

Project_C

Generated by Doxygen 1.9.1

| | |
|--|----------|
| 1 README | 1 |
| 2 File Index | 3 |
| 2.1 File List | 3 |
| 3 File Documentation | 5 |
| 3.1 Contribution au projet/sources/automate.c File Reference | 5 |
| 3.1.1 Function Documentation | 5 |
| 3.1.1.1 alloc_mat_vide() | 5 |
| 3.1.1.2 alloc_tab_1D() | 5 |
| 3.1.1.3 generator_automate() | 6 |
| 3.2 Projet_C/sources/automate.c File Reference | 6 |
| 3.2.1 Function Documentation | 7 |
| 3.2.1.1 add_state_to_composite_table() | 7 |
| 3.2.1.2 add_state_to_rename_table_state() | 8 |
| 3.2.1.3 affichage_1D() | 8 |
| 3.2.1.4 affichage_2D() | 8 |
| 3.2.1.5 affichage_automate() | 9 |
| 3.2.1.6 affichage_case_mat() | 9 |
| 3.2.1.7 affichage_case_tab_1D() | 9 |
| 3.2.1.8 affichage_case_tab_2D() | 10 |
| 3.2.1.9 affichage_col_mat() | 10 |
| 3.2.1.10 affichage_ligne_mat() | 10 |
| 3.2.1.11 affichage_mot() | 11 |
| 3.2.1.12 allocation_mat_vide() | 11 |
| 3.2.1.13 allocation_tab_1D() | 11 |
| 3.2.1.14 allocation_tab_2D() | 12 |
| 3.2.1.15 calculte_number_states_touch() | 12 |
| 3.2.1.16 complement_automate() | 12 |
| 3.2.1.17 concat() | 13 |
| 3.2.1.18 copie_automate() | 13 |
| 3.2.1.19 copy_line_mat() | 13 |
| 3.2.1.20 del_automate() | 14 |
| 3.2.1.21 echange() | 14 |
| 3.2.1.22 est_complet() | 14 |
| 3.2.1.23 est_deterministe() | 15 |
| 3.2.1.24 est_reconnu() | 15 |
| 3.2.1.25 est_un_etat_final() | 15 |
| 3.2.1.26 exist_state_in_tab_transi() | 16 |
| 3.2.1.27 find_next_state() | 16 |
| 3.2.1.28 generate_automate() | 16 |
| 3.2.1.29 generate_automate_null() | 16 |
| 3.2.1.30 group_states_by_same_symbol() | 17 |

| | |
|--|----|
| 3.2.1.31 group_states_by_same_symbol_not_print() | 17 |
| 3.2.1.32 import_automae() | 18 |
| 3.2.1.33 lecture_automate_court() | 18 |
| 3.2.1.34 lecture_automate_long() | 18 |
| 3.2.1.35 lecture_etats_finaux() | 18 |
| 3.2.1.36 lecture_mot() | 19 |
| 3.2.1.37 modify_automate() | 19 |
| 3.2.1.38 mot_saisi() | 19 |
| 3.2.1.39 mot_saisi_avant() | 19 |
| 3.2.1.40 Next_one_state_touch() | 20 |
| 3.2.1.41 Next_states_touch() | 21 |
| 3.2.1.42 Next_states_touch_not_print() | 21 |
| 3.2.1.43 producte_a_b() | 22 |
| 3.2.1.44 reconnu_v2() | 22 |
| 3.2.1.45 remplir_automate() | 22 |
| 3.2.1.46 rendre_complet() | 23 |
| 3.2.1.47 rendre_deterministe() | 23 |
| 3.2.1.48 retourne_etat_couant() | 23 |
| 3.2.1.49 save_automate() | 24 |
| 3.2.1.50 serie_test_reconnaissance() | 24 |
| 3.2.1.51 start_with_tow() | 24 |
| 3.2.1.52 Tab_states_from_q0() | 25 |
| 3.2.1.53 Tab_states_from_q0_not_print() | 25 |
| 3.2.1.54 Tab_transition_automate() | 25 |
| 3.2.1.55 Tab_transition_automate_not_print() | 26 |
| 3.2.1.56 twos_symbol_apply() | 26 |
| 3.2.1.57 union_states_of_same_symbol() | 26 |
| 3.2.1.58 union_states_of_same_symbol_not_print() | 27 |
| 3.3 sources/automate.c File Reference | 27 |
| 3.3.1 Detailed Description | 28 |
| 3.3.2 Function Documentation | 29 |
| 3.3.2.1 add_state_to_composite_table() | 29 |
| 3.3.2.2 add_state_to_rename_table_state() | 29 |
| 3.3.2.3 affichage_1D() | 30 |
| 3.3.2.4 affichage_2D() | 30 |
| 3.3.2.5 affichage_automate() | 30 |
| 3.3.2.6 affichage_case_mat() | 30 |
| 3.3.2.7 affichage_case_tab_1D() | 31 |
| 3.3.2.8 affichage_case_tab_2D() | 31 |
| 3.3.2.9 affichage_col_mat() | 32 |
| 3.3.2.10 affichage_ligne_mat() | 32 |
| 3.3.2.11 affichage_mot() | 32 |

| | |
|--|----|
| 3.3.2.12 allocation_mat_vide() | 32 |
| 3.3.2.13 allocation_tab_1D() | 33 |
| 3.3.2.14 allocation_tab_2D() | 33 |
| 3.3.2.15 calculte_number_states_touch() | 34 |
| 3.3.2.16 complement_automate() | 34 |
| 3.3.2.17 concat() | 34 |
| 3.3.2.18 copie_automate() | 35 |
| 3.3.2.19 copy_line_mat() | 35 |
| 3.3.2.20 del_automate() | 35 |
| 3.3.2.21 echange() | 36 |
| 3.3.2.22 est_complet() | 36 |
| 3.3.2.23 est_deterministe() | 36 |
| 3.3.2.24 est_reconnu() | 36 |
| 3.3.2.25 est_un_etat_final() | 37 |
| 3.3.2.26 exist_state_in_tab_transi() | 37 |
| 3.3.2.27 find_next_state() | 38 |
| 3.3.2.28 generate_automate() | 38 |
| 3.3.2.29 generate_automate_null() | 38 |
| 3.3.2.30 group_states_by_same_symbol() | 38 |
| 3.3.2.31 group_states_by_same_symbol_not_print() | 39 |
| 3.3.2.32 import_automae() | 39 |
| 3.3.2.33 lecture_automate_court() | 40 |
| 3.3.2.34 lecture_automate_long() | 40 |
| 3.3.2.35 lecture_etats_finaux() | 40 |
| 3.3.2.36 lecture_mot() | 40 |
| 3.3.2.37 modify_automate() | 41 |
| 3.3.2.38 mot_saisi() | 41 |
| 3.3.2.39 mot_saisi_avant() | 41 |
| 3.3.2.40 Next_one_state_touch() | 42 |
| 3.3.2.41 Next_states_touch() | 42 |
| 3.3.2.42 Next_states_touch_not_print() | 42 |
| 3.3.2.43 producte_a_b() | 43 |
| 3.3.2.44 reconnu_v2() | 43 |
| 3.3.2.45 remplir_automate() | 44 |
| 3.3.2.46 rendre_complet() | 44 |
| 3.3.2.47 rendre_deterministe() | 44 |
| 3.3.2.48 retourne_etat_couant() | 45 |
| 3.3.2.49 save_automate() | 45 |
| 3.3.2.50 serie_test_reconnaissance() | 45 |
| 3.3.2.51 start_with_tow() | 46 |
| 3.3.2.52 Tab_states_from_q0() | 46 |
| 3.3.2.53 Tab_states_from_q0_not_print() | 46 |

| | |
|--|----|
| 3.3.2.54 Tab_transition_automate() | 47 |
| 3.3.2.55 Tab_transition_automate_not_print() | 47 |
| 3.3.2.56 twos_symbol_apply() | 47 |
| 3.3.2.57 union_states_of_same_symbol() | 48 |
| 3.3.2.58 union_states_of_same_symbol_not_print() | 48 |
| 3.4 Contribution au projet/sources/Automate.h File Reference | 49 |
| 3.5 Projet_C/sources/Automate.h File Reference | 49 |
| 3.5.1 Data Structure Documentation | 50 |
| 3.5.1.1 struct automate | 50 |
| 3.5.1.2 struct table_de_lecture_automate | 51 |
| 3.5.1.3 struct mot | 51 |
| 3.5.2 Typedef Documentation | 51 |
| 3.5.2.1 automate | 51 |
| 3.5.2.2 etats_lus | 51 |
| 3.5.2.3 mot | 51 |
| 3.5.3 Function Documentation | 51 |
| 3.5.3.1 add_state_to_composite_table() | 51 |
| 3.5.3.2 add_state_to_rename_table_state() | 52 |
| 3.5.3.3 affichage_1D() | 52 |
| 3.5.3.4 affichage_2D() | 53 |
| 3.5.3.5 affichage_automate() | 53 |
| 3.5.3.6 affichage_case_mat() | 53 |
| 3.5.3.7 affichage_case_tab_1D() | 53 |
| 3.5.3.8 affichage_case_tab_2D() | 54 |
| 3.5.3.9 affichage_col_mat() | 54 |
| 3.5.3.10 affichage_ligne_mat() | 55 |
| 3.5.3.11 affichage_mot() | 55 |
| 3.5.3.12 allocation_mat_vide() | 55 |
| 3.5.3.13 allocation_tab_1D() | 55 |
| 3.5.3.14 allocation_tab_2D() | 56 |
| 3.5.3.15 calculte_number_states_touch() | 56 |
| 3.5.3.16 complement_automate() | 56 |
| 3.5.3.17 concat() | 57 |
| 3.5.3.18 copie_automate() | 57 |
| 3.5.3.19 copy_line_mat() | 57 |
| 3.5.3.20 del_automate() | 58 |
| 3.5.3.21 echange() | 58 |
| 3.5.3.22 est_complet() | 58 |
| 3.5.3.23 est_deterministe() | 59 |
| 3.5.3.24 est_reconnu() | 59 |
| 3.5.3.25 est_un_etat_final() | 59 |
| 3.5.3.26 exist_state_in_tab_transi() | 60 |

