2020년도 2학기 컴퓨터공학설계및실험Ⅰ

10주차 openFrameWorks

20161663 허재성

1. 실습 목적

openFrameWorks에 대해 이해하고, 여러 예제와 Waterfall 프로그램을 구현하며 그 사용법을 익힌다.

2. 관련 이론

2-1. openFrameWorks에 포함된 각각의 라이브러리에 대해 자세히 정리하시오.

openFrameWorks는 범용적인 목적으로 작동되도록 설계되었으므로, 다음과 같이 일반적으로 사용되는 라이브러리들을 포함하고 있다.[1] 사용 분야별로 나누어보면 다음과 같다.

Graphics

OpenGL : Open Graphics Library(줄여서 OpenGL)는 [실리콘 그래픽스](https://ko.wikipedia.org/wiki/%EC%8B%A4%EB%A6%AC%EC%BD%98_%EA%B7%B8%EB%9E%98%ED%94%BD%EC%8A%A4)(SGI)에서 만든 2차원, 3차원 그래픽스 표준[API](https://ko.wikipedia.org/wiki/API)로, 250여개의 함수를 호출하여 2차원의 단순한 기하 도형부터 복잡한 3차원 장면을 생성할 수 있다. OpenGL은 [CAD](https://ko.wikipedia.org/wiki/CAD), [가상현실](https://ko.wikipedia.org/wiki/%EA%B0%80%EC%83%81%ED%98%84%EC%8B%A4), [정보 시각화](https://ko.wikipedia.org/w/index.php?title=%EC%A0%95%EB%B3%B4%EC%8B%9C%EA%B0%81%ED%99%94&action=edit&redlink=1), 비행 시뮬레이션 등 컴퓨터 그래픽이 필요한 여러 분야에서 활용되고 있다. 또한 컴퓨터 게임 분야에서도 널리 활용되고 있으며, [마이크로소프트](https://ko.wikipedia.org/wiki/%EB%A7%88%EC%9D%B4%ED%81%AC%EB%A1%9C%EC%86%8C%ED%94%84%ED%8A%B8)사의 [Direct3D](https://ko.wikipedia.org/wiki/%EB%8B%A4%EC%9D%B4%EB%A0%89%ED%8A%B83D)와 함께 컴퓨터 그래픽 세계를 양분하고 있다.[2]

GLEW : OpenGL Extension Wrangler Library (줄여서 GLEW)는 OpenGL 확장을 쿼리(query, 질의)하고 로드(load)하는 데 도움이 되는 교차 플랫폼 C/C++ 라이브러리다. GLEW는 대상 플랫폼(운영체제, Windows, Linux, macOS 등)에 따라 지원되는 OpenGL 확장을 결정하기 위한 효율적인 런타임 메커니즘을 제공한다. 모든 OpenGL 확장은 단일 헤더 파일로 노출되며, 이는 공식 확장명 목록에서 기계적으로 생성된다. GLEW는 Windows, Linux, Mac OS X, FreeBSD, IRIX 및 Solaris를 포함한 다양한 운영 체제에서 사용할 수 있다.[3]

