**Projet 2I006**

*Jean de Bodinat*

*Yannis Karmim*

*Rapport*

27/04/2017

|  |
| --- |
| SOMMAIRE |

I. SUJET DU PROJET 2

II. LES STRUCTURES MANIPULÉES 3

III. JEUX D’ESSAIS 3

I. Définir une API web. 3

1. Structure de l’Application. 3

2. Interface de programmation applicative 3

3. Jeu de test d’intégration 4

IV. ANALYSE DES JEUX D’ESSAIS 5

1 - Ordonnancement et durée des phases. 5

Phase 1 : Analyse préliminaire. 5

Phase 2 : Analyse fonctionnelle de l’application. 5

Phase 3 : Définition de l’API Web. 5

Phase 4 : Développement d’un jeu de test. 6

2 - Moyens et outils techniques utilisés. 6

3 - Pilotage de projet et suivi. 6

|  |
| --- |
| **I.** SUJET DU PROJET |

Dans le cadre de l’étude des structures de données ainsi que des algorithmes parcourant ces mêmes données, nous avons travaillé sur des réseaux allant de 12 et 15 liaisons à 9000 nœuds et 11000 liaisons.

Dans une première partie nous avons développé des graphes à partir de listes de chaînes en utilisant des tables de hachages, des arbres et puis donc des graphes.

Ensuite, nous avons développé des algorithmes calculant les plus courts chemins pour essayer de reconstituer ces chaines à partir de graphes.

|  |
| --- |
| II. LES STRUCTURES MANIPULÉES ET STRUCTURE DU CODE. |

## Structure du Code.

### 1. 1ère Partie du Projet.

Dans la première partie du code, nous avons décidé d’utiliser un fichier .c pour chaque structure donnée particulière utilisée, chacun des fichiers \*.c est accompagné d’un fichier \*.h référençant les fonctions utilisées et définissant les structures données utilisées.

* Arbre.c :

Les fonctions principales de ce fichier sont :

* ArbreQuat\* insererNoeudArbre(Noeud\* n, ArbreQuat\* a, ArbreQuat\* parent) ;
* Noeud\* chercherNoeudArbre(CellPoint\* pt, Reseau\* R, ArbreQuat\*\* aptr, ArbreQuat\* parent) ;
* Reseau\* recreeReseauArbre(Chaines\* C) ;
* Chaine.c :

Les fonctions principales de ce fichier sont :

* void lectureChaine( FILE \*f , Chaines\* chaines ) ;
* void ecrireChaineTxt(Chaines\* C,FILE \*f1) ;
* void afficheChaineSVG( Chaines \*C, char\* nomInstance ) ;
* Hachage.c :

-

* Reseau.c :

### 2. 2ème Partie du Projet.

Dans la deuxième partie du code, nous avons décidé d’utiliser un fichier \*.c pour chaque méthode de recherche, ces fichiers sont aussi accompagné d’un fichier \*.h :

* Djikstra.c :

Djisktra.c contient le code nécessaire à l’algorithme de Djikstra.

Il a pour fonctions principales :

* … : Cette fonction fait…
* Largeur.c :

Largeur.c contient le code nécessaire au parcours en largeur du graphe.

Ses fonctions principales sont :

- nbarcmin

## II. Structures de données utilisées.

|  |
| --- |
| III. JEUX D’ESSAIS |

## I. Chaînes :

La première version des jeux d’essai était sous la forme d’une liste de chaînes avec quelques caractéristiques précisées.

NbChain: 8

Gamma: 3

0 3 25.23 97.24 14.05 98.12 16.47 94.44

1 3 14.05 98.12 16.47 96.1 20.09 92.54

2 3 16.3 97.38 16.53 97.38 25.23 97.24

3 4 16.47 96.1 20.09 94.55 22.39 93.37 25.23 97.24

4 4 22.39 93.37 20.09 94.55 17.2 96.29 16.3 97.38

5 5 14.05 98.12 16.47 94.44 20.09 92.54 22.39 93.37 21.52 95.59

6 5 14.05 98.12 16.47 94.44 20.09 92.54 22.39 93.37 22 96.05

7 3 22.39 93.37 20.09 92.54 16.47 96.1

## Réseau :

NbNoeuds: 12

NbLiaison: 15

NbCommodite: 8

Gamma: 3

v 3 22.390000 93.370000

v 2 20.090000 92.540000

v 1 16.470000 96.100000

l 2 3

l 1 2

1 3

## Nouvelles Chaînes :

1 2 3 4 5 -1

6 7 -1

3 2 8 -1

3 4 9 10 30 13 14 15 16 -1

|  |
| --- |
| IV. ANALYSE DES JEUX D’ESSAIS |

## 1 - …

### 2…

### 2…

### 3…

## 2…

## 3…