

Prediction medicale

Generated by Doxygen 1.8.12

Contents

1	Data Structure Index	1
1.1	Data Structures	1
2	File Index	3
2.1	File List	3
3	Data Structure Documentation	5
3.1	pile Struct Reference	5
3.2	Pile Struct Reference	5
3.2.1	Detailed Description	5
4	File Documentation	7
4.1	contextmedical.c File Reference	7
4.1.1	Detailed Description	7
4.1.2	Function Documentation	7
4.1.2.1	lirecontexte()	7
4.2	contextmedical.h File Reference	8
4.2.1	Detailed Description	8
4.2.2	Function Documentation	8
4.2.2.1	lirecontexte()	8
4.3	main.c File Reference	9
4.3.1	Detailed Description	9
4.4	patient.c File Reference	10
4.4.1	Detailed Description	10
4.4.2	Function Documentation	10

4.4.2.1	creermotif()	10
4.4.2.2	limitermotif()	11
4.4.2.3	liremotif()	11
4.5	patient.h File Reference	11
4.5.1	Detailed Description	12
4.5.2	Function Documentation	12
4.5.2.1	creermotif()	12
4.5.2.2	limitermotif()	13
4.5.2.3	liremotif()	13
4.6	pile.c File Reference	13
4.6.1	Detailed Description	14
4.6.2	Function Documentation	14
4.6.2.1	depiler()	14
4.6.2.2	empiler()	15
4.6.2.3	estvide()	15
4.6.2.4	existe()	15
4.6.2.5	vider()	16
4.6.2.6	visualiser()	16
4.7	pile.h File Reference	16
4.7.1	Detailed Description	17
4.7.2	Function Documentation	17
4.7.2.1	depiler()	17
4.7.2.2	empiler()	17
4.7.2.3	estvide()	18
4.7.2.4	existe()	18
4.7.2.5	vider()	18
4.7.2.6	visualiser()	18
4.8	prediction.c File Reference	19
4.8.1	Detailed Description	19
4.8.2	Function Documentation	20

4.8.2.1	conversionint()	20
4.8.2.2	conversiontmotif()	20
4.8.2.3	nomfichierprediction()	20
4.9	prediction.h File Reference	21
4.9.1	Detailed Description	21
4.9.2	Function Documentation	21
4.9.2.1	conversionint()	21
4.9.2.2	conversiontmotif()	22
4.9.2.3	nomfichierprediction()	22
4.10	predictionpartielle.c File Reference	23
4.10.1	Detailed Description	23
4.10.2	Function Documentation	24
4.10.2.1	calculerProba()	24
4.10.2.2	predictionpartielle_motif_2_hospitalisations()	24
4.10.2.3	predictionpartielle_motif_3_hospitalisations()	24
4.11	predictionpartielle.h File Reference	25
4.11.1	Detailed Description	25
4.11.2	Function Documentation	26
4.11.2.1	calculerProba()	26
4.11.2.2	predictionpartielle_motif_2_hospitalisations()	26
4.11.2.3	predictionpartielle_motif_3_hospitalisations()	26
Index		29

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

pile	5
Pile		
Pile contenant les predictions	5

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

contextmedical.c	Contexte medical dans lequel on souhaite realiser la prediction	7
contextmedical.h	Contexte medical dans lequel on souhaite realiser la prediction	8
main.c	Fonction principale du programme	9
patient.c	Antecedents du patient (motif)	10
patient.h	Antecedents du patient (motif)	11
pile.c	Structure de pile. Permet de creer une liste de predictions	13
pile.h	Structure de pile. Permet de creer une liste de predictions	16
prediction.c	Prediction	19
prediction.h	Prediction des suites d'un patient a partir de l'analyse des motifs de patients contenus dans une base sequentielle	21
predictionpartielle.c	Prediction partielle sur un motif de patient (cas ou il n'existe pas de correspondance exacte entre le motif du patient et la base sequentielle)	23
predictionpartielle.h	Prediction partielle sur un motif de patient (cas ou il n'existe pas de correspondance exacte entre le motif du patient et la base sequentielle)	25

Chapter 3

Data Structure Documentation

3.1 pile Struct Reference

Data Fields

- int **valeur**
- struct [pile](#) * **prec**

The documentation for this struct was generated from the following file:

- [pile.h](#)

3.2 Pile Struct Reference

[Pile](#) contenant les predictions.

```
#include <pile.h>
```

3.2.1 Detailed Description

[Pile](#) contenant les predictions.

