

## SDD TP4 : TABLE DE HACHAGE

Nicolas Cavani & Adrien Leduque  
ZZ1 ISIMA  
enseignant : Yves Jean Daniel

Juin 2020

## Table des matières

1	Introduction	2
2	Makefile	2
3	Structure de données	3
4	Cas d'exécution	4

## 1 Introduction

L'objectif de ce TP est de coder une table de hachage indirecte permettant le stockage de mots lus dans un fichier texte. Le code permet de faire :

- Initialiser de la table de hachage
- Remplir la table de hachage depuis un texte lu dans un fichier
- Ajouter un élément dans la table de hachage
- Rechercher un élément dans la table
- Supprimer un élément
- Afficher la table de différentes manières
- Libérer la table de la mémoire

## 2 Makefile

Un Makefile est un fichier constitué de plusieurs règles de la forme :

cible : dépendances  
commandes

### 2.1 Cibles :

all : regroupe dans ses dépendances l'exécutable à produire.

clean : permet de supprimer tous les fichiers intermédiaires.

.PHONY : permet de reconstruire les dépendances de la cible (dépendante de .PHONY), dans le cas où des fichiers porteraient le même nom.

### 2.2 Variables :

CFLAGS : regroupe les options de compilation.

-Wall -Wextra : warnings de compilation

-g : Génère les informations de débogage.

-MMD : permet la génération des fichiers .d pour les dépendances.

SRC : contient la liste des fichiers sources du projet. La commande wildcard permet l'utilisation du joker \* dans la définition d'une variable.

OBJ : contient la liste des fichiers objets. OBJ est rempli à partir de SRC.

\$(patsubst pattern, replacement, text) : Trouve les mots dans text qui correspondent au pattern et les remplace par replacement.

DEP : fichiers .d contenant les dépendances associées à un fichier .o du même nom.

### 2.3 Variables internes :

\$@ : Le nom de la cible.

\$< : Le nom de la première dépendance.

\$^ : La liste des dépendances.

@ : Permet de ne pas afficher la commande dans la console.

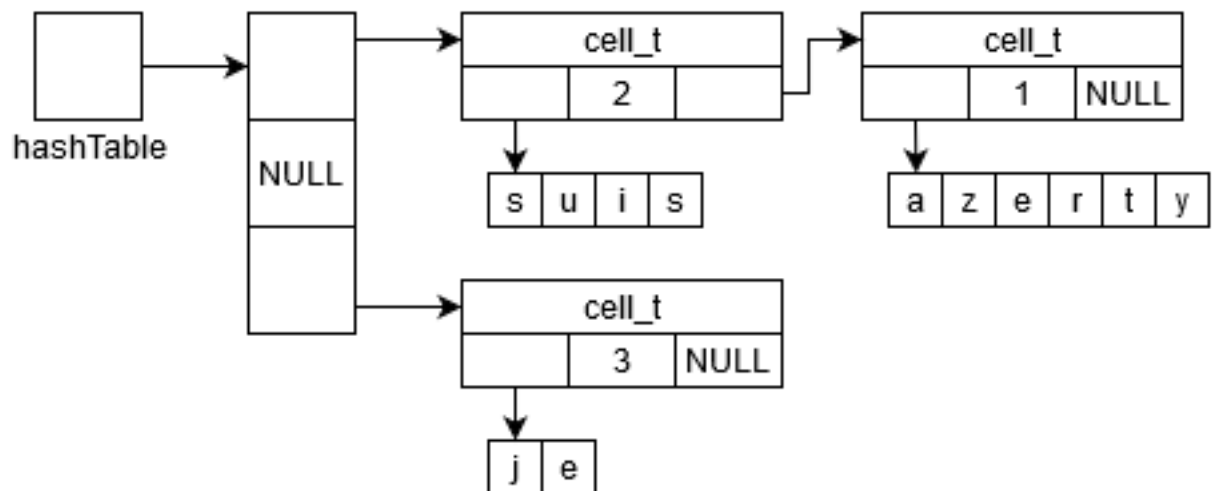
### 3 Structure de données

La table de hachage est un tableau contiguë, nommée table majeure, de taille HASH\_MAX contenant des pointeurs vers les sous tables. Les sous tables sont des listes chaînées de structures. Chaque structure contient :