| | |
|--|----|
| 3.5.3.27 find_next_state() | 60 |
| 3.5.3.28 generate_automate() | 60 |
| 3.5.3.29 generate_automate_null() | 61 |
| 3.5.3.30 group_states_by_same_symbol() | 61 |
| 3.5.3.31 group_states_by_same_symbol_not_print() | 61 |
| 3.5.3.32 import_automae() | 62 |
| 3.5.3.33 lecture_automate_court() | 62 |
| 3.5.3.34 lecture_automate_long() | 62 |
| 3.5.3.35 lecture_etats_finaux() | 63 |
| 3.5.3.36 lecture_mot() | 63 |
| 3.5.3.37 modify_automate() | 63 |
| 3.5.3.38 Next_one_state_touch() | 63 |
| 3.5.3.39 Next_states_touch() | 64 |
| 3.5.3.40 Next_states_touch_not_print() | 64 |
| 3.5.3.41 producte_a_b() | 65 |
| 3.5.3.42 reconnu_v2() | 65 |
| 3.5.3.43 remplir_automate() | 65 |
| 3.5.3.44 rendre_complet() | 66 |
| 3.5.3.45 rendre_deterministe() | 66 |
| 3.5.3.46 retourne_etat_couant() | 66 |
| 3.5.3.47 save_automate() | 67 |
| 3.5.3.48 serie_test_reconnaissance() | 67 |
| 3.5.3.49 start_with_tow() | 67 |
| 3.5.3.50 Tab_states_from_q0() | 68 |
| 3.5.3.51 Tab_states_from_q0_not_print() | 68 |
| 3.5.3.52 Tab_transition_automate() | 68 |
| 3.5.3.53 Tab_transition_automate_not_print() | 69 |
| 3.5.3.54 twos_symbol_apply() | 69 |
| 3.5.3.55 union_states_of_same_symbol() | 69 |
| 3.5.3.56 union_states_of_same_symbol_not_print() | 70 |
| 3.6 sources/Automate.h File Reference | 70 |
| 3.6.1 Detailed Description | 72 |
| 3.6.2 Data Structure Documentation | 72 |
| 3.6.2.1 struct automate | 72 |
| 3.6.2.2 struct table_de_lecture_automate | 72 |
| 3.6.2.3 struct mot | 72 |
| 3.6.3 Typedef Documentation | 73 |
| 3.6.3.1 automate | 73 |
| 3.6.3.2 etats_lus | 73 |
| 3.6.3.3 mot | 73 |
| 3.6.4 Function Documentation | 73 |
| 3.6.4.1 add_state_to_composite_table() | 73 |

| | |
|--|----|
| 3.6.4.2 add_state_to_rename_table_state() | 74 |
| 3.6.4.3 affichage_1D() | 74 |
| 3.6.4.4 affichage_2D() | 75 |
| 3.6.4.5 affichage_automate() | 75 |
| 3.6.4.6 affichage_case_mat() | 75 |
| 3.6.4.7 affichage_case_tab_1D() | 76 |
| 3.6.4.8 affichage_case_tab_2D() | 76 |
| 3.6.4.9 affichage_col_mat() | 76 |
| 3.6.4.10 affichage_ligne_mat() | 77 |
| 3.6.4.11 affichage_mot() | 77 |
| 3.6.4.12 allocation_mat_vide() | 77 |
| 3.6.4.13 allocation_tab_1D() | 78 |
| 3.6.4.14 allocation_tab_2D() | 78 |
| 3.6.4.15 calculte_number_states_touch() | 78 |
| 3.6.4.16 complement_automate() | 79 |
| 3.6.4.17 concat() | 79 |
| 3.6.4.18 copie_automate() | 79 |
| 3.6.4.19 copy_line_mat() | 80 |
| 3.6.4.20 del_automate() | 80 |
| 3.6.4.21 echange() | 80 |
| 3.6.4.22 est_complet() | 81 |
| 3.6.4.23 est_deterministe() | 81 |
| 3.6.4.24 est_reconnu() | 81 |
| 3.6.4.25 est_un_etat_final() | 82 |
| 3.6.4.26 exist_state_in_tab_transi() | 82 |
| 3.6.4.27 find_next_state() | 82 |
| 3.6.4.28 generate_automate() | 83 |
| 3.6.4.29 generate_automate_null() | 83 |
| 3.6.4.30 group_states_by_same_symbol() | 83 |
| 3.6.4.31 group_states_by_same_symbol_not_print() | 84 |
| 3.6.4.32 import_automae() | 84 |
| 3.6.4.33 lecture_automate_court() | 84 |
| 3.6.4.34 lecture_automate_long() | 85 |
| 3.6.4.35 lecture_etats_finaux() | 85 |
| 3.6.4.36 lecture_mot() | 85 |
| 3.6.4.37 modify_automate() | 85 |
| 3.6.4.38 Next_one_state_touch() | 86 |
| 3.6.4.39 Next_states_touch() | 86 |
| 3.6.4.40 Next_states_touch_not_print() | 87 |
| 3.6.4.41 producte_a_b() | 87 |
| 3.6.4.42 reconnu_v2() | 87 |
| 3.6.4.43 remplir_automate() | 88 |

| | |
|--|-----------|
| 3.6.4.44 rendre_complet() | 88 |
| 3.6.4.45 rendre_deterministe() | 88 |
| 3.6.4.46 retourne_etat_couant() | 89 |
| 3.6.4.47 save_automate() | 89 |
| 3.6.4.48 serie_test_reconnaissance() | 89 |
| 3.6.4.49 start_with_tow() | 90 |
| 3.6.4.50 Tab_states_from_q0() | 90 |
| 3.6.4.51 Tab_states_from_q0_not_print() | 90 |
| 3.6.4.52 Tab_transition_automate() | 91 |
| 3.6.4.53 Tab_transition_automate_not_print() | 91 |
| 3.6.4.54 twos_symbol_apply() | 91 |
| 3.6.4.55 union_states_of_same_symbol() | 92 |
| 3.6.4.56 union_states_of_same_symbol_not_print() | 92 |
| 3.7 Projet_C/sources/main.c File Reference | 93 |
| 3.7.1 Function Documentation | 93 |
| 3.7.1.1 main() | 93 |
| 3.8 sources/main.c File Reference | 93 |
| 3.8.1 Function Documentation | 93 |
| 3.8.1.1 main() | 94 |
| 3.9 README.md File Reference | 94 |
| Index | 95 |

Chapter 1

README

AEF

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

| | |
|--|----|
| Contribution au projet/sources/ automate.c | 5 |
| Contribution au projet/sources/ Automate.h | 49 |
| Projet_C/sources/ automate.c | 6 |
| Projet_C/sources/ Automate.h | 49 |
| Projet_C/sources/ main.c | 93 |
| sources/ automate.c | 27 |
| sources/ Automate.h | 70 |
| sources/ main.c | 93 |

Chapter 3

File Documentation

3.1 Contribution au projet/sources/automate.c File Reference

```
#include "Automate.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
```

Include dependency graph for automate.c: This graph shows which files directly or indirectly include this file:

Functions

- int * [alloc_tab_1D](#) (int t1)
- int *** [alloc_mat_vide](#) (int d1, int d2)
- [automate generator_automate](#) ()

3.1.1 Function Documentation

3.1.1.1 [alloc_mat_vide\(\)](#)

```
int*** alloc_mat_vide (
    int d1,
    int d2 )
```

3.1.1.2 [alloc_tab_1D\(\)](#)

```
int* alloc_tab_1D (
    int t1 )
```

Returns

3.1.1.3 generator_automate()

```
automate generator_automate ( )
```

Returns

3.2 Projet_C/sources/automate.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "Automate.h"
#include "main.c"
```

Include dependency graph for automate.c: This graph shows which files directly or indirectly include this file:

Functions

- void [affichage_1D](#) (int *tab, int dim_tab_1D)
- void [affichage_case_tab_1D](#) (int *tab, int position, int dim_tab_1D)
- void [affichage_2D](#) (int **tab, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_case_tab_2D](#) (int **tab, int position_line, int position_col, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_ligne_mat](#) (int ***tab, int position_line, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_col_mat](#) (int ***tab, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_case_mat](#) (int ***tab, int position_line, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_automate](#) (automate a)
- void [lecture_etats_finaux](#) (automate a)
- void [echange](#) (int *a, int *b)
- void [affichage_mot](#) (mot *mot)
- void [lecture_mot](#) (mot m)
- int [est_deterministe](#) (automate a)
- int [est_complet](#) (automate a)
- int [est_un_etat_final](#) (automate a, int etat)
- int [exist_state_in_tab_transi](#) (automate a, int state_touch, int ***tab1)
- int [Next_one_state_touch](#) (automate a, int state_start, int symbol_apply)
- int [calculte_number_states_touch](#) (automate a, int *tab_states_touch)
- int [start_with_tow](#) (int etat_initial_a, int etat_initial_b)
- int [retourne_etat_couant](#) (automate a, int curent_state, int curent_symbol)
- int [est_reconnu](#) (mot m, automate a)
- int [serie_test_reconnaissance](#) (automate a, int n_fois)
- int [reconnu_v2](#) (mot m, automate a)
- int [find_next_state](#) (automate a, mot m)
- int * [allocation_tab_1D](#) (int t1)
- int * [Next_states_touch](#) (automate a, int start_state, int symbol_apply)
- int * [Next_states_touch_not_print](#) (automate a, int start_state, int symbol_apply)
- int * [Tab_states_from_q0](#) (automate a, int symbol_apply)
- int * [Tab_states_from_q0_not_print](#) (automate a, int symbol_apply)
- int * [group_states_by_same_symbol](#) (automate a, int state_apply, int symbol_fix)

- `int * group_states_by_same_symbol_not_print (automate a, int state_apply, int symbol_fix)`
- `int * union_states_of_same_symbol (automate a, int state_apply, int symbol_fix)`
- `int * union_states_of_same_symbol_not_print (automate a, int state_apply, int symbol_fix)`
- `int ** add_state_to_composite_table (automate a, int **tab, int composite_state, int position)`
- `int ** copy_line_mat (automate a, int ***mat, int position_line)`
- `int ** allocation_tab_2D (int t2, int t3)`
- `int *** allocation_mat_vide (int d1, int d2)`
- `int *** lecture_automate_court (automate a)`
- `int *** add_state_to_rename_table_state (automate a, int ***tab_rename, int new_state, int position)`
- `int *** Tab_transition_automate (automate a)`
- `int *** Tab_transition_automate_not_print (automate a)`
- `int *** twos_symbol_apply (automate a, int state_1, int state_2, int symbol_1, int symbol_2)`
- `automate generate_automate_null (int d1, int d2, int d3)`
- `automate generate_automate ()`
- `automate remplir_automate (automate m, int val)`
- `automate import_automae (char path)`
- `automate modify_automate (automate a)`
- `automate save_automate (automate a)`
- `automate del_automate (automate a)`
- `automate copie_automate (automate a)`
- `automate rendre_complet (automate a)`
- `automate concat (automate a, automate b)`
- `automate complement_automate (automate a)`
- `automate rendre_deterministe (automate a)`
- `automate produit_a_b (automate a, automate b)`
- `etats_lus lecture_automate_long (automate a)`
- `mot mot_saisi_avant (automate a)`
- `mot mot_saisi (automate a)`

3.2.1 Function Documentation

3.2.1.1 add_state_to_composite_table()