The documentation for this struct was generated from the following file:

- [pile.h](#)

Chapter 4

File Documentation

4.1 contextemedical.c File Reference

Contexte medical dans lequel on souhaite realiser la prediction.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "contextemedical.h"
```

Functions

- int [lirecontexte](#) (char a)

Fonction permettant de convertir le contexte medical (ecrit sous forme de caractere dans la console) en l'entier qui lui est associe. Un contexte medical est cense etre represente par un entier.

4.1.1 Detailed Description

Contexte medical dans lequel on souhaite realiser la prediction.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.1.2 Function Documentation

4.1.2.1 lirecontexte()

```
int lirecontexte (
    char a )
```

Fonction permettant de convertir le contexte medical (ecrit sous forme de caractere dans la console) en l'entier qui lui est associe. Un contexte medical est cense etre represente par un entier.

Parameters

<i>a</i>	- le contexte medical sous forme de caractere
----------	---

Returns

aint - l'entier represente par a

4.2 contextemedical.h File Reference

Contexte medical dans lequel on souhaite realiser la prediction.

Functions

- int [lirecontexte](#) (char a)

Fonction permettant de convertir le contexte medical (ecrit sous forme de caractere dans la console) en l'entier qui lui est associe. Un contexte medical est cense etre represente par un entier.

4.2.1 Detailed Description

Contexte medical dans lequel on souhaite realiser la prediction.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.2.2 Function Documentation

4.2.2.1 lirecontexte()

```
int lirecontexte (  
    char a )
```

Fonction permettant de convertir le contexte medical (ecrit sous forme de caractere dans la console) en l'entier qui lui est associe. Un contexte medical est cense etre represente par un entier.

Parameters

<i>a</i>	- le contexte medical sous forme de caractere
----------	---

Returns

aint - l'entier represente par a

4.3 main.c File Reference

Fonction principale du programme.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "patient.h"
#include "contextemedical.h"
#include "prediction.h"
#include "predictionpartielle.h"
```

Functions

- int **main** (int argc, char *argv[])

4.3.1 Detailed Description

Fonction principale du programme.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.4 patient.c File Reference

Antecedents du patient (motif)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "patient.h"
```

Functions

- char * [creermotif](#) (int argc, char *argv[])
Fonction permettant de recuperer le motif d'un patient passe en parametre sur le terminal (ses antecedents medicaux)
- int * [liremotif](#) (char *motif, size_t *taille)
Fonction permettant de construire le tableau d'hospitalisations correspondant au motif du patient.
- void [limitermotif](#) (int *tab_motif, size_t *taille)
Procedure permettant de limiter le motif d'un patient a ses trois dernieres hospitalisations, afin de proposer une prediction pertinente.

4.4.1 Detailed Description

Antecedents du patient (motif)

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.4.2 Function Documentation

4.4.2.1 creermotif()

```
char * creermotif (
    int argc,
    char * argv[ ] )
```

Fonction permettant de recuperer le motif d'un patient passe en parametre sur le terminal (ses antecedents medicaux)

Parameters

<i>argc</i>	- le nombre de parametres passes au programme
<i>argv</i>	- les parametres passes au programme

Returns

motif - le motif du patient

4.4.2.2 limitermotif()

```
void limitermotif (
    int * tab_motif,
    size_t * taille )
```

Procédure permettant de limiter le motif d'un patient a ses trois dernieres hospitalisations, afin de proposer une prediction pertinente.

Parameters

<i>tab_motif</i>	- le motif complet du patient sous forme de tableau d'entiers (chaque entier correspond a une hospitalisation)
<i>taille</i>	- la taille du motif du patient (complet)

4.4.2.3 liremotif()

