while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

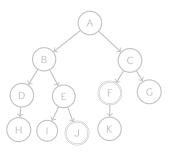
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

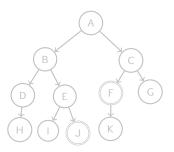
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

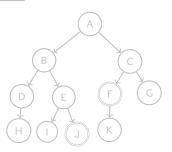
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

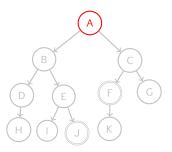
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

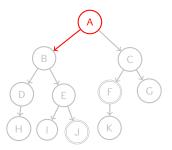
Sol Ch Uf n1 : n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| Ch | n | n1 | fin | Liste |
|-----|---|----|------|-------|
| {A} | Α | В | Faux | {} |
| | | | | |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

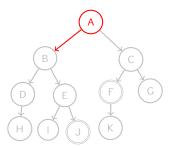
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| Ch | n | n1 | fin | Liste |
|-----|---|----|------|-------|
| {A} | Α | В | Faux | {} |
| | | | | |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

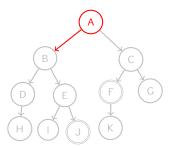
Sol Ch Uf n1; n <- Vrai;

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

n1 < - successeur(n);

Ch Premier(Liste);

| | Ch | n | n1 | fin | Liste |
|---|-----|---|----|------|-------|
| ĺ | {A} | Α | В | Faux | {} |
| | | | | | |
| | | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

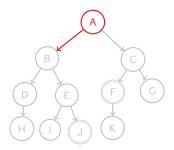
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

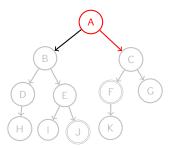
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| | Ch | n | n1 | fin | Liste |
|---|-----|---|----|------|--------|
| ĺ | {A} | Α | С | Faux | {{AB}} |
| | | | | | |
| | | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

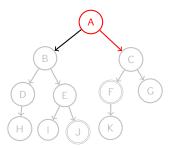
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

Sol Ch Uf n1: n <- Vrai:

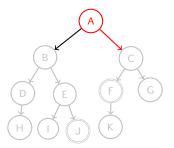
Sol Ch Uf n1; n <- Vrai

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

n1 < - successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

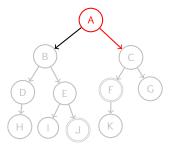
Sol Ch Uf n1 : n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

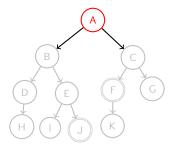
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| | Ch | n | n1 | fin | Liste |
|---|-----|---|----|------|-------------|
| ĺ | {A} | Α | | Faux | {{AB},{AC}} |
| | | | | | |
| | | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

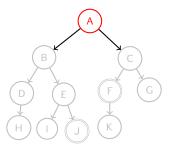
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| | Ch | n | n1 | fin | Liste |
|---|-----|---|----|------|-------------|
| ĺ | {A} | Α | | Faux | {{AB},{AC}} |
| | | | | | |
| ı | | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

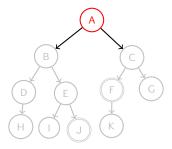
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| Ch | n | n1 | fin | Liste |
|------|---|----|------|-------------|
| {A} | Α | | Faux | {{AB},{AC}} |
| {AB} | | | | |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

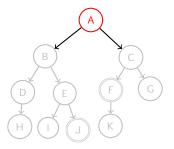
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

| | Ch | n | n1 | fin | Liste |
|---|------|---|----|------|-------------|
| Γ | {A} | Α | | Faux | {{AB},{AC}} |
| | {AB} | | | | |
| | | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

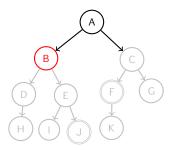
Sol Ch Uf n1: n <- Vrai:

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 | | | {AC} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

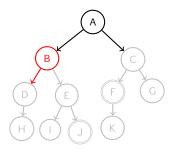
Sol Ch Uf n1: n <- Vrai:

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 | D | | {AC} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

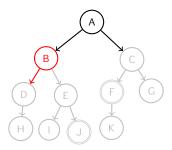
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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 | D | | {AC} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

Sol Ch Uf n1: n <- Vrai:

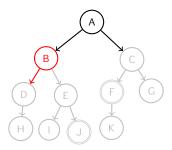
Sol Ch Uf n1; n <- Vra

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 | D | | {AC} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