GLUT : OpenGL Utility Toolkit(줄여서 GLUT)은 호스트 운영체제와 시스템 수준의 입출력을 가능하게 만드는 [OpenGL](https://ko.wikipedia.org/wiki/OpenGL) 프로그램용 유틸리티 라이브러리이다. 주기능으로 창(window)의 크기와 형태를 정의하고 제어하며, 키보드와 마우스 입력을 감지하는 기능이 있다. [정육각형](https://ko.wikipedia.org/wiki/%EC%A0%95%EC%9C%A1%EA%B0%81%ED%98%95), [구](https://ko.wikipedia.org/wiki/%EA%B5%AC_(%EA%B8%B0%ED%95%98%ED%95%99)), [유타 찻잔](https://ko.wikipedia.org/w/index.php?title=%EC%9C%A0%ED%83%80_%EC%B0%BB%EC%9E%94&action=edit&redlink=1)과 같은 컴퓨터 그래픽스에서 요긴한 몇 가지 기하학적인 기본객체(geometric primitives)를 그려주는 기능(routine)을 가지고 있으며, 팝업 메뉴를 생성하는 기능도 제공한다. GLUT를 사용하면 서로 다른 운영체제 사이의 이식성이 좋은 코드를 구현할 수 있고, OpenGL을 쉽게 익힐 수 있다. 또한 코드의 양을 줄일 수 있고 운영체제에 대한 지식이 부족해도 프로그램을 작성할 수 있다. GLUT에서 제공하는 함수는 glutPostRedisplay, glutCreateWindow와 같이glut라는 접두사로 시작한다.[4]

libtess2 : libtess의 리팩터링(결과의 변경은 없이 코드의 구조를 재조정한 것) 버전으로, libtess는SGI의 OpenGL Utility(GLU)의 참조 구현에 있는 테셀레이터(tesselator)에서 포팅된 다각형 테셀레이션(Polygon Tesselation, 다각형을 서로 겹치지 않게, 빈틈없이 모으는 것, 쪽매맞춤) 라이브러이이다.[5]

cairo : cairo는 다중 출력 장치를 지원하는 2D 그래픽 라이브러리다. 현재 지원되는 출력 대상에는 X Window 시스템(Xlib 및 XCB), Quartz, Win32, 이미지 버퍼, PostScript, PDF, SVG 파일 출력 등이 포함된다. 실험 백엔드로는 OpenGL, BeOS, OS/2 및 DirectFB가 있다.  
c airo는 모든 출력 매체에서 일관된 출력을 생성하는 동시에 디스플레이 하드웨어 가속(예: X 렌더 익스텐션을 통한)을 이용하도록 설계되었다.  
cairo API는 PostScript 및 PDF의 도면 운영자와 유사한 작업을 제공한다. cairo로 가능한 작업에는 stroking, filling cubic Bézier splines, 반투명 영상 변환 및 합성, 항염색 텍스트 렌더링 등이 포함된다. 모든 도면 연산은 모든 부속 변형(척도, 회전, 전단 등)에 의해 변형될 수 있다.  
cairo는 C 프로그래밍 언어로 작성된 라이브러리로 구현되지만, 바인딩은 몇 가지 다른 프로그래밍 언어에 사용할 수 있다.[6]

Audio input, output and analysis

RtAudio : RtAudio는 Windows, macOS, Linux운영체제에서 실시간 오디오 입출력용 API를 제공하는 C++ 클래스 세트다. RtAudio는 컴퓨터 오디오 하드웨어와 상호작용하는 과정을 상당히 단순화한다. RtAudio는 C++로 객체 지향적으로 설계되어 있으며, 지원하는 모든 플랫폼에서 이용 가능한 단순하고 공통적인 API를 제공한다. 다른 프로그램 프로젝트에 쉽게 포함 가능하도록 하나의 소스와 하나의 헤더 파일로 이루어져 있다. 장치의 동적 연결을 가능하게 하며, 광범위한 오디오 장치 매개 변수 제어를 제공한다. 또한 오디오 장치 기능을 탐색하는 것을 허용하고 데이터 형식, 채널 번호 보정, 인터리빙 및 제거, 바이트 스와핑을 위한 자동 내부 변환이 가능하다.  
RtAudio에는 오디오 출력(재생) 및/또는 입력(녹음)을 나타내는 오디오 스트림의 개념이 통합되어 있다. 사용 가능한 오디오 장치와 그 기능을 열거한 다음 스트림을 열 때 지정할 수 있다. 해당되는 경우, 복수의 API 지원을 컴파일할 수 있으며, RtAudio 인스턴스를 생성할 때 특정 API를 지정한다.[7]

