*Le dossier doit débuter par un plan avec les numéros de pages !*

1. **Présentation :** ne doit pas excéder 5 lignes et doit donner l’objectif général du TP. Pas de description détaillée des structures par exemple, qui doit être dans la partie 2.

**Exemple :**

L’objectif de ce TP est de gérer une messagerie. Chaque message est stocké dans une liste chainée à partir d’un fichier des messages. Des modifications seront appliquées sur la liste (suppression des message obsolètes, modification des dates de début de certains messages…). La liste obtenue est sauvegardée dans un fichier au même format que le fichier en entrée.

1. **Description des structures de données et des fichiers**

Il faut donner pour chaque structure une description en français de la structure avec le type des champs, leur signification…et surtout donner les hypothèses sur ces structures (critère de tri, erreur possible ou non, …). Pas de langage C ici (pas de couper/coller des déclarations par exemple !).

Et surtout donner un exemple des structures !!

Ceci est valable pour les structures en mémoire (ici la liste chainée) et pour les fichiers de données en entrée et en sortie.

|  |
| --- |
| **Exemple :**   1. **Description de la liste chaînée des messages**   La liste chaînée des messages est une liste simplement chainée avec accès indirecte en tête. Chaque cellule de la liste est de la forme :   * Date de début de parution du message : entier sous la forme AAAAMMJJ (année, mois, jour) * Date de fin de parution du message : entier sous la forme AAAAMMJJ (année, mois, jour) * Message : chaîne de caractères d’au plus 100 caractères. Le message est ajusté à sa taille réelle afin de ne pas perdre de place.   *Donner un exemple simple de liste.*  **Hypothèse** : la liste est triée en ordre croissant sur la date de parution des messages.   1. **Description du fichier des messages en entrée** :   Ce fichier donne pour chaque message à faire paraitre une ligne par message de la forme :   * Une date de début de parution du message : entier sous la forme AAAAMMJJ (année, mois, jour) * Une date de fin de parution du message : entier sous la forme AAAAMMJJ (année, mois, jour) * Le message : chaîne de caractères d’au plus 100 caractères.   *Donner un exemple de ce fichier.*  **Hypothèses** : Les messages sont dans un ordre quelconque et le fichier est sans erreur.   1. **Description du fichier des messages en sotie** :   Ce fichier a le même format que le fichier en entrée. Comme il provient de la liste chainée, il sera trié sur la date de début de parution des messages.  *Donner un exemple de fichier.* |

1. **Organisation du code :**

Donner le découpage en modules avec les .c et .h et les dépendances entre modules (éventuellement faire un schéma des dépendances).

Les sous-programmes contenus dans les modules seront listés avec leur objectif en une phrase.

1. **Présentation du programme :**

Ne pas couper/coller chaque sous-programme dans le compte rendu, mais donner les programmes imprimés dans leur totalité. Chaque sous-programme sera décrit dans son en-tête avec le principe, le lexique des variables en entrée, en sortie et des variables de travail.

1. **Compte rendu d’exécution** :

* Donner en début la liste des cas testés.
* Pour chaque cas donner les entrées traitées : ici le fichier en entrée avec pour chaque ligne du fichier le cas testé (insertion en tête de la liste, insertion au milieu, insertion en fin, …). Puis donner le résultat obtenu par une capture d’écran de DDD donnant le contenu de la liste chainée obtenue par exemple.
* En cas de test de modification, il faut donner la situation avant modification, la modification testée et le résultat après modification.

1. Makefile et consignes pour exécuter le programme.