```
int** add_state_to_composite_table (
    automate a,
    int ** tab,
    int composite_state,
    int position )
```

Parameters

| | |
|------------------------|--|
| <i>a</i> | |
| <i>tab</i> | |
| <i>composite_state</i> | |
| <i>position</i> | |

Returns

int**

3.2.1.2 add_state_to_rename_table_state()

```
int*** add_state_to_rename_table_state (
    automate a,
    int *** tan_rename,
    int new_state,
    int position )
```

Parameters

| | |
|-------------------|--|
| <i>a</i> | |
| <i>tan_rename</i> | |
| <i>new_state</i> | |
| <i>position</i> | |

Returns

int***

3.2.1.3 affichage_1D()

```
void affichage_1D (
    int * tab,
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>dim_tab_1D</i> | |

3.2.1.4 affichage_2D()

```
void affichage_2D (
    int ** tab,
    int dim_tab_2D_ligne,
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.2.1.5 affichage_automate()

```
void affichage_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.2.1.6 affichage_case_mat()

```
void affichage_case_mat (  
    int *** tab,  
    int position_line,  
    int position_col,  
    int dim_tab_3D_ligne,  
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.2.1.7 affichage_case_tab_1D()

```
void affichage_case_tab_1D (  
    int * tab,  
    int position,  
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>position</i> | |
| <i>dim_tab_1D</i> | |

3.2.1.8 affichage_case_tab_2D()

```
void affichage_case_tab_2D (
    int ** tab,
    int position_line,
    int position_col,
    int dim_tab_2D_ligne,
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.2.1.9 affichage_col_mat()

```
void affichage_col_mat (
    int *** tab,
    int position_col,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.2.1.10 affichage_ligne_mat()

```
void affichage_ligne_mat (
    int *** tab,
    int position_line,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|------------|--|
| <i>tab</i> | |
|------------|--|

Parameters

| | |
|-------------------------|--|
| <i>position_line</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.2.1.11 affichage_mot()

```
void affichage_mot (
    mot * mot )
```

Parameters

| | |
|------------|--|
| <i>mot</i> | |
|------------|--|

3.2.1.12 allocation_mat_vide()

```
int*** allocation_mat_vide (
    int d1,
    int d2 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |

Returns

int***

3.2.1.13 allocation_tab_1D()

```
int* allocation_tab_1D (
    int t1 )
```

Parameters

| | |
|-----------|--|
| <i>t1</i> | |
|-----------|--|

Returns

int*

3.2.1.14 allocation_tab_2D()

```
int** allocation_tab_2D (
    int t2,
    int t3 )
```

Parameters

| | |
|-----------|--|
| <i>t2</i> | |
| <i>t3</i> | |

Returns

int**

3.2.1.15 calculte_number_states_touch()

```
int calculte_number_states_touch (
    automate a,
    int * tab_states_touch )
```

Parameters

| | |
|-------------------------|--|
| <i>a</i> | |
| <i>tab_states_touch</i> | |

Returns

int

3.2.1.16 complement_automate()

```
automate complement_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.17 concat()

```
automate concat (  
    automate a,  
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.2.1.18 copie_automate()

```
automate copie_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.19 copy_line_mat()

```
int** copy_line_mat (  
    automate a,  
    int *** mat,  
    int position_line )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>mat</i> | |
| <i>position_line</i> | |

Returns

int**

3.2.1.20 del_automate()

```
automate del_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.21 echange()

```
void echange (  
    int * a,  
    int * b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

3.2.1.22 est_complet()

```
int est_complet (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.2.1.23 est_deterministe()

```
int est_deterministe (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.2.1.24 est_reconnu()

```
int est_reconnu (
    mot m,
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.2.1.25 est_un_etat_final()

```
int est_un_etat_final (
    automate a,
    int etat )
```

Parameters

| | |
|-------------|--|
| <i>a</i> | |
| <i>etat</i> | |

Returns

int

3.2.1.26 exist_state_in_tab_transi()

```
int exist_state_in_tab_transi (
    automate a,
    int state_touch,
    int *** tab1 )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_touch</i> | |
| <i>tab1</i> | |

Returns

int

3.2.1.27 find_next_state()

```
int find_next_state (
    automate a,
    mot m )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>m</i> | |

Returns

int

3.2.1.28 generate_automate()

```
automate generate_automate ( )
```

Returns

automate

3.2.1.29 generate_automate_null()

```
automate generate_automate_null (
    int d1,
    int d2,
    int d3 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |
| <i>d3</i> | |

Returns

automate

3.2.1.30 group_states_by_same_symbol()

```
int* group_states_by_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.2.1.31 group_states_by_same_symbol_not_print()

```
int* group_states_by_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.2.1.32 import_automae()

```
automate import_automae (  
    char path )
```

Parameters

| | |
|-------------|--|
| <i>path</i> | |
|-------------|--|

Returns

automate

3.2.1.33 lecture_automate_court()

```
int*** lecture_automate_court (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.2.1.34 lecture_automate_long()

```
etats_lus lecture_automate_long (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

etats_lus

3.2.1.35 lecture_etats_finaux()

```
void lecture_etats_finaux (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.2.1.36 lecture_mot()

```
void lecture_mot (
    mot m )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
|----------|--|

3.2.1.37 modify_automate()

```
automate modify_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.38 mot_saisi()

```
mot mot_saisi (
    automate a )
```

3.2.1.39 mot_saisi_avant()

```
mot mot_saisi_avant (
    automate a )
```

3.2.1.40 Next_one_state_touch()

```
int Next_one_state_touch (
    automate a,
    int state_start,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>state_start</i> | |
| <i>symbol_apply</i> | |

Returns

int

3.2.1.41 Next_states_touch()

```
int* Next_states_touch (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.2.1.42 Next_states_touch_not_print()

```
int* Next_states_touch_not_print (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.2.1.43 producte_a_b()

```
automate producte_a_b (  
    automate a,  
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.2.1.44 reconnu_v2()

```
int reconnu_v2 (  
    mot m,  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.2.1.45 remplir_automate()

```
automate remplir_automate (  
    automate m,  
    int val )
```

Parameters

| | |
|------------|--|
| <i>m</i> | |
| <i>val</i> | |

Returns

automate

3.2.1.46 rendre_complet()

```
automate rendre_complet (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.47 rendre_deterministe()

```
automate rendre_deterministe (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.48 retourne_etat_couant()

```
int retourne_etat_couant (  
    automate a,  
    int curent_state,  
    int curent_symbol )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>curent_state</i> | |
| <i>curent_symbol</i> | |

Returns

int

3.2.1.49 save_automate()

```
automate save_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.2.1.50 serie_test_reconnaissance()

```
int serie_test_reconnaissance (
    automate a,
    int n_fois )
```

Parameters

| | |
|---------------|--|
| <i>a</i> | |
| <i>n_fois</i> | |

Returns

int

3.2.1.51 start_with_tow()

```
int start_with_tow (
    int etat_initial_a,
    int etat_initial_b )
```

Parameters

| | |
|---|--|
| <i>etat_initial</i> _{<i>a</i>} | |
| <i>etat_initial</i> _{<i>b</i>} | |

Returns

int

3.2.1.52 Tab_states_from_q0()

```
int* Tab_states_from_q0 (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.2.1.53 Tab_states_from_q0_not_print()

```
int* Tab_states_from_q0_not_print (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.2.1.54 Tab_transition_automate()

```
int*** Tab_transition_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.2.1.55 Tab_transition_automate_not_print()

```
int** Tab_transition_automate_not_print (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int**

3.2.1.56 twos_symbol_apply()

```
int** twos_symbol_apply (
    automate a,
    int state_1,
    int state_2,
    int symbol_1,
    int symbol_2 )
```

Parameters

| | |
|---------------------------------------|--|
| <i>a</i> | |
| <i>state_1</i> | |
| <i>state_2</i> | |
| <i>symbol_↔₁</i> | |
| <i>symbol_↔₂</i> | |

Returns

int**

3.2.1.57 union_states_of_same_symbol()

```
int* union_states_of_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.2.1.58 union_states_of_same_symbol_not_print()

