

Лабораторная работа №6

Последовательность базовых заданий

1. Создайте класс *Equalizer*, который является наследником класса *handle*

| Номер | Свойства класса | Размер | Тип данных | Атрибуты свойств (Property Attributes) | Проверка свойств (validation functions) | Значение по умолчанию |
|-------|------------------|--------|------------|---|---|-----------------------|
| 1 | <i>freqArray</i> | 10x1 | double | Константа | Вещественные значения | Из 3 лаб.раб |
| 2 | <i>gain</i> | 10x1 | double | Доступны любым функциям | Вещественные значения | Единицы |
| 3 | <i>order</i> | 1x1 | double | Доступны к получению любыми функциями, к изменению только для членов класса и наследников | Целые значения | 64 |
| 4 | <i>bBank</i> | - | double | Доступно только для членов класса и наследников | Вещественные значения | нет |
| 7 | <i>fs</i> | 1x1 | double | Доступны к получению любыми функциями, к изменению только для членов класса и наследников | Вещественные значения | 44100 |
| 8 | <i>initB</i> | | double | Доступно только для членов класса и наследников | Вещественные значения | [] |

| Методы класса | Описание |
|------------------------|--|
| <i>Конструктор</i> | Создайте конструктор с двумя аргументами: <i>order</i> и <i>fS</i> . По входным параметрам необходимо рассчитывать и проинициализировать коэффициенты фильтров <i>bBank</i> |
| <i>Filtering</i> | Добавьте метод фильтрации использовав ветку <i>filter</i> из функции <i>FilteringBanks</i> |
| <i>CreateFilters</i> | Создайте метод аналогичный одноименной функции из 3 лабораторной работы |
| <i>GetFreqResponse</i> | Возвращает частотную характеристику итогового фильтра <i>Прим.</i> Используйте функцию <i>freqz</i> как в 4 лаб. р., в качестве коэффициентов $b = \text{sum}(\text{gain}.*b)$ <i>Прим.</i> Метод возвращает модуль H в дБ и w в соответствии с частотой дискретизации (см. 4 лаб. р.) |

- Откройте окно создания нового приложения *Apps*→*Design App*→*Blank App*.
Сохраните *EqualizerApp_SurnameN.mlapp*
- В появившемся окне задайте разрешение ширину и высоту окна 1024x768
- Добавьте на окно 4 кнопки (*Button*): *Load*, *Play*, *Pause*, *Stop*. Измените подписи и название объектов в соответствии с их назначением. По желанию можно изменить размер и цвет кнопки.
- Добавьте элемент *Slider*. Измените его положение на вертикальное, разместите подпись ровно по центру под элементом. Выставьте пределы от -10 до 10 с начальным значение равным 0.
- Измените подписи и имя объекта в соответствии с частотой, например: подпись: *31 Hz* (из *freqArray*), имя: *gain1*. Посредством копирования создайте 10 элементов *Slider* в соответствии с количеством фильтром.
- Добавьте на окно два элемента типа оси: *Spectrum* и *FreqResponse*. Подпишите оси, графики, по желанию можете дополнительно отредактировать объекты.
- Добавьте элемент *Label*, в котором будет отображаться имя mp3-файла.
- Перейдите на вкладку *Code View* и добавьте следующие свойства класса:

| Номер | Свойства класса | Размер | Тип данных | Атрибуты свойств (Property Attributes) | Проверка свойств (validation functions) | Значение по умолчанию |
|-------|---------------------|--------|----------------------------|---|---|---------------------------------|
| 1 | <i>equalizer</i> | - | <i>Equalizer</i> | Доступны любым функциям | - | Аргументы (64, 44100) |
| 2 | <i>deviceWriter</i> | - | <i>audioDeviceWriter</i> | Доступны любым функциям | - | Аргументы ('SampleRate', 44100) |
| 3 | <i>fileReader</i> | - | <i>dsp.AudioFileReader</i> | Доступны любым функциям | - | |
| 4 | <i>filename</i> | | | Доступно только для членов класса и наследников | - | |
| | <i>stopFlag</i> | 1x1 | | Доступно только для членов класса и наследников | Логические значения | false |

Примечание. Объекты *deviceWriter* и *fileReader* создайте по аналогии с лаб.р. 3.

10. Создайте метод класса *app PlotFreqResponse()*, который вызывает метод *GetFreqResponse()* класса *Equalizer* и строит результат на осях *FreqResponse*.
11. Создайте *Callback startupFcn*, в которой вызывается метод *PlotFreqResponse()*. Функция *startupFcn* вызывается после создания объекта *app*.
12. Добавьте *Callback* для слайдеров. Изменение слайдеров меняет соответствующий элемент в массиве *gain* (первый слайдер – *gain(1)* и тд.). Также в каждом *Callback* необходимо вызвать метод *PlotFreqResponse()*.

13. Добавьте *Callback* для кнопки *Load*. Используя функцию *uigetfile* проинициализируйте *filename*. Используя новый *filename* заново проинициализируйте *fileReader*, также изменив свойство *PlayCount* на значение *inf* и свойство *SamplesPerFrame* на значение 4096. Измените содержимое *Label* (поле *Text*) на содержимое *filename*.
14. Добавьте *Callback* для кнопки *Run*. Используйте бесконечный цикл наподобие того, что использовался в 3 лаб. р., добавив дополнительный выход через проверку переменной *stopFlag*. Внутри цикла фильтрация осуществляется при помощи метода *Filtering* объекта *equalizer*.
В цикле также необходимо вычислить спектр используя *pspectrum* и отображать результат на осях *Spectrum*. По оси *x* используйте логарифмический масштаб. По оси *y* можно отображать также дБ (или оставить по умолчанию).
Примечание. Также всегда необходимо при нажатии кнопки *Run* делать *stopFlag = false*.
15. Добавьте *Callback* для кнопки *Pause*, в котором выполняется: *stopFlag = true*;
16. Добавьте *Callback* для кнопки *Stop*: *stopFlag = false*, а также используйте функцию *reset* для перезагрузки объектов *deviceWriter* и *fileReader*.
17. В конце каждого метода и *callback*-ов добавьте *drawnow* (в том числе в теле бесконечного цикла *callback* кнопки *Run*)

Дополнительные задания

1. Отформатируйте формочку, сделав собственное визуальное оформление.