**Fiche technique**

**Les includes :**

stdio.h : Gérer les entrées/sorties du clavier (scanf)

stdlib.h : Gère l’affichage (printf)

string.h : Gestion des chaînes de caractères

time.h : Gestion du temps et de la fonction aléatoire (srand)

« afficheur.h » : Sert à l’intégration des fonctions externe

unistd.h : Sert à la gestion des processus

sys/types.h : Pour accéder aux informations des processus (pid\_t)

errno.h : Gestion des erreurs

**Constantes :**

\_MAX\_WIDTH : Largeur maximale de l’image de l’écran de veille statique

\_MAX\_HEIGHT : Hauteur maximale de l’image de l’écran de veille statique

\_SCREEN\_HEIGHT : Hauteur maximale de la console

\_SCREEN\_WIDTH : Largeur maximale de la console

\_TIME\_WIDTH : Largeur maximale de l’image de l’écran de veille dynamique

\_TIME\_HEIGHT : Hauteur maximale de l’image de l’écran de veille dynamique

\_MSG : Taille du message « Cet écran sera actualisé dans quelques secondes »

\_NBSEC : Nombre de secondes avant le rafraichissement de l’heure

Msg[] : variable globale

**Structures :**

imgPBM : Sert à stocker les données concernant les images (hauteur : largeur, tableau de pixels).

localTime : Stocker les informations de l’heure, minutes et les secondes.

**Fonctions :**

imgPBM getImgFromFile(char \*path) : retourne la structure imgPBM avec les informations du fichier donné en paramètre.

showImg(imgPBM img) : afficher l’image dans la console

« //void ascii() ; » : servais à afficher les caractères ascii.

printToXy(int x, int y, char text) : affiche un caractère aux coordonnées X et Y dans la console.

imgPBM getImgFromTime(int hour, int min, int sec) : retourne la structure imgPBM à partir de l’heure donné en paramètre.

imgPBM getImgChiffre(int value) : retourne la structure imgPBM du chiffre donné en paramètre.

localTime getTime() : retourne l’heure actuelle dans la structure localTime