```
int* union_states_of_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.3 sources/automate.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "Automate.h"
#include "main.c"
```

Include dependency graph for automate.c: This graph shows which files directly or indirectly include this file:

Functions

- void [affichage_1D](#) (int *tab, int dim_tab_1D)
- void [affichage_case_tab_1D](#) (int *tab, int position, int dim_tab_1D)
- void [affichage_2D](#) (int **tab, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_case_tab_2D](#) (int **tab, int position_line, int position_col, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_ligne_mat](#) (int ***tab, int position_line, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_col_mat](#) (int ***tab, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_case_mat](#) (int ***tab, int position_line, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_automate](#) (automate a)
- void [lecture_etats_finaux](#) (automate a)
- void [echange](#) (int *a, int *b)
- void [affichage_mot](#) (mot *mot)
- void [lecture_mot](#) (mot m)
- int [est_deterministe](#) (automate a)

- `int est_complet (automate a)`
- `int est_un_etat_final (automate a, int etat)`
- `int exist_state_in_tab_transi (automate a, int state_touch, int ***tab1)`
- `int Next_one_state_touch (automate a, int state_start, int symbol_apply)`
- `int calculte_number_states_touch (automate a, int *tab_states_touch)`
- `int start_with_tow (int etat_initial_a, int etat_initial_b)`
- `int retourne_etat_couant (automate a, int curent_state, int curent_symbol)`
- `int est_reconnu (mot m, automate a)`
- `int serie_test_reconnaissance (automate a, int n_fois)`
- `int reconnu_v2 (mot m, automate a)`
- `int find_next_state (automate a, mot m)`
- `int * allocation_tab_1D (int t1)`
- `int * Next_states_touch (automate a, int start_state, int symbol_apply)`
- `int * Next_states_touch_not_print (automate a, int start_state, int symbol_apply)`
- `int * Tab_states_from_q0 (automate a, int symbol_apply)`
- `int * Tab_states_from_q0_not_print (automate a, int symbol_apply)`
- `int * group_states_by_same_symbol (automate a, int state_apply, int symbol_fix)`
- `int * group_states_by_same_symbol_not_print (automate a, int state_apply, int symbol_fix)`
- `int * union_states_of_same_symbol (automate a, int state_apply, int symbol_fix)`
- `int * union_states_of_same_symbol_not_print (automate a, int state_apply, int symbol_fix)`
- `int ** add_state_to_composite_table (automate a, int **tab, int composite_state, int position)`
- `int ** copy_line_mat (automate a, int ***mat, int position_line)`
- `int ** allocation_tab_2D (int t2, int t3)`
- `int *** allocation_mat_vide (int d1, int d2)`
- `int *** lecture_automate_court (automate a)`
- `int *** add_state_to_rename_table_state (automate a, int ***tab_rename, int new_state, int position)`
- `int *** Tab_transition_automate (automate a)`
- `int *** Tab_transition_automate_not_print (automate a)`
- `int *** twos_symbol_apply (automate a, int state_1, int state_2, int symbol_1, int symbol_2)`
- `automate generate_automate_null (int d1, int d2, int d3)`
- `automate generate_automate ()`
- `automate remplir_automate (automate m, int val)`
- `automate import_automae (char path)`
- `automate modify_automate (automate a)`
- `automate save_automate (automate a)`
- `automate del_automate (automate a)`
- `automate copie_automate (automate a)`
- `automate rendre_complet (automate a)`
- `automate concat (automate a, automate b)`
- `automate complement_automate (automate a)`
- `automate rendre_deterministe (automate a)`
- `automate producte_a_b (automate a, automate b)`
- `etats_lus lecture_automate_long (automate a)`
- `mot mot_saisi_avant (automate a)`
- `mot mot_saisi (automate a)`

3.3.1 Detailed Description

Author

Charly romy TANGA - Abdoulaye MBAYE - Berkehan USLU (you@domain.com)

Version

0.1

Date

2024-01-11

Copyright

Copyright (c) 2024

3.3.2 Function Documentation

3.3.2.1 add_state_to_composite_table()

```
int** add_state_to_composite_table (
    automate a,
    int ** tab,
    int composite_state,
    int position )
```

Parameters

| | |
|------------------------|--|
| <i>a</i> | |
| <i>tab</i> | |
| <i>composite_state</i> | |
| <i>position</i> | |

Returns

int**

3.3.2.2 add_state_to_rename_table_state()

```
int*** add_state_to_rename_table_state (
    automate a,
    int *** tab_rename,
    int new_state,
    int position )
```

Parameters

| | |
|-------------------|--|
| <i>a</i> | |
| <i>tab_rename</i> | |
| <i>new_state</i> | |
| <i>position</i> | |

Returns

int***

3.3.2.3 affichage_1D()

```
void affichage_1D (
    int * tab,
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>dim_tab_1D</i> | |

3.3.2.4 affichage_2D()

```
void affichage_2D (
    int ** tab,
    int dim_tab_2D_ligne,
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.3.2.5 affichage_automate()

```
void affichage_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.3.2.6 affichage_case_mat()

```
void affichage_case_mat (
```



```
int *** tab,  
int position_line,  
int position_col,  
int dim_tab_3D_ligne,  
int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.3.2.7 affichage_case_tab_1D()

```
void affichage_case_tab_1D (  
    int * tab,  
    int position,  
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>position</i> | |
| <i>dim_tab_1D</i> | |

3.3.2.8 affichage_case_tab_2D()

```
void affichage_case_tab_2D (  
    int ** tab,  
    int position_line,  
    int position_col,  
    int dim_tab_2D_ligne,  
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.3.2.9 affichage_col_mat()

```
void affichage_col_mat (
    int *** tab,
    int position_col,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.3.2.10 affichage_ligne_mat()

```
void affichage_ligne_mat (
    int *** tab,
    int position_line,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.3.2.11 affichage_mot()

```
void affichage_mot (
    mot * mot )
```

Parameters

| | |
|------------|--|
| <i>mot</i> | |
|------------|--|

3.3.2.12 allocation_mat_vide()

```
int*** allocation_mat_vide (
```

```
int d1,  
int d2 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |

Returns

int**

3.3.2.13 allocation_tab_1D()

```
int* allocation_tab_1D (  
    int t1 )
```

Parameters

| | |
|-----------|--|
| <i>t1</i> | |
|-----------|--|

Returns

int*

3.3.2.14 allocation_tab_2D()

```
int** allocation_tab_2D (  
    int t2,  
    int t3 )
```

Parameters

| | |
|-----------|--|
| <i>t2</i> | |
| <i>t3</i> | |

Returns

int**

3.3.2.15 `calculte_number_states_touch()`

```
int calculte_number_states_touch (
    automate a,
    int * tab_states_touch )
```

Parameters

| | |
|-------------------------|--|
| <i>a</i> | |
| <i>tab_states_touch</i> | |

Returns

int

3.3.2.16 `complement_automate()`

```
automate complement_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.17 `concat()`

```
automate concat (
    automate a,
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.3.2.18 copie_automate()

```
automate copie_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.19 copy_line_mat()

```
int** copy_line_mat (  
    automate a,  
    int *** mat,  
    int position_line )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>mat</i> | |
| <i>position_line</i> | |

Returns

int**

3.3.2.20 del_automate()

```
automate del_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.21 exchange()

```
void exchange (
    int * a,
    int * b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

3.3.2.22 est_complet()

```
int est_complet (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.3.2.23 est_deterministe()

```
int est_deterministe (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.3.2.24 est_reconnu()

```
int est_reconnu (
    mot m,
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.3.2.25 est_un_etat_final()

```
int est_un_etat_final (
    automate a,
    int etat )
```

Parameters

| | |
|-------------|--|
| <i>a</i> | |
| <i>etat</i> | |

Returns

int

3.3.2.26 exist_state_in_tab_transi()

```
int exist_state_in_tab_transi (
    automate a,
    int state_touch,
    int *** tab1 )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_touch</i> | |
| <i>tab1</i> | |

Returns

int

3.3.2.27 find_next_state()

```
int find_next_state (
    automate a,
    mot m )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>m</i> | |

Returns

int

3.3.2.28 generate_automate()

```
automate generate_automate ( )
```

Returns

automate

3.3.2.29 generate_automate_null()

```
automate generate_automate_null (
    int d1,
    int d2,
    int d3 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |
| <i>d3</i> | |

Returns

automate

3.3.2.30 group_states_by_same_symbol()

```
int* group_states_by_same_symbol (
    automate a,
```



```
int state_apply,  
int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.3.2.31 group_states_by_same_symbol_not_print()

```
int* group_states_by_same_symbol_not_print (  
    automate a,  
    int state_apply,  
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.3.2.32 import_automae()

```
automate import_automae (  
    char path )
```

Parameters

| | |
|-------------|--|
| <i>path</i> | |
|-------------|--|

Returns

automate

3.3.2.33 lecture_automate_court()

```
int** lecture_automate_court (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int**

3.3.2.34 lecture_automate_long()

```
etats_lus lecture_automate_long (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

etats_lus

3.3.2.35 lecture_etats_finaux()

```
void lecture_etats_finaux (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.3.2.36 lecture_mot()

```
void lecture_mot (
    mot m )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
|----------|--|

3.3.2.37 modify_automate()

```
automate modify_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.38 mot_saisi()

```
mot mot_saisi (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

mot

3.3.2.39 mot_saisi_avant()

```
mot mot_saisi_avant (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

mot

3.3.2.40 Next_one_state_touch()

```
int Next_one_state_touch (
    automate a,
    int state_start,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>state_start</i> | |
| <i>symbol_apply</i> | |

Returns

int

3.3.2.41 Next_states_touch()

```
int* Next_states_touch (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.3.2.42 Next_states_touch_not_print()

```
int* Next_states_touch_not_print (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.3.2.43 producte_a_b()

```
automate producte_a_b (  
    automate a,  
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.3.2.44 reconnu_v2()

```
int reconnu_v2 (  
    mot m,  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.3.2.45 remplir_automate()

```
automate remplir_automate (  
    automate m,  
    int val )
```

Parameters

| | |
|------------|--|
| <i>m</i> | |
| <i>val</i> | |

Returns

automate

3.3.2.46 rendre_complet()

```
automate rendre_complet (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.47 rendre_deterministe()

```
automate rendre_deterministe (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.48 retourne_etat_couant()

```
int retourne_etat_couant (
    automate a,
    int curent_state,
    int curent_symbol )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>curent_state</i> | |
| <i>curent_symbol</i> | |

Returns

int

3.3.2.49 save_automate()

```
automate save_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.3.2.50 serie_test_reconnaissance()

```
int serie_test_reconnaissance (
    automate a,
    int n_fois )
```

Parameters

| | |
|---------------|--|
| <i>a</i> | |
| <i>n_fois</i> | |

Returns

int

3.3.2.51 start_with_tow()

```
int start_with_tow (
    int etat_initial_a,
    int etat_initial_b )
```

Parameters

| | |
|-----------------------------|--|
| <i>etat_initial</i> ↔ _a | |
| <i>etat_initial</i> ↔ _b | |

Returns

int

3.3.2.52 Tab_states_from_q0()

```
int* Tab_states_from_q0 (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.3.2.53 Tab_states_from_q0_not_print()

```
int* Tab_states_from_q0_not_print (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.3.2.54 Tab_transition_automate()

```
int*** Tab_transition_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.3.2.55 Tab_transition_automate_not_print()

```
int*** Tab_transition_automate_not_print (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.3.2.56 twos_symbol_apply()

```
int*** twos_symbol_apply (  
    automate a,  
    int state_1,  
    int state_2,  
    int symbol_1,  
    int symbol_2 )
```

Parameters

| | |
|----------------|--|
| <i>a</i> | |
| <i>state_1</i> | |

Parameters

| | |
|---|--|
| <i>state_2</i> | |
| <i>symbol</i> _{<i>k</i>↔} <i>_1</i> | |
| <i>symbol</i> _{<i>k</i>↔} <i>_2</i> | |

Returns

int***

3.3.2.57 union_states_of_same_symbol()

```
int* union_states_of_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.3.2.58 union_states_of_same_symbol_not_print()

```
int* union_states_of_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.4 Contribution au projet/sources/Automate.h File Reference

```
#include "automate.c"
```

```
#include "main.c"
```