```
int * liremotif (
    char * motif,
    size_t * taille )
```

Fonction permettant de construire le tableau d'hospitalisations correspondant au motif du patient.

Parameters

<i>motif</i>	- le motif du patient sous forme de chaine de caracteres (chaque hospitalisation dans le motif est separee de la suivante par un espace)
<i>taille</i>	- la taille du motif du patient

Returns

tab_motif - le motif du patient sous forme de tableau d'entiers (chaque entier correspondant a une hospitalisation du motif)

4.5 patient.h File Reference

Antecedents du patient (motif)

Functions

- char * **creermotif** (int argc, char *argv[])
Fonction permettant de recuperer le motif d'un patient passe en parametre sur le terminal (ses antecedents medicaux)
- int * **liremotif** (char *motif, size_t *taille)
Fonction permettant de construire le tableau d'hospitalisations correspondant au motif du patient.
- void **limitermotif** (int *tab_motif, size_t *taille)
Procedure permettant de limiter le motif d'un patient a ses trois dernieres hospitalisations, afin de proposer une prediction pertinente.

4.5.1 Detailed Description

Antecedents du patient (motif)

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.5.2 Function Documentation

4.5.2.1 creermotif()

```
char* creermotif (
    int argc,
    char * argv[] )
```

Fonction permettant de recuperer le motif d'un patient passe en parametre sur le terminal (ses antecedents medicaux)

Parameters

<i>argc</i>	- le nombre de parametres passes au programme
<i>argv</i>	- les parametres passes au programme

Returns

motif - le motif du patient

4.5.2.2 limitermotif()

```
void limitermotif (
    int * tab_motif,
    size_t * taille )
```

Procedure permettant de limiter le motif d'un patient a ses trois dernieres hospitalisations, afin de proposer une prediction pertinente.

Parameters

<i>tab_motif</i>	- le motif complet du patient sous forme de tableau d'entiers (chaque entier correspond a une hospitalisation)
<i>taille</i>	- la taille du motif du patient (complet)

4.5.2.3 liremotif()

```
int* liremotif (
    char * motif,
    size_t * taille )
```

Fonction permettant de construire le tableau d'hospitalisations correspondant au motif du patient.

Parameters

<i>motif</i>	- le motif du patient sous forme de chaine de caracteres (chaque hospitalisation dans le motif est separee de la suivante par un espace)
<i>taille</i>	- la taille du motif du patient

Returns

tab_motif - le motif du patient sous forme de tableau d'entiers (chaque entier correspondant a une hospitalisation du motif)

4.6 pile.c File Reference

Structure de pile. Permet de creer une liste de predictions.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "pile.h"
```

Functions

- void [empiler](#) ([pile](#) **, int valeur)
Fonction permettant d'empiler une valeur.

- int `depiler (pile **p)`
Fonction permettant de depiler une valeur.
- void `vider (pile **p)`
Fonction permettant de vider la pile.
- void `visualiser (pile *p)`
Fonction permettant de visualiser la pile.
- int `existe (pile *p, int valeur)`
Fonction permettant de verifier si une valeur existe dans la pile.
- int `estvide (pile **p)`
Fonction permettant de verifier si une pile est vide.

4.6.1 Detailed Description

Structure de pile. Permet de creer une liste de predictions.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.6.2 Function Documentation

4.6.2.1 depiler()

```
depiler (  
    pile ** p )
```

Fonction permettant de depiler une valeur.

Parameters

<i>p</i>	- la pile
----------	-----------

Returns

valeur - la valeur qu'on vient de depiler

4.6.2.2 empiler()

```
empiler (
    pile ** p,
    int val )
```

Fonction permettant d'empiler une valeur.

Parameters

<i>p</i>	- la pile
<i>valeur</i>	- la valeur a empiler

4.6.2.3 estvide()

```
estvide (
    pile ** p )
```

Fonction permettant de verifier si une pile est vide.

Parameters

<i>p</i>	- la pile
----------	-----------

Returns

1 si la pile est vide
0 si la pile n'est pas vide

4.6.2.4 existe()

```
existe (
    pile * p,
    int valeur )
```

Fonction permettant de verifier si une valeur existe dans la pile.

Parameters

<i>p</i>	- la pile
<i>valeur</i>	- la valeur dont on veut verifier la presence dans la pile

Returns

existe - 0 si la valeur n'existe pas dans la pile, 1 sinon

4.6.2.5 vider()

```
vider (
    pile ** p )
```

Fonction permettant de vider la pile.

