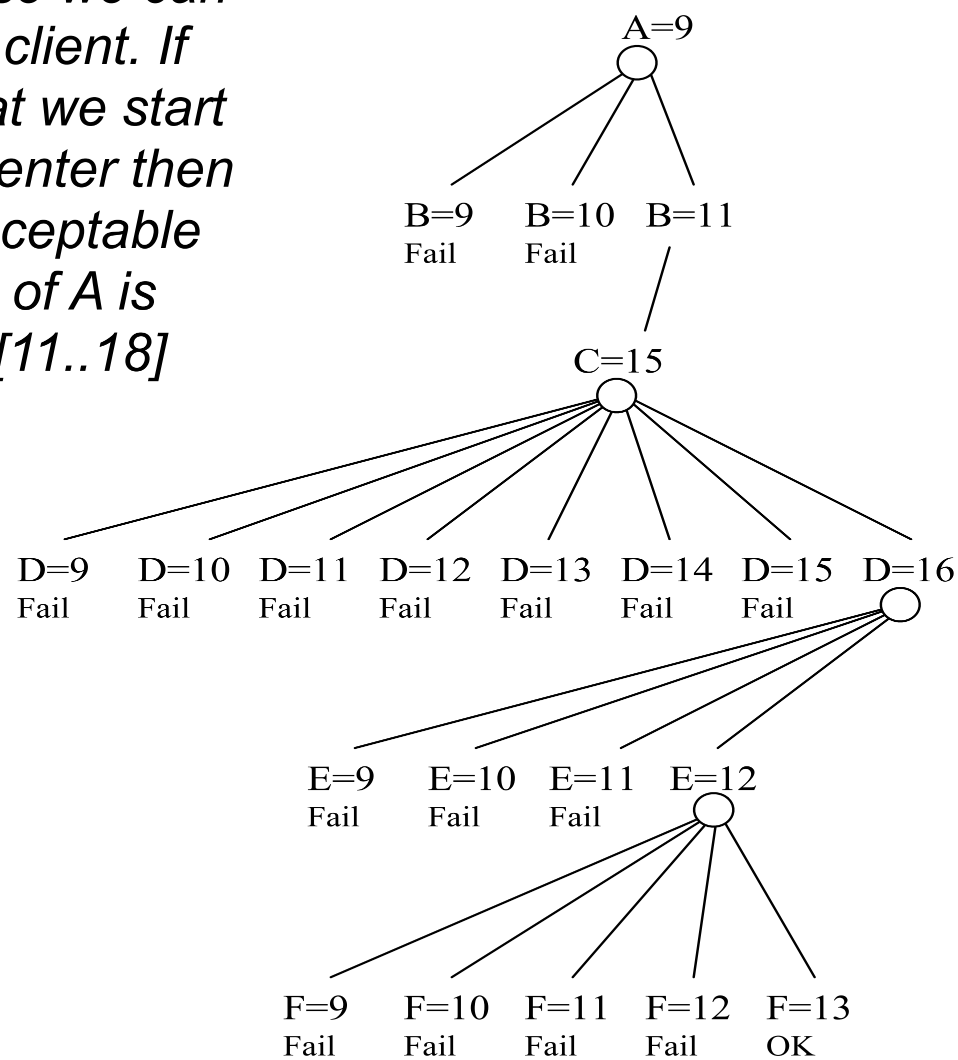
    C <D **F <D**
  - A is a customer out of town, while B, C, D, E and F are all in the center. Then, to move from A to any other customer takes 1 hour, while any shift in the city center is carried out at negligible time (= 0).

    $\forall$ **X** $\in$ **[B, C, D, E, F]** **A $\geq$ X + 2 OR X $\geq$ A + 2**
  - The client C can only be visited from 15 to 17.
  - Unary constraint on C which reduces its rule