- Le mot
- Le nombre d'apparitions du mot
- Un pointeur sur le mot suivant dans la liste chaînée de même clef de hachage

cell_t		
word	number	next
NULL	0	NULL

HASH\_MAX = 3

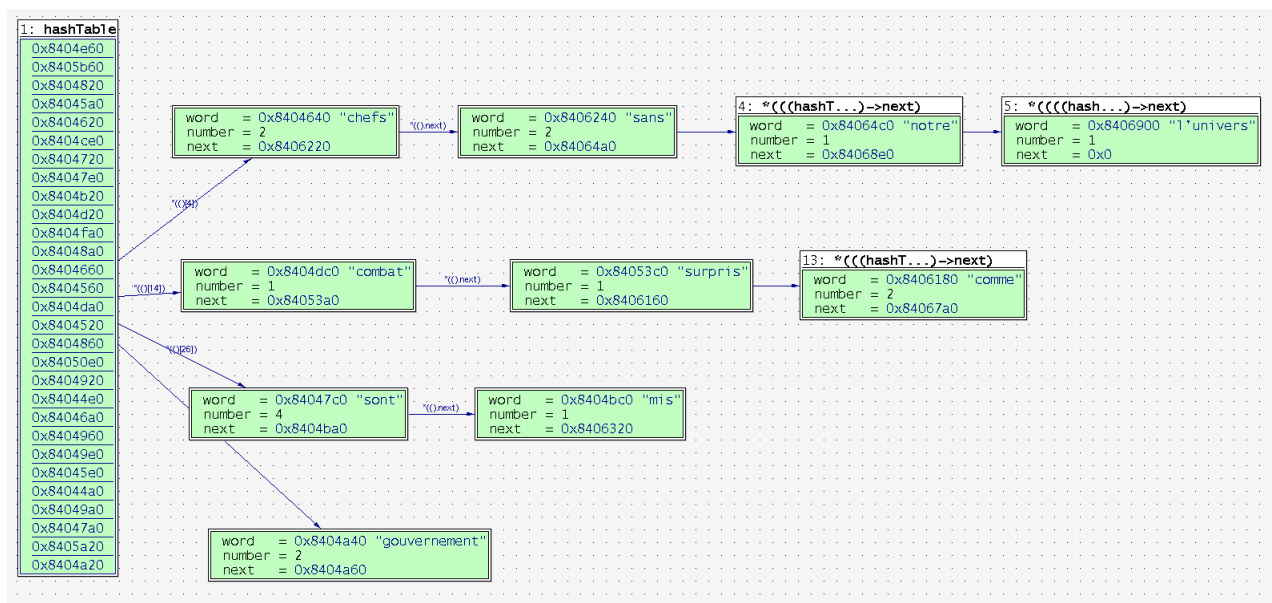


## 4 Cas d'exécution

### 4.1 Cas général : Long texte, plusieurs phrases, de nombreux mots, différents et identiques. Les 29 clefs sont utilisées et plusieurs mots ont la même clef.

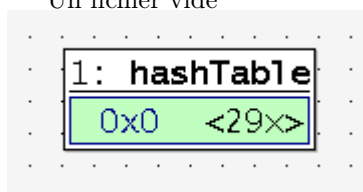
texte : Long texte.

- Présence de retour à la ligne
- Des phrases d'un mot
- Des phrases de plus d'un mot
- La table majeure est complètement utilisée
- Des mots sont répétés
- Des mots sont uniques
- Des mots différents ont la même clef