Include dependency graph for Automate.h:

3.5 Projet_C/sources/Automate.h File Reference

```
#include "automate.c"
```

```
#include "main.c"
```

Include dependency graph for Automate.h: This graph shows which files directly or indirectly include this file:

Data Structures

- struct [automate](#)
- struct [table_de_lecture_automate](#)
- struct [mot](#)

Typedefs

- typedef struct [automate](#) [automate](#)
- typedef struct [table_de_lecture_automate](#) [etats_lus](#)
- typedef struct [mot](#) [mot](#)

Functions

- void [affichage_1D](#) (int *tab, int dim_tab_1D)
- void [affichage_case_tab_1D](#) (int *tab, int position, int dim_tab_1D)
- void [affichage_2D](#) (int **tab, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_case_tab_2D](#) (int **tab, int position_line, int position_col, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_ligne_mat](#) (int ***tab, int position_line, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_col_mat](#) (int ***tab, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_case_mat](#) (int ***tab, int position_line, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_automate](#) ([automate](#) a)
- void [lecture_etats_finaux](#) ([automate](#) a)
- void [echange](#) (int *a, int *b)
- void [affichage_mot](#) ([mot](#) *mot)
- void [lecture_mot](#) ([mot](#) m)
- int [est_deterministe](#) ([automate](#) a)
- int [est_complet](#) ([automate](#) a)
- int [est_un_etat_final](#) ([automate](#) a, int etat)
- int [exist_state_in_tab_transi](#) ([automate](#) a, int state_touch, int ***tab1)

- int [Next_one_state_touch](#) (automate a, int state_start, int symbol_apply)
- int [calculte_number_states_touch](#) (automate a, int *tab_states_touch)
- int [start_with_tow](#) (int etat_initial_a, int etat_initial_b)
- int [retourne_etat_couant](#) (automate a, int curent_state, int curent_symbol)
- int [est_reconnu](#) (mot m, automate a)
- int [serie_test_reconnaissance](#) (automate a, int n_fois)
- int [reconnu_v2](#) (mot m, automate a)
- int [find_next_state](#) (automate a, mot m)
- int * [allocation_tab_1D](#) (int t1)
- int * [Next_states_touch](#) (automate a, int start_state, int symbol_apply)
- int * [Next_states_touch_not_print](#) (automate a, int start_state, int symbol_apply)
- int * [Tab_states_from_q0](#) (automate a, int symbol_apply)
- int * [Tab_states_from_q0_not_print](#) (automate a, int symbol_apply)
- int * [group_states_by_same_symbol](#) (automate a, int state_apply, int symbol_fix)
- int * [group_states_by_same_symbol_not_print](#) (automate a, int state_apply, int symbol_fix)
- int * [union_states_of_same_symbol](#) (automate a, int state_apply, int symbol_fix)
- int * [union_states_of_same_symbol_not_print](#) (automate a, int state_apply, int symbol_fix)
- int ** [add_state_to_composite_table](#) (automate a, int **tab, int composite_state, int position)
- int ** [copy_line_mat](#) (automate a, int ***mat, int position_line)
- int ** [allocation_tab_2D](#) (int t2, int t3)
- int *** [allocation_mat_vide](#) (int d1, int d2)
- int *** [lecture_automate_court](#) (automate a)
- int *** [add_state_to_rename_table_state](#) (automate a, int ***tan_rename, int new_state, int position)
- int *** [Tab_transition_automate](#) (automate a)
- int *** [Tab_transition_automate_not_print](#) (automate a)
- int *** [twos_symbol_apply](#) (automate a, int state_1, int state_2, int symbol_1, int symbol_2)
- [automate generate_automate_null](#) (int d1, int d2, int d3)
- [automate generate_automate](#) ()
- [automate remplir_automate](#) (automate m, int val)
- [automate import_automae](#) (char path)
- [automate modify_automate](#) (automate a)
- [automate save_automate](#) (automate a)
- [automate del_automate](#) (automate a)
- [automate copie_automate](#) (automate a)
- [automate rendre_complet](#) (automate a)
- [automate concat](#) (automate a, automate b)
- [automate complement_automate](#) (automate a)
- [automate rendre_deterministe](#) (automate a)
- [automate produit_a_b](#) (automate a, automate b)
- [etats_lus_lecture_automate_long](#) (automate a)

3.5.1 Data Structure Documentation

3.5.1.1 struct automate

Data Fields

| | | |
|---------|------------------|--|
| int | etat_initial | |
| int * | Etats_finaux | |
| int *** | matrice | |
| int | nb_Etats | |
| int | nb_Etats_finaux | |
| int | nb_Symboles | |
| int *** | Table_transition | |

3.5.1.2 struct table_de_lecture_automate

Data Fields

| | | |
|---------|--------------------------|--|
| int *** | etat_lu_depart | |
| int *** | etats_lus_finaux | |
| int *** | etats_lus_intermediaires | |

3.5.1.3 struct mot

Data Fields

| | | |
|-------|------------|--|
| int | size_mot | |
| int * | Tab_caract | |

3.5.2 Typedef Documentation

3.5.2.1 automate

```
typedef struct automate automate
```

3.5.2.2 etats_lus

```
typedef struct table_de_lecture_automate etats_lus
```

3.5.2.3 mot

```
typedef struct mot mot
```

3.5.3 Function Documentation

3.5.3.1 add_state_to_composite_table()

```
int** add_state_to_composite_table (  
    automate a,  
    int ** tab,  
    int composite_state,  
    int position )
```

Parameters

| | |
|------------------------|--|
| <i>a</i> | |
| <i>tab</i> | |
| <i>composite_state</i> | |
| <i>position</i> | |

Returns

int**

3.5.3.2 add_state_to_rename_table_state()

```
int*** add_state_to_rename_table_state (
    automate a,
    int *** tab_rename,
    int new_state,
    int position )
```

Parameters

| | |
|-------------------|--|
| <i>a</i> | |
| <i>tab_rename</i> | |
| <i>new_state</i> | |
| <i>position</i> | |

Returns

int***

3.5.3.3 affichage_1D()

```
void affichage_1D (
    int * tab,
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>dim_tab_1D</i> | |

3.5.3.4 affichage_2D()

```
void affichage_2D (
    int ** tab,
    int dim_tab_2D_ligne,
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.5.3.5 affichage_automate()

```
void affichage_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.5.3.6 affichage_case_mat()

```
void affichage_case_mat (
    int *** tab,
    int position_line,
    int position_col,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.5.3.7 affichage_case_tab_1D()

```
void affichage_case_tab_1D (
    int * tab,
```

```
int position,  
int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>position</i> | |
| <i>dim_tab_1D</i> | |

3.5.3.8 affichage_case_tab_2D()

```
void affichage_case_tab_2D (  
    int ** tab,  
    int position_line,  
    int position_col,  
    int dim_tab_2D_ligne,  
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.5.3.9 affichage_col_mat()

```
void affichage_col_mat (  
    int *** tab,  
    int position_col,  
    int dim_tab_3D_ligne,  
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.5.3.10 affichage_ligne_mat()

```
void affichage_ligne_mat (
    int *** tab,
    int position_line,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.5.3.11 affichage_mot()

```
void affichage_mot (
    mot * mot )
```

Parameters

| | |
|------------|--|
| <i>mot</i> | |
|------------|--|

3.5.3.12 allocation_mat_vide()

```
int*** allocation_mat_vide (
    int d1,
    int d2 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |

Returns

int***

3.5.3.13 allocation_tab_1D()

```
int* allocation_tab_1D (
    int t1 )
```

Parameters

| | |
|-----------|--|
| <i>t1</i> | |
|-----------|--|

Returns

int*

3.5.3.14 allocation_tab_2D()

```
int** allocation_tab_2D (
    int t2,
    int t3 )
```

Parameters

| | |
|-----------|--|
| <i>t2</i> | |
| <i>t3</i> | |

Returns

int**

3.5.3.15 calculte_number_states_touch()

```
int calculte_number_states_touch (
    automate a,
    int * tab_states_touch )
```

Parameters

| | |
|-------------------------|--|
| <i>a</i> | |
| <i>tab_states_touch</i> | |

Returns

int

3.5.3.16 complement_automate()

```
automate complement_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.17 concat()

```
automate concat (  
    automate a,  
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.5.3.18 copie_automate()

```
automate copie_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.19 copy_line_mat()

```
int** copy_line_mat (  
    automate a,  
    int *** mat,  
    int position_line )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>mat</i> | |
| <i>position_line</i> | |

Returns

int**

3.5.3.20 del_automate()

```
automate del_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.21 echange()

```
void echange (  
    int * a,  
    int * b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

3.5.3.22 est_complet()

```
int est_complet (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.5.3.23 est_deterministe()

```
int est_deterministe (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.5.3.24 est_reconnu()

```
int est_reconnu (  
    mot m,  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.5.3.25 est_un_etat_final()

```
int est_un_etat_final (  
    automate a,  
    int etat )
```

Parameters

| | |
|-------------|--|
| <i>a</i> | |
| <i>etat</i> | |

Returns

int

3.5.3.26 exist_state_in_tab_transi()

```
int exist_state_in_tab_transi (
    automate a,
    int state_touch,
    int *** tab1 )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_touch</i> | |
| <i>tab1</i> | |

Returns

int

3.5.3.27 find_next_state()

```
int find_next_state (
    automate a,
    mot m )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>m</i> | |

Returns

int

3.5.3.28 generate_automate()

```
automate generate_automate ( )
```

Returns

automate

3.5.3.29 generate_automate_null()

```
automate generate_automate_null (
    int d1,
    int d2,
    int d3 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |
| <i>d3</i> | |

Returns

automate

3.5.3.30 group_states_by_same_symbol()

```
int* group_states_by_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.5.3.31 group_states_by_same_symbol_not_print()

```
int* group_states_by_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.5.3.32 import_automae()

```
automate import_automae (
    char path )
```

Parameters

| | |
|-------------|--|
| <i>path</i> | |
|-------------|--|

Returns

automate

3.5.3.33 lecture_automate_court()

```
int*** lecture_automate_court (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.5.3.34 lecture_automate_long()

```
etats_lus lecture_automate_long (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

etats_lus

3.5.3.35 lecture_etats_finaux()

```
void lecture_etats_finaux (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.5.3.36 lecture_mot()

```
void lecture_mot (  
    mot m )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
|----------|--|

3.5.3.37 modify_automate()

