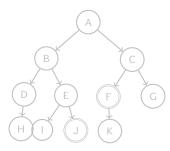
Ch <- Premier(Liste); fin <- Faux; while Ch!= {} et non(fin) do Liste <- Liste / {Ch}; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf {n1}; fin <- Vrai; else Liste <- Liste Uf {(Ch Uf {n1})}; n <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
-	-	-	-	{{ A }}	

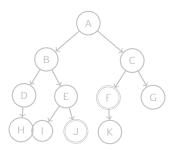


Ch <- Premier(Liste); fin <- Faux;

return fin;

$$\label{eq:while_chieff} \begin{split} & \text{while } Ch! = || \text{ et non}(fin) \text{ do} \\ & \text{Liste} \cdot \text{Liste} / \{Ch]; n <- \text{dernierNoeud}(Ch); \\ & 1 <- \text{successeur}(n); \\ & \text{while non}(fin) \text{ et } n1 \text{ est valide do} \\ & \text{if } n1 \text{ est solution then} \\ & \text{Sol } Ch \text{ Uf } \{n1\}; \text{fin} <- \text{Vrai}; \\ & \text{else } \text{Liste} \cdot \text{Liste } \text{Uf } \{(\text{Ch Uf } \{n1\})\}; \\ & n1 <- \text{successeur}(n); \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

n	n1	fin	Liste	Sol
-	-	Faux	{{ A }}	
	<u>-</u>			



Ch <- Premier(Liste); fin <- Faux; while Ch!= {} et non(fin) do

Liste <- Liste / {Ch}; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(fin) et n1 est valide do

if n1 est solution then

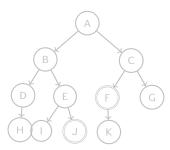
Sol Ch Uf $\{n1\}$; fin <- Vrai;

else Liste <- Liste Uf {(Ch Uf {n1})};

n1 <- successeur(n):

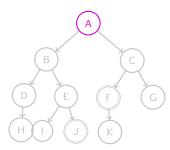
Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	-	-	Faux	{{ A }}	



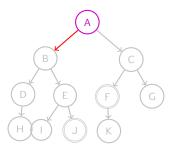
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	-	Faux	{}	



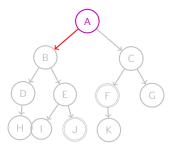
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	В	Faux	{}	



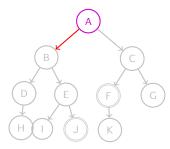
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch UF [n1]; fin <- Vrai; else Liste <- Liste UF [(Ch UF [n1])); n1 <- successeur(n); Ch Premier(Liste); return fin;

Ch	n	n1	fin	Liste	Sol
{A}	Α	В	Faux	{}	



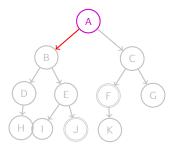
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	В	Faux	{}	



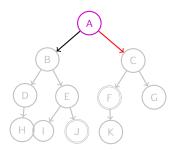
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])]; n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	В	Faux	{{AB}}	



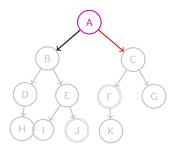
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	С	Faux	{{AB}}	



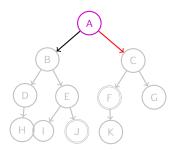
Ch <- Premier(Liste); fin <- Faux; while Ch!= {} et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])]; n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	С	Faux	{{AB}}	



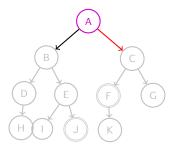
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	С	Faux	{{AB}}	



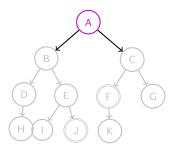
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} <\text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n} 1 <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} <\text{-Vrai;} \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{split}$$

Ch	n	n1	fin	Liste	Sol
{A}	Α	С	Faux	{{AB},{AC}}	



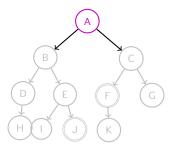
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	Т	Faux	{{AB},{AC}}	



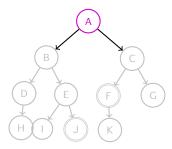
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A}	Α	Τ	Faux	{{AB},{AC}}	