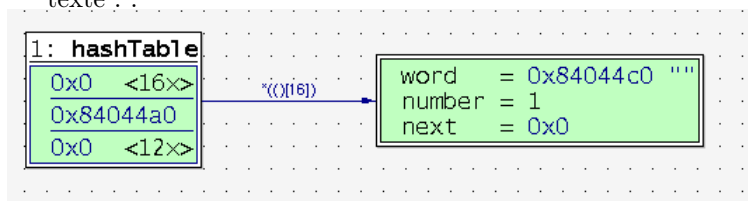
### 4.2 Texte vide

Un fichier vide



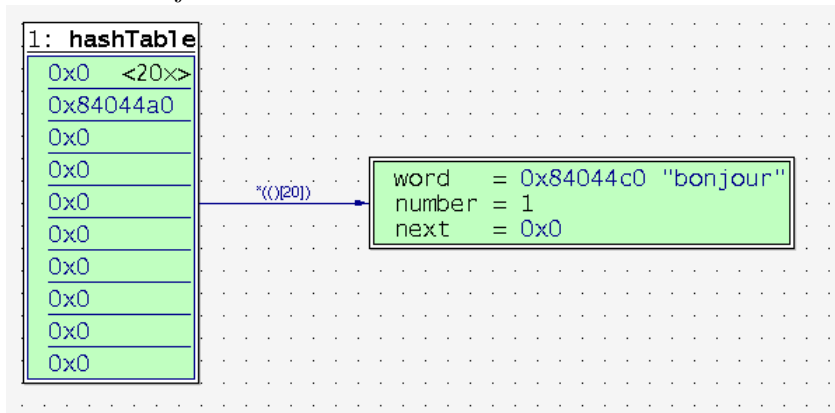
### 4.3 Une phrase sans mot

texte :



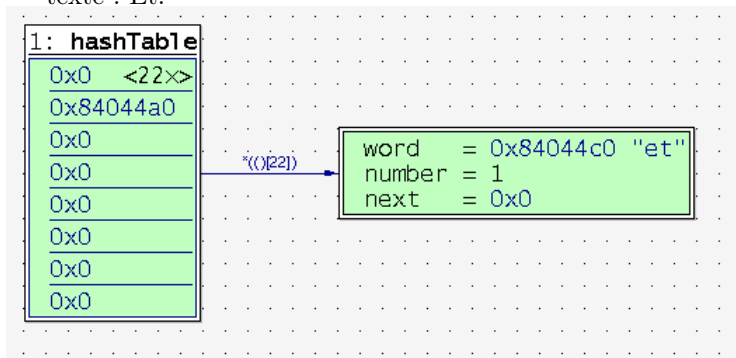
#### 4.4 Texte avec un unique mot sans point

texte : Bonjour



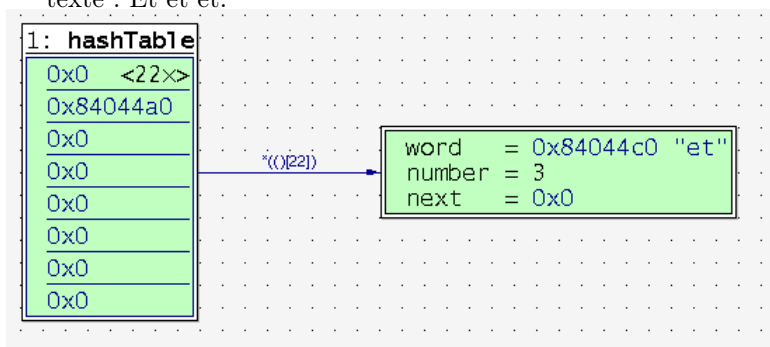
#### 4.5 Texte avec un unique mot avec un point

texte : Et.