Parameters

<i>p</i>	- la pile
----------	-----------

4.6.2.6 visualiser()

```
visualiser (
    pile * p )
```

Fonction permettant de visualiser la pile.

Parameters

<i>p</i>	- la pile
----------	-----------

4.7 pile.h File Reference

Structure de pile. Permet de creer une liste de predictions.

Data Structures

- struct [pile](#)

Typedefs

- typedef struct [pile](#) **pile**

Functions

- void [empiler](#) ([pile](#) **p, int valeur)
Fonction permettant d'empiler une valeur.
- int [depiler](#) ([pile](#) **p)
Fonction permettant de depiler une valeur.
- void [vider](#) ([pile](#) **p)
Fonction permettant de vider la pile.
- void [visualiser](#) ([pile](#) *p)
Fonction permettant de visualiser la pile.
- int [existe](#) ([pile](#) *p, int valeur)
Fonction permettant de verifier si une valeur existe dans la pile.
- int [estvide](#) ([pile](#) **p)
Fonction permettant de verifier si une pile est vide.

4.7.1 Detailed Description

Structure de pile. Permet de creer une liste de predictions.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.7.2 Function Documentation

4.7.2.1 depiler()

```
int depiler (
    pile ** p )
```

Fonction permettant de depiler une valeur.

Parameters

<i>p</i>	- la pile
----------	-----------

Returns

valeur - la valeur qu'on vient de depiler

4.7.2.2 empiler()

```
void empiler (
    pile ** p,
    int valeur )
```

Fonction permettant d'empiler une valeur.

Parameters

<i>p</i>	- la pile
<i>valeur</i>	- la valeur a empiler

4.7.2.3 estvide()

```
int estvide (
    pile ** p )
```

Fonction permettant de verifier si une pile est vide.

Parameters

<i>p</i>	- la pile
----------	-----------

Returns

1 si la pile est vide
0 si la pile n'est pas vide

4.7.2.4 existe()

```
int existe (
    pile * p,
    int valeur )
```

Fonction permettant de verifier si une valeur existe dans la pile.

Parameters

<i>p</i>	- la pile
<i>valeur</i>	- la valeur dont on veut verifier la presence dans la pile

Returns

existe - 0 si la valeur n'existe pas dans la pile, 1 sinon

4.7.2.5 vider()

```
void vider (
    pile ** p )
```

Fonction permettant de vider la pile.

Parameters

<i>p</i>	- la pile
----------	-----------

4.7.2.6 visualiser()

```
void visualiser (
```



```
pile * p )
```

Fonction permettant de visualiser la pile.

Parameters

<i>p</i>	- la pile
----------	-----------

4.8 prediction.c File Reference

Prediction.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "pile.h"
#include "prediction.h"
#include "patient.h"
```

Functions

- char * [conversionint](#) (int i)
Fonction de conversion d'un entier en chaine de caractere. Utile pour traiter les motifs de la base sequentielle.
- char * [conversiontmotif](#) (int *tmotif, int taillemotif)
Fonction de conversion d'un tableau d'entiers (tableau d'hospitalisations) en chaine de caracteres (motif). Utile pour traiter les motifs de la base sequentielle.
- char * [nomfichierprediction](#) (int contexte, int *motif, size_t *taille)
Fonction permettant d'ecrire le nom du fichier qui contiendra les predictions associees a un motif dans un contexte medical precis.
- int **score** (int contexte, int *motif, size_t *taille)
- int **prediction** (int contexte, int *motif, size_t *taille)

4.8.1 Detailed Description

Prediction.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.8.2 Function Documentation

4.8.2.1 conversionint()

```
char * conversionint (
    int i )
```

Fonction de conversion d'un entier en chaine de caractere. Utile pour traiter les motifs de la base sequentielle.

Parameters

<i>i</i>	- l'entier a convertir
----------	------------------------

Returns

s - la chaine de caractere equivalente a l'entier

4.8.2.2 conversiontmotif()

```
char * conversiontmotif (
    int * tmotif,
    int taillemotif )
```

Fonction de conversion d'un tableau d'entiers (tableau d'hospitalisations) en chaine de caracteres (motif). Utile pour traiter les motifs de la base sequentielle.

Parameters

<i>tmotif</i>	- le tableau d'entiers representant un motif
<i>taillemotif</i>	- la taille du tableau d'entiers

Returns

motif - le motif sous forme de chaine de caracteres

4.8.2.3 nomfichierprediction()