$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!=} \; | \; \text{th ron}(\text{fin}) \; \text{do} \\ & \text{Liste} <\text{-Liste} \; / \; | \text{Ch} \; ; \; \text{n} \; <\text{-dernierNoeud(Ch)}; \\ & \text{11} \; <\text{-successeur(n)}; \\ & \text{while non(fin)} \; \text{et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]}; & \text{fin} \; <\text{-Vrai}; \\ & \text{else Liste} \; <\text{-Liste UF [(Ch UF [n1])]}; \\ & \text{n1} \; <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste	Sol
{A} {AB}	А		Faux	{{AB},{AC}}	



Ch <- Premier(Liste); fin <- Faux;

while Ch!= {} et non(fin) do

Liste <- Liste / {Ch}; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(fin) et n1 est valide do

if n1 est solution then

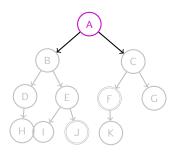
Sol Ch Uf $\{n1\}$; fin <- Vrai;

else Liste <- Liste Uf {(Ch Uf {n1})};

n1 <- successeur(n):

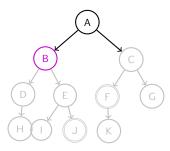
Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A	Τ	Faux	{{AB},{AC}}	



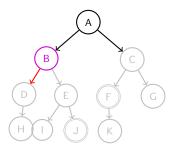
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B		Faux	{{AB},{AC}} { <mark>AC</mark> }	



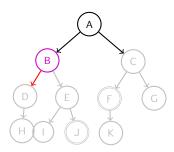
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} <\text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]}; & \text{fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	D D	Faux	{{AB},{AC}} {AC}	



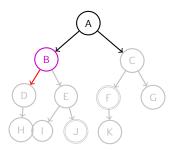
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	D D	Faux	{{AB},{AC}} {AC}	



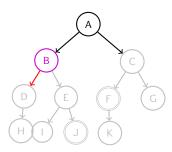
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	D D	Faux	{{AB},{AC}} {AC}	



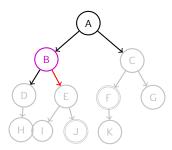
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])]; n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	D D	Faux	{{AB},{AC}} {{AC},{ABD}}	



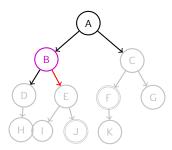
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	E	Faux	{{AB},{AC}} {{AC},{ABD}}	



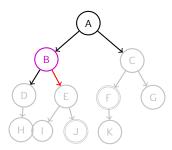
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!=} \; | \; \text{th ron}(& \text{fin}) \; \text{do} \\ & \text{Liste} <\text{-Liste} \; / \; | \; \text{Ch} \; ; \; \text{n} \; <\text{-dernierNoeud(Ch)}; \\ & \text{11} \; <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} \; <\text{-Vrai}; \\ & \text{else Liste} \; <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} \; <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	E	Faux	{{AB},{AC}} {{AC},{ABD}}	



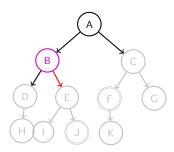
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste	Sol
{A} {AB}	A B	E	Faux	{{AB},{AC}} {{AC},{ABD}}	



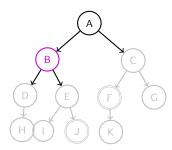
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])]; n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	A	±	Faux	{{AB},{AC}}
{AB}	B	E		{{AC},{ABD},{ABE}}



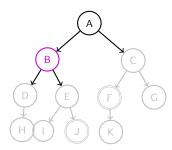
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	Α		Faux	{{AB},{AC}}
{AB}	В	1		{{AC},{ABD},{ABE}}



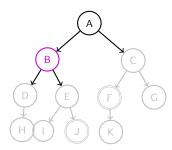
Ch <- Premier(Liste); fin <- Faux; while Ch!= {} et non(fin) do Liste <- Liste / {Ch}; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf {n1}; fin <- Vrai; else Liste <- Liste Uf {(Ch Uf {n1})}; n <- vrai; ch <- successeur(n); ch Premier(Liste);

Ch	n	n1	fin	Liste
{A} {AB}	A B		Faux	{{AB},{AC}} {{AC},{ABD},{ABE}}



$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} <\text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]}; & \text{fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch UF [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В	1		{{AC},{ABD},{ABE}}
{AC}				



Ch <- Premier(Liste); fin <- Faux;

while Ch!= {} et non(fin) do

Liste <- Liste / {Ch}; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(fin) et n1 est valide do

if n1 est solution then

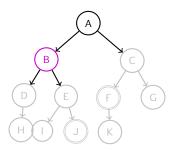
Sol Ch Uf $\{n1\}$; fin <- Vrai;

else Liste <- Liste Uf $\{(Ch\ Uf\ \{n1\})\};$

n1 <- successeur(n);