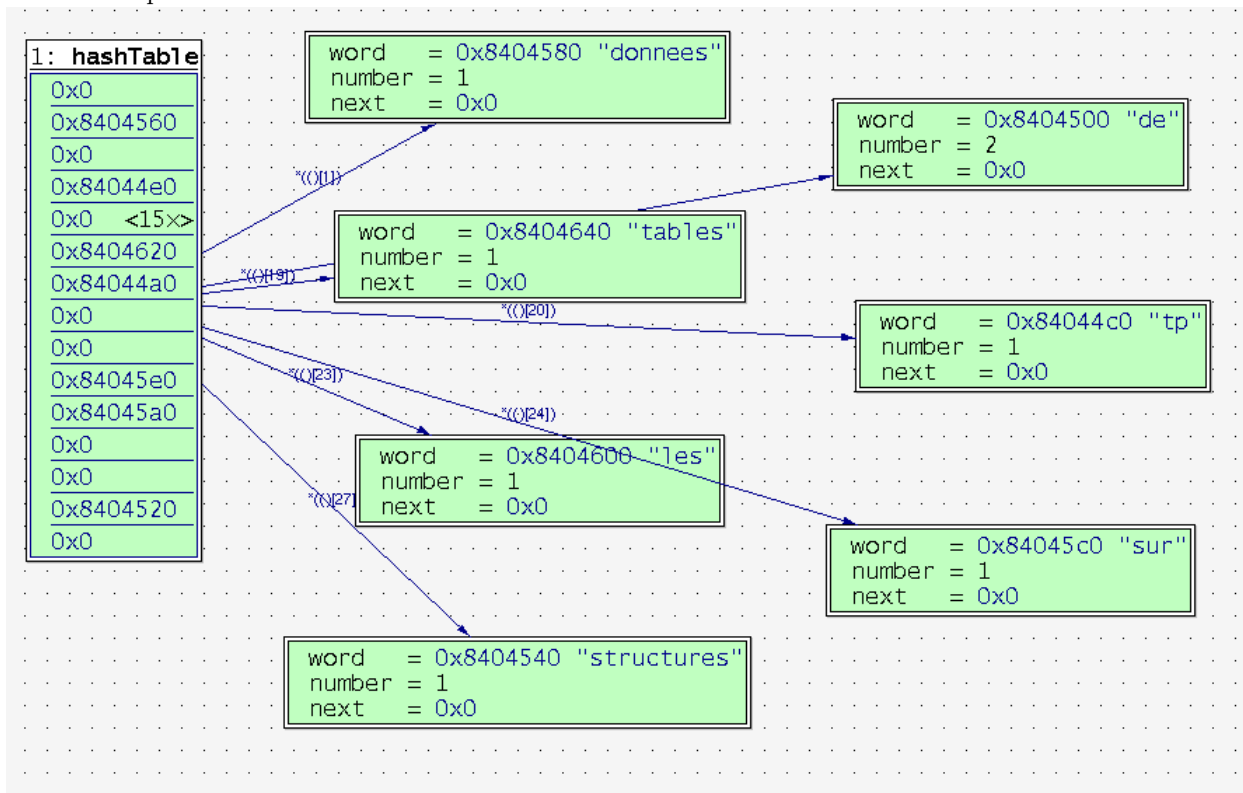
#### 4.6 Texte avec un unique mot répété

texte : Et et et.



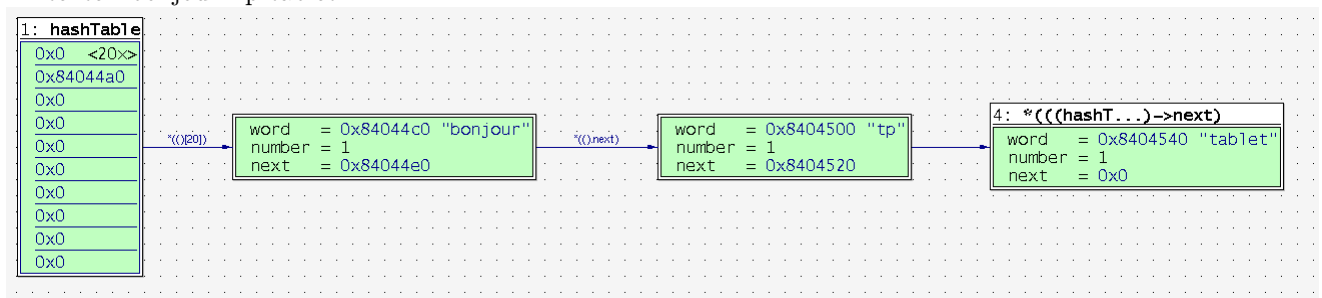
## 4.7 Une seule phrase

texte : Tp de structures de données sur les tables.



## 4.8 Plusieurs mots différents avec la même clef

texte : bonjour Tp tablet.



## 4.9 29 mots pour 29 clés différentes

texte : a b c d e f g h i j k l m n o p q r s t u v w x y z zn zo zp.