```
char * nomfichierprediction (
    int contexte,
    int * motif,
    size_t * taille )
```

Fonction permettant d'ecrire le nom du fichier qui contiendra les predictions associees a un motif dans un contexte medical precis.

Parameters

<i>contexte</i>	- le contexte medical dans lequel on se place
<i>motif</i>	- le motif du patient
<i>taille</i>	- la taille du motif du patient

Returns

nom_fichier - le nom du fichier qui contiendra les predictions associees au motif dans le contexte medical

4.9 prediction.h File Reference

Prediction des suites d'un patient a partir de l'analyse des motifs de patients contenus dans une base sequentielle.

Functions

- char * [conversionint](#) (int i)
Fonction de conversion d'un entier en chaine de caractere. Utile pour traiter les motifs de la base sequentielle.
- char * [conversiontmotif](#) (int *tmotif, int taillemotif)
Fonction de conversion d'un tableau d'entiers (tableau d'hospitalisations) en chaine de caracteres (motif). Utile pour traiter les motifs de la base sequentielle.
- char * [nomfichierprediction](#) (int contexte, int *motif, size_t *taille)
Fonction permettant d'ecrire le nom du fichier qui contiendra les predictions associees a un motif dans un contexte medical precis.
- int **score** (int contexte, int *motif, size_t *taille)
- int **prediction** (int contexte, int *motif, size_t *taille)

4.9.1 Detailed Description

Prediction des suites d'un patient a partir de l'analyse des motifs de patients contenus dans une base sequentielle.

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.9.2 Function Documentation

4.9.2.1 conversionint()

```
char* conversionint (  
    int i )
```

Fonction de conversion d'un entier en chaine de caractere. Utile pour traiter les motifs de la base sequentielle.

Parameters

<i>i</i>	- l'entier a convertir
----------	------------------------

Returns

s - la chaine de caractere equivalente a l'entier

4.9.2.2 conversiontmotif()

```
char* conversiontmotif (  
    int * tmotif,  
    int taillemotif )
```

Fonction de conversion d'un tableau d'entiers (tableau d'hospitalisations) en chaine de caracteres (motif). Utile pour traiter les motifs de la base sequentielle.

Parameters

<i>tmotif</i>	- le tableau d'entiers representant un motif
<i>taillemotif</i>	- la taille du tableau d'entiers

Returns

motif - le motif sous forme de chaine de caracteres

4.9.2.3 nomfichierprediction()

```
char* nomfichierprediction (  
    int contexte,  
    int * motif,  
    size_t * taille )
```

Fonction permettant d'ecrire le nom du fichier qui contiendra les predictions associees a un motif dans un contexte medical precis.

Parameters

<i>contexte</i>	- le contexte medical dans lequel on se place
<i>motif</i>	- le motif du patient
<i>taille</i>	- la taille du motif du patient

Returns

nom_fichier - le nom du fichier qui contiendra les predictions associees au motif dans le contexte medical

4.10 predictionpartielle.c File Reference

Prediction partielle sur un motif de patient (cas ou il n'existe pas de correspondance exacte entre le motif du patient et la base sequentielle)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "pile.h"
#include "patient.h"
#include "prediction.h"
#include "predictionpartielle.h"
```

Functions

- int **scoremax** (int contexte, int *motif_1, int *motif_2, int *motif_3)
- [pile](#) * **remplirListePredictions2** (int contexte, int *motif)
- float **calculerProba2** (int contexte, int pred, int *motif, int amotif)
- [pile](#) * **remplirListePredictions1** (int contexte, int *hospitalisation_1, int *hospitalisation_2, int *hospitalisation_3)
- float **calculerProba1** (int contexte, int pred, int *hospitalisation_1, int *hospitalisation_2, int *hospitalisation_3)
- void [predictionpartielle_motif_3_hospitalisations](#) (int contexte, int *motif)
Procédure permettant de réaliser une prediction partielle dans un contexte medical donne pour un motif compose de trois hospitalisations.
- [pile](#) * **remplirListePredictions** (int contexte, int *hospitalisation_1, int *hospitalisation_2)
- float [calculerProba](#) (int contexte, int pred, int *hospitalisation_1, int *hospitalisation_2)
Fonction permettant de calculer la probabilité d'une prediction pour chaque hospitalisation prise de maniere isolee.
- void [predictionpartielle_motif_2_hospitalisations](#) (int contexte, int *motif)
Procédure permettant de réaliser une prediction partielle dans un contexte medical donne pour un motif compose de deux hospitalisations.
- void **predictionpartielle** (int contexte, int *motif, size_t *taille)