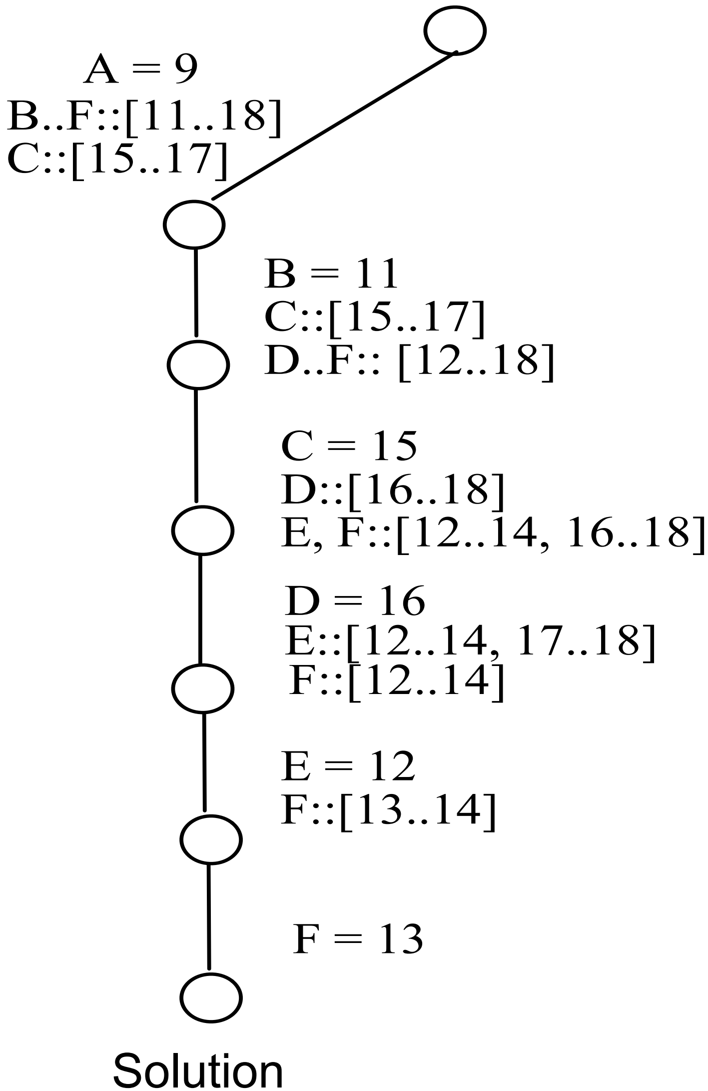
    C $\geq$ 15, C $\leq$ 17

STANDARD BACKTRACKING

Hp: we suppose we can start from any client. If the hp was that we start from the city center then A = 9 is not acceptable as the domain of A is reduced to A: [11..18]



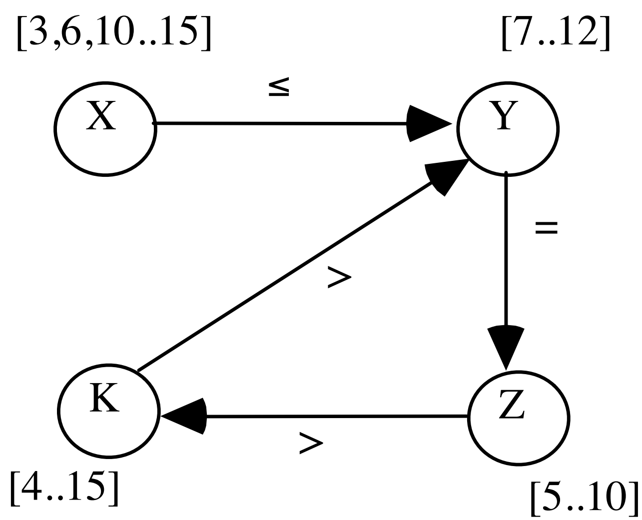
FORWARD CHECKING



NOTE no failures here while
With standard backtracking
we have 16 failures

EXERCISE 5

Consider the following constraints network



X :: [3,6,10..15], Y :: [7..12], K :: [4..15]

Z :: [5..10] Y = Z, Z < K, K > Y, X ≤ Y

Apply arc-consistency. In addition discuss what happens by applying the arc-consistency to the same network if we add the constraint X = K.

SOLUTION

```
X :: [3,6,10..15], Y :: [7..12], K :: [4..15]
Z :: [5..10] Y = Z, Z <K, K> Y, X ≤ Y
```

Arc-consistency

```
X = [3, 6, 10]
Z = [7..10]
K = [8..15]
Y = [7..10]
```

Introducing the new constraint leads to a failure. Note the incremental computation

EXERCISE 6

- Given the following constraints:

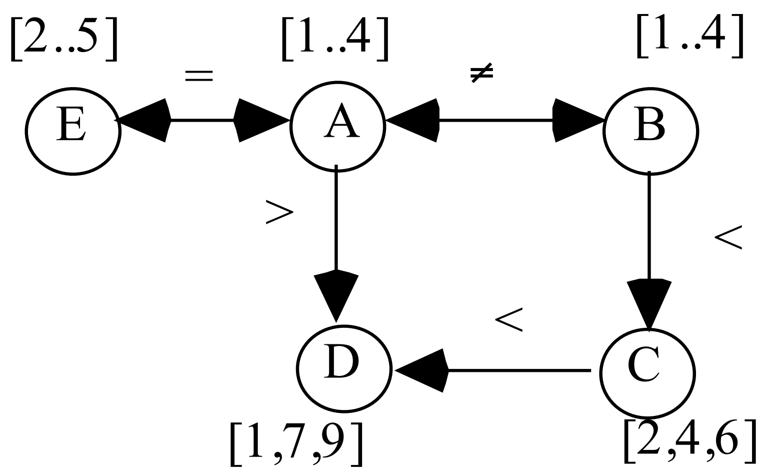
  A :: [1..4], B :: [1..4], C :: [2,4,6] D :: [1,7,9], E :: [2..5]

  A> D, A ≠ B, B < C, C > D, E = A

- Draw the graph corresponding to the constraint satisfaction problem and apply arc-consistency. Show the search tree to get to the first solution using as a heuristic of the first-fail assignment of values to variables (MRV) is at each instantiation apply the arc-consistency to the remaining network.

SOLUTION

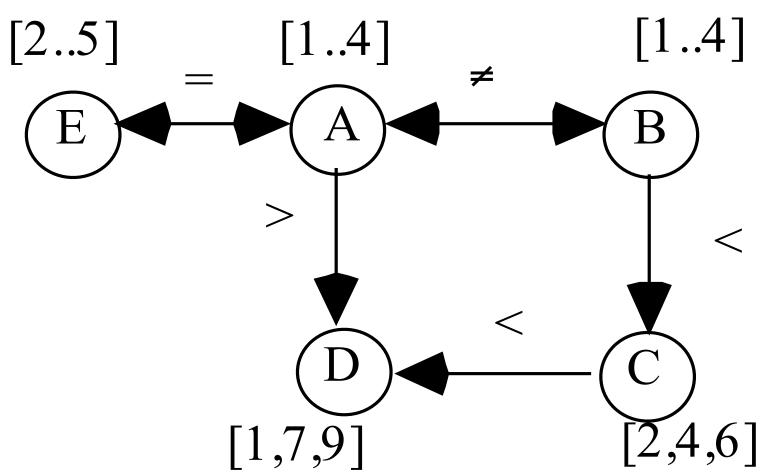
Graph corresponding to the problem



We do not eliminate anything because any value in B as a consistent number in C

SOLUTION

Graph corresponding to the problem



After the application of the Arc-consistency to the original problem we obtain

`A :: [2..4]`

`B :: [1..4]`

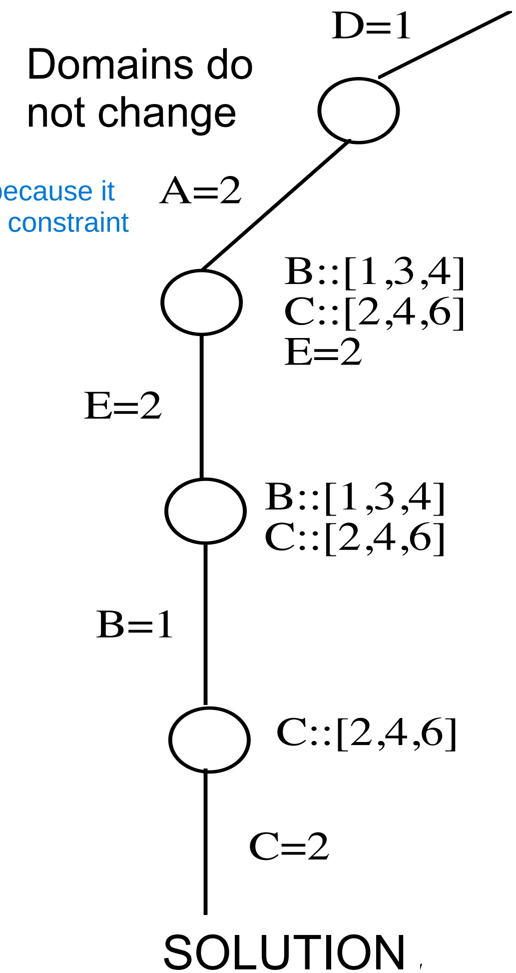
`C :: [2, 4, 6]`

`D = 1`

`E :: [2..4]`

SEARCH

we start with A because it
appears in more constraint
(most difficult)



SOLUTION
A = 2, B = 1, C = 2, D = 1, E = 2