```
automate modify_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.38 Next_one_state_touch()

```
int Next_one_state_touch (  
    automate a,  
    int state_start,  
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>state_start</i> | |
| <i>symbol_apply</i> | |

Returns

int

3.5.3.39 Next_states_touch()

```
int* Next_states_touch (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.5.3.40 Next_states_touch_not_print()

```
int* Next_states_touch_not_print (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.5.3.41 producte_a_b()

```
automate producte_a_b (  
    automate a,  
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.5.3.42 reconnu_v2()

```
int reconnu_v2 (  
    mot m,  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.5.3.43 remplir_automate()

```
automate remplir_automate (  
    automate m,  
    int val )
```

Parameters

| | |
|------------|--|
| <i>m</i> | |
| <i>val</i> | |

Returns

automate

3.5.3.44 rendre_complet()

```
automate rendre_complet (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.45 rendre_deterministe()

```
automate rendre_deterministe (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.46 retourne_etat_couant()

```
int retourne_etat_couant (  
    automate a,  
    int curent_state,  
    int curent_symbol )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>curent_state</i> | |
| <i>curent_symbol</i> | |

Returns

int

3.5.3.47 save_automate()

```
automate save_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.5.3.48 serie_test_reconnaissance()

```
int serie_test_reconnaissance (  
    automate a,  
    int n_fois )
```

Parameters

| | |
|---------------|--|
| <i>a</i> | |
| <i>n_fois</i> | |

Returns

int

3.5.3.49 start_with_tow()

```
int start_with_tow (  
    int etat_initial_a,  
    int etat_initial_b )
```

Parameters

| | |
|---|--|
| <i>etat_initial</i> _{<i>a</i>} | |
| <i>etat_initial</i> _{<i>b</i>} | |

Returns

int

3.5.3.50 Tab_states_from_q0()

```
int* Tab_states_from_q0 (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.5.3.51 Tab_states_from_q0_not_print()

```
int* Tab_states_from_q0_not_print (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.5.3.52 Tab_transition_automate()

```
int*** Tab_transition_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.5.3.53 Tab_transition_automate_not_print()

```
int*** Tab_transition_automate_not_print (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.5.3.54 twos_symbol_apply()

```
int*** twos_symbol_apply (
    automate a,
    int state_1,
    int state_2,
    int symbol_1,
    int symbol_2 )
```

Parameters

| | |
|---|--|
| <i>a</i> | |
| <i>state_1</i> | |
| <i>state_2</i> | |
| <i>symbol</i> _↔ <i>_1</i> | |
| <i>symbol</i> _↔ <i>_2</i> | |

Returns

int***

3.5.3.55 union_states_of_same_symbol()

```
int* union_states_of_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.5.3.56 union_states_of_same_symbol_not_print()

```
int* union_states_of_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.6 sources/Automate.h File Reference

```
#include "automate.c"
#include "main.c"
```

Include dependency graph for Automate.h: This graph shows which files directly or indirectly include this file:

Data Structures

- struct [automate](#)
- struct [table_de_lecture_automate](#)
- struct [mot](#)

Typedefs

- typedef struct [automate](#) [automate](#)
- typedef struct [table_de_lecture_automate](#) [etats_lus](#)
- typedef struct [mot](#) [mot](#)

Functions

- void [affichage_1D](#) (int *tab, int dim_tab_1D)
- void [affichage_case_tab_1D](#) (int *tab, int position, int dim_tab_1D)
- void [affichage_2D](#) (int **tab, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_case_tab_2D](#) (int **tab, int position_line, int position_col, int dim_tab_2D_ligne, int dim_tab_2D_col)
- void [affichage_ligne_mat](#) (int ***tab, int position_line, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_col_mat](#) (int ***tab, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_case_mat](#) (int ***tab, int position_line, int position_col, int dim_tab_3D_ligne, int dim_tab_3D_col)
- void [affichage_automate](#) (automate a)
- void [lecture_etats_finaux](#) (automate a)
- void [echange](#) (int *a, int *b)
- void [affichage_mot](#) (mot *mot)
- void [lecture_mot](#) (mot m)
- int [est_deterministe](#) (automate a)
- int [est_complet](#) (automate a)
- int [est_un_etat_final](#) (automate a, int etat)
- int [exist_state_in_tab_transi](#) (automate a, int state_touch, int ***tab1)
- int [Next_one_state_touch](#) (automate a, int state_start, int symbol_apply)
- int [calculte_number_states_touch](#) (automate a, int *tab_states_touch)
- int [start_with_tow](#) (int etat_initial_a, int etat_initial_b)
- int [retourne_etat_couant](#) (automate a, int curent_state, int curent_symbol)
- int [est_reconnu](#) (mot m, automate a)
- int [serie_test_reconnaissance](#) (automate a, int n_fois)
- int [reconnu_v2](#) (mot m, automate a)
- int [find_next_state](#) (automate a, mot m)
- int * [allocation_tab_1D](#) (int t1)
- int * [Next_states_touch](#) (automate a, int start_state, int symbol_apply)
- int * [Next_states_touch_not_print](#) (automate a, int start_state, int symbol_apply)
- int * [Tab_states_from_q0](#) (automate a, int symbol_apply)
- int * [Tab_states_from_q0_not_print](#) (automate a, int symbol_apply)
- int * [group_states_by_same_symbol](#) (automate a, int state_apply, int symbol_fix)
- int * [group_states_by_same_symbol_not_print](#) (automate a, int state_apply, int symbol_fix)
- int * [union_states_of_same_symbol](#) (automate a, int state_apply, int symbol_fix)
- int * [union_states_of_same_symbol_not_print](#) (automate a, int state_apply, int symbol_fix)
- int ** [add_state_to_composite_table](#) (automate a, int **tab, int composite_state, int position)
- int ** [copy_line_mat](#) (automate a, int ***mat, int position_line)
- int ** [allocation_tab_2D](#) (int t2, int t3)
- int *** [allocation_mat_vide](#) (int d1, int d2)
- int *** [lecture_automate_court](#) (automate a)
- int *** [add_state_to_rename_table_state](#) (automate a, int ***tan_rename, int new_state, int position)
- int *** [Tab_transition_automate](#) (automate a)
- int *** [Tab_transition_automate_not_print](#) (automate a)
- int *** [twos_symbol_apply](#) (automate a, int state_1, int state_2, int symbol_1, int symbol_2)
- [automate generate_automate_null](#) (int d1, int d2, int d3)
- [automate generate_automate](#) ()
- [automate remplir_automate](#) (automate m, int val)
- [automate import_automae](#) (char path)
- [automate modify_automate](#) (automate a)
- [automate save_automate](#) (automate a)
- [automate del_automate](#) (automate a)
- [automate copie_automate](#) (automate a)
- [automate rendre_complet](#) (automate a)

- [automate concat](#) ([automate a](#), [automate b](#))
- [automate complement_automate](#) ([automate a](#))
- [automate rendre_deterministe](#) ([automate a](#))
- [automate produit_a_b](#) ([automate a](#), [automate b](#))
- [etats_lus lecture_automate_long](#) ([automate a](#))

3.6.1 Detailed Description

Author

Charly romy TANGA - Abdoulaye MBAYE - Berkehan USLU (you@domain.com)

Version

0.1

Date

2024-01-11

Copyright

Copyright (c) 2024

3.6.2 Data Structure Documentation

3.6.2.1 struct automate

Data Fields

| | | |
|---------|------------------|--|
| int | etat_initial | |
| int * | Etats_finaux | |
| int *** | matrice | |
| int | nb_Etats | |
| int | nb_Etats_finaux | |
| int | nb_Symboles | |
| int *** | Table_transition | |

3.6.2.2 struct table_de_lecture_automate

Data Fields

| | | |
|---------|--------------------------|--|
| int *** | etat_lu_depart | |
| int *** | etats_lus_finaux | |
| int *** | etats_lus_intermediaires | |

3.6.2.3 struct mot

Data Fields

| | | |
|-------|------------|--|
| int | size_mot | |
| int * | Tab_caract | |

3.6.3 Typedef Documentation

3.6.3.1 automate

```
typedef struct automate automate
```

3.6.3.2 etats_lus

```
typedef struct table\_de\_lecture\_automate etats\_lus
```

3.6.3.3 mot

```
typedef struct mot mot
```

3.6.4 Function Documentation

3.6.4.1 add_state_to_composite_table()

```
int** add_state_to_composite_table (  
    automate a,  
    int ** tab,  
    int composite_state,  
    int position )
```

Parameters

| | |
|------------------------|--|
| <i>a</i> | |
| <i>tab</i> | |
| <i>composite_state</i> | |
| <i>position</i> | |

Returns

int**

3.6.4.2 add_state_to_rename_table_state()

```
int*** add_state_to_rename_table_state (
    automate a,
    int *** tab_rename,
    int new_state,
    int position )
```

Parameters

| | |
|-------------------|--|
| <i>a</i> | |
| <i>tan_rename</i> | |
| <i>new_state</i> | |
| <i>position</i> | |

Returns

int***

Parameters

| | |
|-------------------|--|
| <i>a</i> | |
| <i>tab_rename</i> | |
| <i>new_state</i> | |
| <i>position</i> | |

Returns

int***

3.6.4.3 affichage_1D()

```
void affichage_1D (
    int * tab,
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>dim_tab_1D</i> | |

3.6.4.4 affichage_2D()

```
void affichage_2D (
    int ** tab,
    int dim_tab_2D_ligne,
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.6.4.5 affichage_automate()

```
void affichage_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.6.4.6 affichage_case_mat()

```
void affichage_case_mat (
    int *** tab,
    int position_line,
    int position_col,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.6.4.7 affichage_case_tab_1D()

```
void affichage_case_tab_1D (
    int * tab,
    int position,
    int dim_tab_1D )
```

Parameters

| | |
|-------------------|--|
| <i>tab</i> | |
| <i>position</i> | |
| <i>dim_tab_1D</i> | |

3.6.4.8 affichage_case_tab_2D()

```
void affichage_case_tab_2D (
    int ** tab,
    int position_line,
    int position_col,
    int dim_tab_2D_ligne,
    int dim_tab_2D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>position_col</i> | |
| <i>dim_tab_2D_ligne</i> | |
| <i>dim_tab_2D_col</i> | |

3.6.4.9 affichage_col_mat()

```
void affichage_col_mat (
    int *** tab,
    int position_col,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_col</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.6.4.10 affichage_ligne_mat()