4.10.1 Detailed Description

Prediction partielle sur un motif de patient (cas ou il n'existe pas de correspondance exacte entre le motif du patient et la base sequentielle)

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.10.2 Function Documentation

4.10.2.1 calculerProba()

```
calculerProba (
    int contexte,
    int pred,
    int * hospitalisation_1,
    int * hospitalisation_2 )
```

Fonction permettant de calculer la probabilité d'une prédiction pour chaque hospitalisation prise de manière isolée.

Parameters

<i>contexte</i>	- le contexte médical considéré
<i>pred</i>	- la prédiction
<i>hospitalisation_1</i>	- une hospitalisation
<i>hospitalisation_2</i>	- une hospitalisation

Returns

proba - la probabilité de la prédiction

4.10.2.2 predictionpartielle_motif_2_hospitalisations()

```
predictionpartielle_motif_2_hospitalisations (
    int contexte,
    int * motif )
```

Procédure permettant de réaliser une prédiction partielle dans un contexte médical donné pour un motif composé de deux hospitalisations.

Parameters

<i>contexte</i>	- le contexte médical considéré
<i>motif</i>	- le motif du patient composé de deux hospitalisations

4.10.2.3 predictionpartielle_motif_3_hospitalisations()

```
predictionpartielle_motif_3_hospitalisations (
    int contexte,
    int * motif )
```

Procédure permettant de réaliser une prédiction partielle dans un contexte médical donné pour un motif composé de trois hospitalisations.

Parameters

<i>contexte</i>	- le contexte medical considere
<i>motif</i>	- le motif du patient compose de trois hospitalisations

4.11 predictionpartielle.h File Reference

Prediction partielle sur un motif de patient (cas ou il n'existe pas de correspondance exacte entre le motif du patient et la base sequentielle)

```
#include "pile.h"
#include "prediction.h"
```

Functions

- int **scoremax** (int contexte, int *motif_1, int *motif_2, int *motif_3)
- [pile](#) * **remplirListePredictions2** (int contexte, int *motif)
- float **calculerProba2** (int contexte, int pred, int *motif, int amotif)
- [pile](#) * **remplirListePredictions1** (int contexte, int *hospitalisation_1, int *hospitalisation_2, int *hospitalisation_3)
- float **calculerProba1** (int contexte, int pred, int *hospitalisation_1, int *hospitalisation_2, int *hospitalisation_3)
- void [predictionpartielle_motif_3_hospitalisations](#) (int contexte, int *motif)
Procédure permettant de réaliser une prediction partielle dans un contexte medical donne pour un motif compose de trois hospitalisations.
- [pile](#) * **remplirListePredictions** (int contexte, int *hospitalisation_1, int *hospitalisation_2)
- float [calculerProba](#) (int contexte, int pred, int *hospitalisation_1, int *hospitalisation_2)
Fonction permettant de calculer la probabilité d'une prediction pour chaque hospitalisation prise de maniere isolee.
- void [predictionpartielle_motif_2_hospitalisations](#) (int contexte, int *motif)
Procédure permettant de réaliser une prediction partielle dans un contexte medical donne pour un motif compose de deux hospitalisations.
- void **predictionpartielle** (int contexte, int *motif, size_t *taille)

4.11.1 Detailed Description

Prediction partielle sur un motif de patient (cas ou il n'existe pas de correspondance exacte entre le motif du patient et la base sequentielle)

Author

Stella Zevio

Version

1.0

Date

2015-06-20

Prediction medicale

4.11.2 Function Documentation

4.11.2.1 calculerProba()

```
float calculerProba (
    int contexte,
    int pred,
    int * hospitalisation_1,
    int * hospitalisation_2 )
```

Fonction permettant de calculer la probabilité d'une prédiction pour chaque hospitalisation prise de manière isolée.

