Alexandre Dréan

Tadeusz Plagué-Makowiecki

Groupe 10

Projet IN104 :

Architecture du programme :

|  |
| --- |
| main( audio.mp3 )  {  int\* liste\_notes = [] ; int\* liste\_temps = [] ;  audio\_to\_piano\_notes( audio.mp3 ) -> liste\_notes, liste\_temps (modification en passage par adresse) ;  piano\_notes\_to\_video( liste\_notes, liste\_temps ) -> void (fenêtre) ;  free(liste\_notes,liste\_temps) ;  return 0 ;  } |

|  |
| --- |
| audio\_to\_piano\_notes( audio.wav )  {  ;  } |

|  |
| --- |
| piano\_notes\_to\_video( liste\_notes , liste\_temps)  {  Générer une fenêtre et un rendu en SDL  tant que (fenêtre non fermé et toutes les notes n’ont pas encore été joués)  {  t = temps écoulé depuis le début du programme  Si t est dans liste\_temps : faire une animation en SDL pour la note de piano associée ;  update le rendu SDL ;  }  }  L’animation est à détailler par la suite |

On travail d’abord sur une seule note pour simplifier le travail

|  |
| --- |
| Audio\_to\_piano\_une\_note( audio.wav )  // Utiliser le programme de l’université de nice pour lire le fichier audio.wav et enregistrer dans -> tab\_amplitudes, tab\_temps  // en fonction de la documentation de GSL, obtenir la tranformé de fourier du signal temporel entre t1 et t2  //parcourir les fréquences telles que X(f) > seuil et calculer log(produit(|X(if)­|²)) et retenir la fréquence qui maximise  retourner temps et notes (passage par adresse) |