```
void affichage_ligne_mat (
    int *** tab,
    int position_line,
    int dim_tab_3D_ligne,
    int dim_tab_3D_col )
```

Parameters

| | |
|-------------------------|--|
| <i>tab</i> | |
| <i>position_line</i> | |
| <i>dim_tab_3D_ligne</i> | |
| <i>dim_tab_3D_col</i> | |

3.6.4.11 affichage_mot()

```
void affichage_mot (
    mot * mot )
```

Parameters

| | |
|------------|--|
| <i>mot</i> | |
|------------|--|

3.6.4.12 allocation_mat_vide()

```
int*** allocation_mat_vide (
    int d1,
    int d2 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |

Returns

int***

3.6.4.13 allocation_tab_1D()

```
int* allocation_tab_1D (  
    int t1 )
```

Parameters

| | |
|-----------|--|
| <i>t1</i> | |
|-----------|--|

Returns

int*

3.6.4.14 allocation_tab_2D()

```
int** allocation_tab_2D (  
    int t2,  
    int t3 )
```

Parameters

| | |
|-----------|--|
| <i>t2</i> | |
| <i>t3</i> | |

Returns

int**

3.6.4.15 calculte_number_states_touch()

```
int calculte_number_states_touch (  
    automate a,  
    int * tab_states_touch )
```

Parameters

| | |
|-------------------------|--|
| <i>a</i> | |
| <i>tab_states_touch</i> | |

Returns

int

3.6.4.16 complement_automate()

```
automate complement_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.17 concat()

```
automate concat (  
    automate a,  
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.6.4.18 copie_automate()

```
automate copie_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.19 copy_line_mat()

```
int** copy_line_mat (
    automate a,
    int *** mat,
    int position_line )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>mat</i> | |
| <i>position_line</i> | |

Returns

int**

3.6.4.20 del_automate()

```
automate del_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.21 exchange()

```
void exchange (
    int * a,
    int * b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

3.6.4.22 est_complet()

```
int est_complet (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.6.4.23 est_deterministe()

```
int est_deterministe (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int

3.6.4.24 est_reconnu()

```
int est_reconnu (
    mot m,
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.6.4.25 est_un_etat_final()

```
int est_un_etat_final (
    automate a,
    int etat )
```

Parameters

| | |
|-------------|--|
| <i>a</i> | |
| <i>etat</i> | |

Returns

int

3.6.4.26 exist_state_in_tab_transi()

```
int exist_state_in_tab_transi (
    automate a,
    int state_touch,
    int *** tab1 )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_touch</i> | |
| <i>tab1</i> | |

Returns

int

3.6.4.27 find_next_state()

```
int find_next_state (
    automate a,
    mot m )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>m</i> | |

Returns

int

3.6.4.28 generate_automate()

```
automate generate_automate ( )
```

Returns

automate

3.6.4.29 generate_automate_null()

```
automate generate_automate_null (
    int d1,
    int d2,
    int d3 )
```

Parameters

| | |
|-----------|--|
| <i>d1</i> | |
| <i>d2</i> | |
| <i>d3</i> | |

Returns

automate

3.6.4.30 group_states_by_same_symbol()

```
int* group_states_by_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.6.4.31 group_states_by_same_symbol_not_print()

```
int* group_states_by_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.6.4.32 import_automae()

```
automate import_automae (
    char path )
```

Parameters

| | |
|-------------|--|
| <i>path</i> | |
|-------------|--|

Returns

automate

3.6.4.33 lecture_automate_court()

```
int*** lecture_automate_court (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.6.4.34 lecture_automate_long()

```
etats_lus lecture_automate_long (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

etats_lus

3.6.4.35 lecture_etats_finaux()

```
void lecture_etats_finaux (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

3.6.4.36 lecture_mot()

```
void lecture_mot (  
    mot m )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
|----------|--|

3.6.4.37 modify_automate()

```
automate modify_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.38 Next_one_state_touch()

```
int Next_one_state_touch (
    automate a,
    int state_start,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>state_start</i> | |
| <i>symbol_apply</i> | |

Returns

int

3.6.4.39 Next_states_touch()

```
int* Next_states_touch (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.6.4.40 Next_states_touch_not_print()

```
int* Next_states_touch_not_print (
    automate a,
    int start_state,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>start_state</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.6.4.41 producte_a_b()

```
automate producte_a_b (
    automate a,
    automate b )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
| <i>b</i> | |

Returns

automate

3.6.4.42 reconnu_v2()

```
int reconnu_v2 (
    mot m,
    automate a )
```

Parameters

| | |
|----------|--|
| <i>m</i> | |
| <i>a</i> | |

Returns

int

3.6.4.43 remplir_automate()

```
automate remplir_automate (  
    automate m,  
    int val )
```

Parameters

| | |
|------------|--|
| <i>m</i> | |
| <i>val</i> | |

Returns

automate

3.6.4.44 rendre_complet()

```
automate rendre_complet (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.45 rendre_deterministe()

```
automate rendre_deterministe (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.46 retourne_etat_couant()

```
int retourne_etat_couant (
    automate a,
    int curent_state,
    int curent_symbol )
```

Parameters

| | |
|----------------------|--|
| <i>a</i> | |
| <i>curent_state</i> | |
| <i>curent_symbol</i> | |

Returns

int

3.6.4.47 save_automate()

```
automate save_automate (
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

automate

3.6.4.48 serie_test_reconnaissance()

```
int serie_test_reconnaissance (
    automate a,
    int n_fois )
```

Parameters

| | |
|---------------|--|
| <i>a</i> | |
| <i>n_fois</i> | |

Returns

int

3.6.4.49 start_with_tow()

```
int start_with_tow (
    int etat_initial_a,
    int etat_initial_b )
```

Parameters

| | |
|---|--|
| <i>etat_initial</i> _↔ <i>_a</i> | |
| <i>etat_initial</i> _↔ <i>_b</i> | |

Returns

int

3.6.4.50 Tab_states_from_q0()

```
int* Tab_states_from_q0 (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.6.4.51 Tab_states_from_q0_not_print()

```
int* Tab_states_from_q0_not_print (
    automate a,
    int symbol_apply )
```

Parameters

| | |
|---------------------|--|
| <i>a</i> | |
| <i>symbol_apply</i> | |

Returns

int*

3.6.4.52 Tab_transition_automate()

```
int*** Tab_transition_automate (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.6.4.53 Tab_transition_automate_not_print()

```
int*** Tab_transition_automate_not_print (  
    automate a )
```

Parameters

| | |
|----------|--|
| <i>a</i> | |
|----------|--|

Returns

int***

3.6.4.54 twos_symbol_apply()

```
int*** twos_symbol_apply (  
    automate a,  
    int state_1,  
    int state_2,  
    int symbol_1,  
    int symbol_2 )
```

Parameters

| | |
|---|--|
| <i>a</i> | |
| <i>state_1</i> | |
| <i>state_2</i> | |
| <i>symbol</i> _↔ <i>_1</i> | |
| <i>symbol</i> _↔ <i>_2</i> | |

Returns

int***

3.6.4.55 union_states_of_same_symbol()

```
int* union_states_of_same_symbol (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.6.4.56 union_states_of_same_symbol_not_print()

```
int* union_states_of_same_symbol_not_print (
    automate a,
    int state_apply,
    int symbol_fix )
```

Parameters

| | |
|--------------------|--|
| <i>a</i> | |
| <i>state_apply</i> | |
| <i>symbol_fix</i> | |

Returns

int*

3.7 Proj_C/sources/main.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "Automate.h"
```

Include dependency graph for main.c: This graph shows which files directly or indirectly include this file:

Functions

- int [main](#) ()

3.7.1 Function Documentation

3.7.1.1 main()

```
int main ( )
```

Returns

3.8 sources/main.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include "Automate.h"
```

Include dependency graph for main.c: This graph shows which files directly or indirectly include this file:

Functions

- int [main](#) ()

3.8.1 Function Documentation

3.8.1.1 main()

```
int main ( )
```

Returns

3.9 README.md File Reference

Index

add_state_to_composite_table
 automate.c, [7](#), [29](#)
 Automate.h, [51](#), [73](#)
add_state_to_rename_table_state
 automate.c, [8](#), [29](#)
 Automate.h, [52](#), [74](#)
affichage_1D
 automate.c, [8](#), [30](#)
 Automate.h, [52](#), [74](#)
affichage_2D
 automate.c, [8](#), [30](#)
 Automate.h, [52](#), [75](#)
affichage_automate
 automate.c, [9](#), [30](#)
 Automate.h, [53](#), [75](#)
affichage_case_mat
 automate.c, [9](#), [30](#)
 Automate.h, [53](#), [75](#)
affichage_case_tab_1D
 automate.c, [9](#), [31](#)
 Automate.h, [53](#), [75](#)
affichage_case_tab_2D
 automate.c, [10](#), [31](#)
 Automate.h, [54](#), [76](#)
affichage_col_mat
 automate.c, [10](#), [31](#)
 Automate.h, [54](#), [76](#)
affichage_ligne_mat
 automate.c, [10](#), [32](#)
 Automate.h, [54](#), [77](#)
affichage_mot
 automate.c, [11](#), [32](#)
 Automate.h, [55](#), [77](#)
alloc_mat_vide
 automate.c, [5](#)
alloc_tab_1D
 automate.c, [5](#)
allocation_mat_vide
 automate.c, [11](#), [32](#)
 Automate.h, [55](#), [77](#)
allocation_tab_1D
 automate.c, [11](#), [33](#)
 Automate.h, [55](#), [77](#)
allocation_tab_2D
 automate.c, [12](#), [33](#)
 Automate.h, [56](#), [78](#)
automate, [50](#), [72](#)
 Automate.h, [51](#), [73](#)
automate.c
add_state_to_composite_table, [7](#), [29](#)
add_state_to_rename_table_state, [8](#), [29](#)
affichage_1D, [8](#), [30](#)
affichage_2D, [8](#), [30](#)
affichage_automate, [9](#), [30](#)
affichage_case_mat, [9](#), [30](#)
affichage_case_tab_1D, [9](#), [31](#)
affichage_case_tab_2D, [10](#), [31](#)
affichage_col_mat, [10](#), [31](#)
affichage_ligne_mat, [10](#), [32](#)
affichage_mot, [11](#), [32](#)
alloc_mat_vide, [5](#)
alloc_tab_1D, [5](#)
allocation_mat_vide, [11](#), [32](#)
allocation_tab_1D, [11](#), [33](#)
allocation_tab_2D, [12](#), [33](#)
calculte_number_states_touch, [12](#), [33](#)
complement_automate, [12](#), [34](#)
concat, [13](#), [34](#)
copie_automate, [13](#), [34](#)
copy_line_mat, [13](#), [35](#)
del_automate, [14](#), [35](#)
echange, [14](#), [35](#)
est_complet, [14](#), [36](#)
est_deterministe, [14](#), [36](#)
est_reconnu, [15](#), [36](#)
est_un_etat_final, [15](#), [37](#)
exist_state_in_tab_transi, [15](#), [37](#)
find_next_state, [16](#), [37](#)
generate_automate, [16](#), [38](#)
generate_automate_null, [16](#), [38](#)
generator_automate, [5](#)
group_states_by_same_symbol, [17](#), [38](#)
group_states_by_same_symbol_not_print, [17](#), [39](#)
import_automae, [17](#), [39](#)
lecture_automate_court, [18](#), [39](#)
lecture_automate_long, [18](#), [40](#)
lecture_etats_finaux, [18](#), [40](#)
lecture_mot, [19](#), [40](#)
modify_automate, [19](#), [41](#)
mot_saisi, [19](#), [41](#)
mot_saisi_avant, [19](#), [41](#)
Next_one_state_touch, [19](#), [42](#)
Next_states_touch, [21](#), [42](#)
Next_states_touch_not_print, [21](#), [42](#)
producte_a_b, [21](#), [43](#)
reconnu_v2, [22](#), [43](#)
remplir_automate, [22](#), [43](#)
rendre_complet, [22](#), [44](#)