Parameters

<i>contexte</i>	- le contexte médical considéré
<i>pred</i>	- la prédiction
<i>hospitalisation_1</i>	- une hospitalisation
<i>hospitalisation_2</i>	- une hospitalisation

Returns

proba - la probabilité de la prédiction

4.11.2.2 predictionpartielle_motif_2_hospitalisations()

```
void predictionpartielle_motif_2_hospitalisations (
    int contexte,
    int * motif )
```

Procédure permettant de réaliser une prédiction partielle dans un contexte médical donné pour un motif composé de deux hospitalisations.

Parameters

<i>contexte</i>	- le contexte médical considéré
<i>motif</i>	- le motif du patient composé de deux hospitalisations

4.11.2.3 predictionpartielle_motif_3_hospitalisations()

```
void predictionpartielle_motif_3_hospitalisations (
    int contexte,
    int * motif )
```

Procédure permettant de réaliser une prédiction partielle dans un contexte médical donné pour un motif composé de trois hospitalisations.

Parameters

<i>contexte</i>	- le contexte medical considere
<i>motif</i>	- le motif du patient compose de trois hospitalisations

Index

calculerProba
 predictionpartielle.c, [24](#)
 predictionpartielle.h, [26](#)
contextemedical.c, [7](#)
 lirecontexte, [7](#)
contextemedical.h, [8](#)
 lirecontexte, [8](#)
conversionint
 prediction.c, [20](#)
 prediction.h, [21](#)
conversiontmotif
 prediction.c, [20](#)
 prediction.h, [22](#)
creermotif
 patient.c, [10](#)
 patient.h, [12](#)

depiler
 pile.c, [14](#)
 pile.h, [17](#)

empiler
 pile.c, [14](#)
 pile.h, [17](#)
estvide
 pile.c, [15](#)
 pile.h, [18](#)
existe
 pile.c, [15](#)
 pile.h, [18](#)

limitermotif
 patient.c, [11](#)
 patient.h, [12](#)
lirecontexte
 contextemedical.c, [7](#)
 contextemedical.h, [8](#)
liremotif
 patient.c, [11](#)
 patient.h, [13](#)

main.c, [9](#)

nomfichierprediction
 prediction.c, [20](#)
 prediction.h, [22](#)

patient.c, [10](#)
 creermotif, [10](#)
 limitermotif, [11](#)
 liremotif, [11](#)

patient.h, [11](#)
 creermotif, [12](#)
 limitermotif, [12](#)
 liremotif, [13](#)
Pile, [5](#)
pile, [5](#)
pile.c, [13](#)
 depiler, [14](#)
 empiler, [14](#)
 estvide, [15](#)
 existe, [15](#)
 vider, [15](#)
 visualiser, [16](#)
pile.h, [16](#)
 depiler, [17](#)
 empiler, [17](#)
 estvide, [18](#)
 existe, [18](#)
 vider, [18](#)
 visualiser, [18](#)
prediction.c, [19](#)
 conversionint, [20](#)
 conversiontmotif, [20](#)
 nomfichierprediction, [20](#)
prediction.h, [21](#)
 conversionint, [21](#)
 conversiontmotif, [22](#)
 nomfichierprediction, [22](#)
predictionpartielle.c, [23](#)
 calculerProba, [24](#)
 predictionpartielle_motif_2_hospitalisations, [24](#)
 predictionpartielle_motif_3_hospitalisations, [24](#)
predictionpartielle.h, [25](#)
 calculerProba, [26](#)
 predictionpartielle_motif_2_hospitalisations, [26](#)
 predictionpartielle_motif_3_hospitalisations, [26](#)
predictionpartielle_motif_2_hospitalisations
 predictionpartielle.c, [24](#)
 predictionpartielle.h, [26](#)
predictionpartielle_motif_3_hospitalisations
 predictionpartielle.c, [24](#)
 predictionpartielle.h, [26](#)

vider
 pile.c, [15](#)
 pile.h, [18](#)
visualiser
 pile.c, [16](#)
 pile.h, [18](#)