برای شروع کار، فایلی که برای رکورد و سیو صدا قرار داده شد، استفاده گردید. این صدا به صورت آرایه ای با مقادیر 0 و 1 در متغیر data ذخیره میگردد. سپس با استفاده از تابع numpy.fft.fft() از این آرایه تبدیل فوریه ی گسسته گرفته میشود. سپس چون ضرایب تبدیل به فرم مختلط هستند، از این ضرایب absolute یا قدرمطلق گرفته میشود تا با اعداد حقیقی کار کنیم. همچنین به دلیل متقارن بودن تنها نیمه ابتدایی سیگنال را برای پردازش نگه میداریم. در نهایت با آزمون و خطا مشاهده گردید که همواره در فرکانس های ابتدایی فایل صوتی، یک نویز بسیار زیاد وجود دارد که این نویز برای جلوگیری از خطا، از سیگنال حذف کردید.

برای یافتن دو سینوسی مربوط به کلید فشرده شده، ابتدا یک بار بین ضرایب فوریه ماکزیمم میگیریم و اندیس آن ماکزیمم را نگه میداریم (این اندیس موید فرکانس زیرتر میباشد.). سپس آن ماکزیمم و اطرافش را (به همسایگی 50) صفر میکنیم و یک بار دیگر این کار را تکرار میکنیم تا فرکانس سینوسی دیگر را هم بدست آوریم.

حال این دو فرکانس را با 16 حالت موجود در جدول مقایسه میکنیم. برای مقایسه نیز اندکی تولرانس در نظر میگیریم تا اگر اندکی با فرکانس اصلی اختلاف داشت ignore نشود.

همچنین نکته ای که وجود دارد این است که چون در این روش از ماکزیمم همه مقادیر استفاده میشود، در تست ها مشاهده شد که وجود نویز بیش از حد بلند موجب بروز خطا بشود. ولی در نویز های معمولی و در صورتی که صدای کلید های فشار داده شده به اندازه کافی بلند باشد، برنامه کاملا درست نتیجه میدهد.

آرش مشیرنیا

95521432