- rendre_deterministe, [23, 44](#)
- retourne_etat_couant, [23, 44](#)
- save_automate, [23, 45](#)
- serie_test_reconnaissance, [24, 45](#)
- start_with_tow, [24, 45](#)
- Tab_states_from_q0, [24, 46](#)
- Tab_states_from_q0_not_print, [25, 46](#)
- Tab_transition_automate, [25, 47](#)
- Tab_transition_automate_not_print, [25, 47](#)
- twos_symbol_apply, [26, 47](#)
- union_states_of_same_symbol, [26, 48](#)
- union_states_of_same_symbol_not_print, [27, 48](#)
- Automate.h
 - add_state_to_composite_table, [51, 73](#)
 - add_state_to_rename_table_state, [52, 74](#)
 - affichage_1D, [52, 74](#)
 - affichage_2D, [52, 75](#)
 - affichage_automate, [53, 75](#)
 - affichage_case_mat, [53, 75](#)
 - affichage_case_tab_1D, [53, 75](#)
 - affichage_case_tab_2D, [54, 76](#)
 - affichage_col_mat, [54, 76](#)
 - affichage_ligne_mat, [54, 77](#)
 - affichage_mot, [55, 77](#)
 - allocation_mat_vide, [55, 77](#)
 - allocation_tab_1D, [55, 77](#)
 - allocation_tab_2D, [56, 78](#)
 - automate, [51, 73](#)
 - calculte_number_states_touch, [56, 78](#)
 - complement_automate, [56, 78](#)
 - concat, [57, 79](#)
 - copie_automate, [57, 79](#)
 - copy_line_mat, [57, 79](#)
 - del_automate, [58, 80](#)
 - echange, [58, 80](#)
 - est_complet, [58, 80](#)
 - est_deterministe, [59, 81](#)
 - est_reconnu, [59, 81](#)
 - est_un_etat_final, [59, 81](#)
 - etats_lus, [51, 73](#)
 - exist_state_in_tab_transi, [60, 82](#)
 - find_next_state, [60, 82](#)
 - generate_automate, [60, 83](#)
 - generate_automate_null, [60, 83](#)
 - group_states_by_same_symbol, [61, 83](#)
 - group_states_by_same_symbol_not_print, [61, 84](#)
 - import_automae, [62, 84](#)
 - lecture_automate_court, [62, 84](#)
 - lecture_automate_long, [62, 85](#)
 - lecture_etats_finaux, [63, 85](#)
 - lecture_mot, [63, 85](#)
 - modify_automate, [63, 85](#)
 - mot, [51, 73](#)
 - Next_one_state_touch, [63, 86](#)
 - Next_states_touch, [64, 86](#)
 - Next_states_touch_not_print, [64, 86](#)
 - producte_a_b, [64, 87](#)
 - reconnu_v2, [65, 87](#)
 - remplir_automate, [65, 88](#)
 - rendre_complet, [65, 88](#)
 - rendre_deterministe, [66, 88](#)
 - retourne_etat_couant, [66, 89](#)
 - save_automate, [66, 89](#)
 - serie_test_reconnaissance, [67, 89](#)
 - start_with_tow, [67, 90](#)
 - Tab_states_from_q0, [67, 90](#)
 - Tab_states_from_q0_not_print, [68, 90](#)
 - Tab_transition_automate, [68, 91](#)
 - Tab_transition_automate_not_print, [68, 91](#)
 - twos_symbol_apply, [69, 91](#)
 - union_states_of_same_symbol, [69, 92](#)
 - union_states_of_same_symbol_not_print, [70, 92](#)
- calculte_number_states_touch
 - automate.c, [12, 33](#)
 - Automate.h, [56, 78](#)
- complement_automate
 - automate.c, [12, 34](#)
 - Automate.h, [56, 78](#)
- concat
 - automate.c, [13, 34](#)
 - Automate.h, [57, 79](#)
- Contribution au projet/sources/automate.c, [5](#)
- Contribution au projet/sources/Automate.h, [49](#)
- copie_automate
 - automate.c, [13, 34](#)
 - Automate.h, [57, 79](#)
- copy_line_mat
 - automate.c, [13, 35](#)
 - Automate.h, [57, 79](#)
- del_automate
 - automate.c, [14, 35](#)
 - Automate.h, [58, 80](#)
- echange
 - automate.c, [14, 35](#)
 - Automate.h, [58, 80](#)
- est_complet
 - automate.c, [14, 36](#)
 - Automate.h, [58, 80](#)
- est_deterministe
 - automate.c, [14, 36](#)
 - Automate.h, [59, 81](#)
- est_reconnu
 - automate.c, [15, 36](#)
 - Automate.h, [59, 81](#)
- est_un_etat_final
 - automate.c, [15, 37](#)
 - Automate.h, [59, 81](#)
- etats_lus
 - Automate.h, [51, 73](#)
- exist_state_in_tab_transi
 - automate.c, [15, 37](#)
 - Automate.h, [60, 82](#)
- find_next_state

- automate.c, [16](#), [37](#)
- Automate.h, [60](#), [82](#)
- generate_automate
 - automate.c, [16](#), [38](#)
 - Automate.h, [60](#), [83](#)
- generate_automate_null
 - automate.c, [16](#), [38](#)
 - Automate.h, [60](#), [83](#)
- generator_automate
 - automate.c, [5](#)
- group_states_by_same_symbol
 - automate.c, [17](#), [38](#)
 - Automate.h, [61](#), [83](#)
- group_states_by_same_symbol_not_print
 - automate.c, [17](#), [39](#)
 - Automate.h, [61](#), [84](#)
- import_automae
 - automate.c, [17](#), [39](#)
 - Automate.h, [62](#), [84](#)
- lecture_automate_court
 - automate.c, [18](#), [39](#)
 - Automate.h, [62](#), [84](#)
- lecture_automate_long
 - automate.c, [18](#), [40](#)
 - Automate.h, [62](#), [85](#)
- lecture_etats_finaux
 - automate.c, [18](#), [40](#)
 - Automate.h, [63](#), [85](#)
- lecture_mot
 - automate.c, [19](#), [40](#)
 - Automate.h, [63](#), [85](#)
- main
 - main.c, [93](#)
- main.c
 - main, [93](#)
- modify_automate
 - automate.c, [19](#), [41](#)
 - Automate.h, [63](#), [85](#)
- mot, [51](#), [72](#)
 - Automate.h, [51](#), [73](#)
- mot_saisi
 - automate.c, [19](#), [41](#)
- mot_saisi_avant
 - automate.c, [19](#), [41](#)
- Next_one_state_touch
 - automate.c, [19](#), [42](#)
 - Automate.h, [63](#), [86](#)
- Next_states_touch
 - automate.c, [21](#), [42](#)
 - Automate.h, [64](#), [86](#)
- Next_states_touch_not_print
 - automate.c, [21](#), [42](#)
 - Automate.h, [64](#), [86](#)
- producte_a_b
 - automate.c, [21](#), [43](#)
 - Automate.h, [64](#), [87](#)
- Projet_C/sources/automate.c, [6](#)
- Projet_C/sources/Automate.h, [49](#)
- Projet_C/sources/main.c, [93](#)
- README.md, [94](#)
- reconnu_v2
 - automate.c, [22](#), [43](#)
 - Automate.h, [65](#), [87](#)
- remplir_automate
 - automate.c, [22](#), [43](#)
 - Automate.h, [65](#), [88](#)
- rendre_complet
 - automate.c, [22](#), [44](#)
 - Automate.h, [65](#), [88](#)
- rendre_deterministe
 - automate.c, [23](#), [44](#)
 - Automate.h, [66](#), [88](#)
- retourne_etat_couant
 - automate.c, [23](#), [44](#)
 - Automate.h, [66](#), [89](#)
- save_automate
 - automate.c, [23](#), [45](#)
 - Automate.h, [66](#), [89](#)
- serie_test_reconnaissance
 - automate.c, [24](#), [45](#)
 - Automate.h, [67](#), [89](#)
- sources/automate.c, [27](#)
- sources/Automate.h, [70](#)
- sources/main.c, [93](#)
- start_with_tow
 - automate.c, [24](#), [45](#)
 - Automate.h, [67](#), [90](#)
- Tab_states_from_q0
 - automate.c, [24](#), [46](#)
 - Automate.h, [67](#), [90](#)
- Tab_states_from_q0_not_print
 - automate.c, [25](#), [46](#)
 - Automate.h, [68](#), [90](#)
- Tab_transition_automate
 - automate.c, [25](#), [47](#)
 - Automate.h, [68](#), [91](#)
- Tab_transition_automate_not_print
 - automate.c, [25](#), [47](#)
 - Automate.h, [68](#), [91](#)
- table_de_lecture_automate, [51](#), [72](#)
- twos_symbol_apply
 - automate.c, [26](#), [47](#)
 - Automate.h, [69](#), [91](#)
- union_states_of_same_symbol
 - automate.c, [26](#), [48](#)
 - Automate.h, [69](#), [92](#)
- union_states_of_same_symbol_not_print
 - automate.c, [27](#), [48](#)
 - Automate.h, [70](#), [92](#)