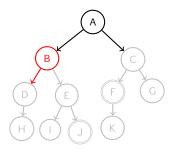
Sol Ch Uf n1: n <- Vrai:

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 | D | | {{AC},{ABD}} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

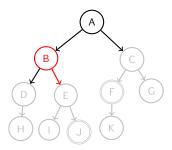
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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 | Ε | | {{AC},{ABD}} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

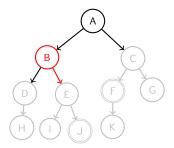
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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 | Ε | | {{AC},{ABD}} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

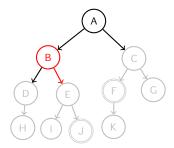
Sol Ch Uf n1; n <- Vrai;

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 | Ε | | {{AC},{ABD}} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

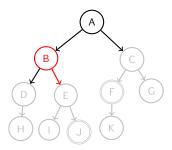
Sol Ch Uf n1: n <- Vrai:

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 | E | | {{AC},{ABD},{ABE}} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

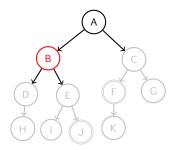
Sol Ch Uf n1: n <- Vrai:

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 | | | {{AC},{ABD},{ABE}} |
| | | | | |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then Sol Ch Uf n1: n <- Vrai:

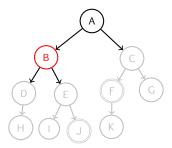
else Liste <- Liste Uf (Ch Uf n1);

n1 <- successeur(n);

Ch Premier(Liste);

....

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then Sol Ch Uf n1: n <- Vrai:

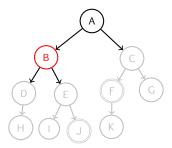
else Liste <- Liste Uf (Ch Uf n1);

n1 <- successeur(n);

Ch Premier(Liste);

Cit Freitiler (Liste

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

Sol Ch Uf n1: n <- Vrai:

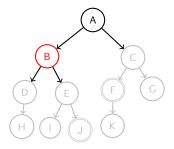
else Liste <- Liste Uf (Ch Uf n1);

n1 <- successeur(n);

Ch Premier(Liste);

return n;

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

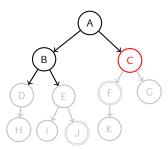
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

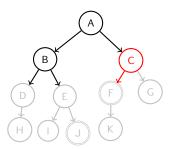
Sol Ch Uf n1 : n <- Vrai:

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 | | | {{AC},{ABD},{ABE}} |
| {AC} | С | F | | {{ABD},{ABE}} |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

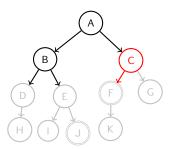
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

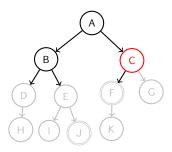
Sol Ch Uf n1; n <- Vrai;

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 | | | {{AC},{ABD},{ABE}} |
| {AC} | С | F | | {{ABD},{ABE}} |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

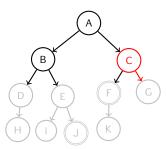
Sol Ch Uf n1; n <- Vrai;

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

n1 < - successeur(n);

 $Ch\ Premier(Liste);$

| Ch | n | nI | fin | Liste |
|------|---|----|------|--------------------|
| {A} | Α | | Faux | {{AB},{AC}} |
| {AB} | В | | | {{AC},{ABD},{ABE}} |
| {AC} | С | F | Vrai | {{ABD},{ABE}} |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

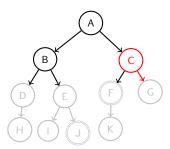
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then

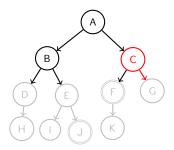
Sol Ch Uf n1: n <- Vrai:

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 | | | {{AC},{ABD},{ABE}} |
| {AC} | С | G | Vrai | {{ABD},{ABE}} |



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

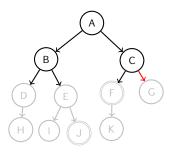
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

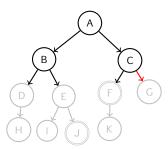
if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do if n1 est solution then

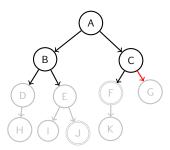
Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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



while Ch!= et non(n) do

Liste <- Liste Ch; n <- dernierNoeud(Ch);

n1 <- successeur(n);

while non(n) et n1 est valide do

if n1 est solution then Sol Ch Uf n1: n <- Vrai:

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

n1 <- successeur(n);

Ch Premier(Liste);

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