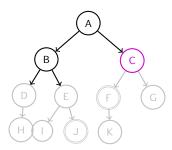
Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	Α		Faux	{{AB},{AC}}
{AB}	В	1		{{AC},{ABD},{ABE}}
{AC}				



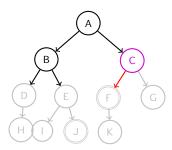
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
{AC}	С			{{ABD},{ABE}}



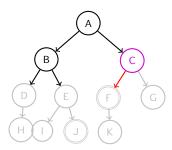
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} <\text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]}; & \text{fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
(AC)	С	F		{{ABD},{ABE}}



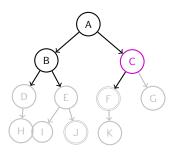
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Chl} = || \text{ et non(fin) do} \\ & \text{Liste} < \text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n} 1 <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n} 1 <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \end{split}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
{AC}	С	F		{{ABD},{ABE}}



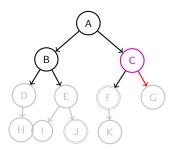
Ch <- Premier(Liste); fin <- Faux; while Ch!= || et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])); n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
(AC)	С	F		{{ABD},{ABE}}



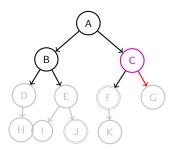
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} <\text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
{AC}	С	F	Vrai	{{ABD},{ABE}}



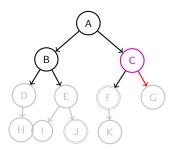
$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} <\text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]}; & \text{fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
(AC)	С	G	Vrai	{{ABD},{ABE}}



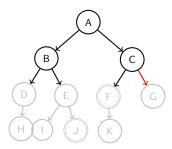
Ch <- Premier(Liste); fin <- Faux; while Ch!= {} et non(fin) do Liste <- Liste / [Ch]; n <- dernierNoeud(Ch); n1 <- successeur(n); while non(fin) et n1 est valide do if n1 est solution then Sol Ch Uf [n1]; fin <- Vrai; else Liste <- Liste Uf [(Ch Uf [n1])]; n1 <- successeur(n); Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В	1		{{AC},{ABD},{ABE}}
{AC}	С	G	Vrai	{{ABD},{ABE}}



$$\begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Chl} = || \text{ et non(fin) do} \\ & \text{Liste} < \text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n} 1 <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch UF [n1])]}; \\ & \text{n} 1 <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \\ \end{aligned}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В	1		{{AC},{ABD},{ABE}}
{AC}	С	G	Vrai	{{ABD},{ABE}}



Ch <- Premier(Liste); fin <- Faux;

while Ch!= {} et non(fin) do

Liste <- Liste / {Ch}; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(fin) et n1 est valide do

if n1 est solution then

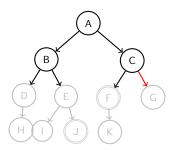
Sol Ch Uf $\{n1\}$; fin <- Vrai;

else Liste <- Liste Uf {(Ch Uf {n1})};

n1 <- successeur(n):

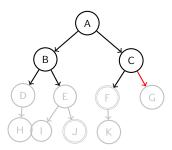
Ch Premier(Liste);

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В	1		{{AC},{ABD},{ABE}}
{AC}	С	G	Vrai	{{ABD},{ABE}}



$$\label{eq:Characteristics} \begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} < \text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \end{split}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
(AC)	С	G	Vrai	{{ABD},{ABE}}



$$\label{eq:Characteristics} \begin{split} & \text{Ch} <\text{-Premier(Liste)}; & \text{fin} <\text{-Faux}; \\ & \text{while Ch!} = || \text{ et non(fin) do} \\ & \text{Liste} < \text{-Liste} / |\text{Ch}|; & \text{n} <\text{-dernierNoeud(Ch)}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{while non(fin) et n1 est valide do} \\ & \text{if n1 est solution then} \\ & \text{Sol Ch UF [n1]; fin} <\text{-Vrai}; \\ & \text{else Liste} <\text{-Liste UF [(Ch Uf [n1])]}; \\ & \text{n1} <\text{-successeur(n)}; \\ & \text{Ch Premier(Liste)}; \end{split}$$

Ch	n	n1	fin	Liste
{A}	Α	1	Faux	{{AB},{AC}}
{AB}	В			{{AC},{ABD},{ABE}}
(AC)	С	G	Vrai	{{ABD},{ABE}}

