SV cần làm các việc và báo cáo (nói trên slide và viết bằng file MS-Word) theo trình tự sau:

**1. Thu thập tín hiệu (signal acquisition):**

Nhóm SV dùng các tín hiệu đã được thu âm sẵn trong folder “TinHieuHuanLuyen” để thử nghiệm và viết báo cáo.

**2. Phân tích tín hiệu thủ công (manual signal analysis):**

Mỗi SV đo chu kỳ cơ bản T0 (fundamental period, **chính xác đến 1/10 milisec**ond) của đoạn tín hiệu bằng cách đo thủ công 1 chu kỳ nào đó của sóng tín hiệu (waveform). Tính nghịch đảo của T0 để thu được ước lượng (estimate) của tần số cơ bản F0 (fundamental frequency, theo đơn vị Hz) đo trên miền thời gian.

Gợi ý: dùng phần mềm WaveSurfer (Pitch Contour pane).

**3. Phân tích tín hiệu tự động (automatic signal analysis):**

Tìm hiểu lý thuyết và cài đặt 02 thuật toán tự động tính F0 trên miền thời gian dùng hàm tự tương quan (autocorrelation function) hoặc hàm vi sai biên độ trung bình (AMDF function).

Chú ý: Dải giá trị F0 của người trưởng thành: từ 70 đến 400 Hz (dùng dải này để thu hẹp phạm vi tìm kiếm F0, tránh các giá trị F0 “ảo” do khung tín hiệu gần như tuần hoàn nhưng ko phải tiếng nói).

**Các TLTK:**

[1] Luận văn cao học “Luận văn\_Tran Van Tam\_2019.pdf”: Chương 2 và 3.

[2] CS425 Audio and Speech Processing\_Hodgkinson\_2012: Chapter 3. Time-Domain pitch estimation

[3] textbook “Digital Speech\_Kondoz\_2004”: 6.2.1 Time-Domain PDAs (phần thuật toán AMDF).

[4] Cải tiến thuật toán tự tương quan tìm cao độ của tín hiệu đàn ghi-ta trên vi xử lý ARM Cortex-M4\_2017.

[5] A method for silence removal and segmentation of speech signals\_Giannakopoulos\_2014 (thuật toán dùng histogram).

**4. Yêu cầu:**

* Mỗi SV cài đặt và demo 01 thuật toán nêu ở trên, trong đó yêu cầu:

xuất hình vẽ kết quả hàm tự tương quan/AMDF của 2 khung tín hiệu (tự chọn để làm ví dụ minh hoạ) và nêu rõ vì sao hàm này giúp tìm F0:

* 1 khung chứa tiếng nói tuần hoàn (hữu thanh) 🡪 F0=? (Hz)
* 1 khung chứa tiếng nói không tuần hoàn (vô thanh) 🡪 F0 ko xác định.
* Chú ý khảo sát ảnh hưởng các tham số của thuật toán lên kết quả:
  + Ngưỡng xác định khung tín hiệu tuần hoàn/ko tuần hoàn.
  + Độ dài khung (20 ms vs. 30 ms)
* Các tín hiệu huấn luyện (training data) đóng vai trò giúp xác định bộ tham số tối ưu của thuật toán:

Muốn xác định ngưỡng V/U tin cậy dựa trên dữ liệu, SV cần khảo sát tất cả các voiced frames trong dữ liệu huấn luyện (timestamps lấy từ \*.lab) và thống kê xem biên độ của highest/lowest local peak/dip của normalized ACF/AMDF có phân bố dữ liệu ntn (meanV=?, stdV=?). Làm tương tự với các unvoiced frames để có (meanU=?, stdU=?). Dựa vào 2 phân bố dữ liệu này để tìm ra ngưỡng T giúp tách biệt 2 phân bố này (giả sử dữ liệu theo phân bố chuẩn).

Nếu ko dựa trên thống kê chỉ có cách làm mò mẫm (tự đưa ra 1 giá trị ngưỡng nào đó) --> ko thuyết phục và khó đúng với nhiều tín hiệu.

Ngoài ra, SV có thể áp dụng thuật toán tìm kiếm nhị phân trong [2] hoặc thuật toán dựa trên histogram trong [5] để tìm ngưỡng phân biệt.

* Hiệu suất của thuật toán đề xuất sẽ được kiểm chứng thực sự trên tập tín hiệu kiểm thử (test data) (GV sẽ upload sau 02 tuần nữa)
* Để tiết kiệm thời gian chấm thi, mỗi SV chạy script cài đặt riêng task của mình, duyệt qua 4 file tín hiệu kiểm thử và xuất ra 4 figure thể hiện input & output (mỗi figure cho 1 file tín hiệu) trong 01 lần chạy CT duy nhất để GV kiểm tra kết quả.
* Kết quả chuẩn của mỗi file tín hiệu \*.wav được chứa trong file \*.lab tương ứng. Định dạng của file .lab được mô tả trong file README. SV dùng dữ liệu chuẩn này để đưa ra đánh giá định lượng về độ chính xác của thuật toán như sau: viết script tính F0mean và F0std của các giá trị F0 tính được bằng thuật toán trên file \*.wav. Sau đó viết script tính độ lệch của 2 giá trị này so với F0mean và F0std chứa trong file \*.lab tương ứng. Để đưa ra đánh giá trực quan về độ chính xác của thuật toán, SV vẽ F0 contour của mỗi tín hiệu tương ứng về thời gian với tín hiệu.