PortAudio : PortAudio는 무료 cross platform, 오픈 소스, 오디오 I/O 라이브러리이다. Windows, macOS, 유닉스(OSS/ALSA)를 포함한 많은 플랫폼에서 C'나 C++로 간단한 오디오 프로그램을 작성하여 컴파일 및 실행할 수 있다. 서로 다른 플랫폼에서 개발자 간 오디오 소프트웨어 교환을 촉진하기 위해 개발되었으며, 많은 애플리케이션이 오디오 I/O용 PortAudio를 사용한다.  
P ortAudio는 간단한 콜백 기능이나 차단된 읽기/쓰기 인터페이스를 사용하여 소리를 녹음하거나 재생할 수 있는 매우 간단한 API를 제공한다. 사인(sine)파 재생, 오디오 입력 처리(guitar fuzz), 오디오 녹음 및 재생, 사용 가능한 오디오 장치 나열 등의 프로그램이 예시 프로그램에 포함된다.[8]

OpenAL : OpenAL(Open Audio Library, 오픈 오디오 라이브러리)은 무료 크로스 플랫폼 오디오 [API](https://ko.wikipedia.org/wiki/API)이다. 다중 채널 3차원 오디오 출력을 효율적으로 구현해내기 위해 설계되었다. API의 스타일과 만들어진 경위는 [OpenGL](https://ko.wikipedia.org/wiki/OpenGL)과 비슷하다.[9] OpenAL은 게임 어플리케이션 및 기타 많은 유형의 오디오 어플리케이션과 함께 사용하기에 적합한 크로스 플랫폼 3D 오디오 API이다.  
 OpenAL은 3D 공간에서 움직이는, 한 명의 청취자가 그 공간 어딘가에서 들을 수 있는 오디오 소스 모음집을 모델링한다 기본 OpenAL 객체는 Listener, Source 및 Buffer이다. 오디오 데이터를 포함하는 버퍼 수가 많을 수 있다. 각 버퍼를 하나 이상의 소스에 부착할 수 있으며, 소스는 오디오를 방출하는 3D 공간의 점을 나타낸다. 오디오 컨텍스트마다 항상 하나의 수신기 객체가 존재하며, 이는 소스가 청취되는 위치를 나타낸다. 렌더링은 수신기의 관점에서 수행된다.[10]

Kiss FFT : Kiss FFT는 작고 간단한 FFT(고속 푸리에 변환)[11] 라이브러리이다. 복잡한 특정 플랫폼 별 최적화가 없으며 재컴파일만으로도 고정 소수점 또는 부동 소수점을 다룰 수 있다.[12]

FMOD : FMOD는 파이어라이트 테크놀로지스에서 만든 음향 [미들웨어](https://ko.wikipedia.org/wiki/%EB%AF%B8%EB%93%A4%EC%9B%A8%EC%96%B4)이다. 버전 3.75 이후로 FMOD는 3가지 라이브러리와 프로그램으로 구성되어 있다. 먼저 FMOD Ex는 저수준 사운드 엔진이다. 두 번째로 FMOD 이벤트 시스템(FMOD Event System)는 FMOD 디자이너로 만들어진 추상적인 사운드 컨텐츠를 이용하여 데이터 기반 프로그래밍을 간단하게 할 수 있도록 도와준다. 마지막으로 FMOD 디자이너(FMOD Designer)는 사운드 디자이너를 위한 툴로 음악의 재생과 복잡한 사운드 이벤트를 만들어 낼 수 있다. FMOD는 C, C++, C#으로 개발되었으며, Windows, macOS, Linux, iOS, Android 등의 환경에서 지원된다.[13]

Fonts

Free Type : FreeType은 [C 언어](https://ko.wikipedia.org/wiki/C_%EC%96%B8%EC%96%B4)로 작성된 소프트웨어 라이브러리로 [글꼴 래스터라이제이션](https://ko.wikipedia.org/w/index.php?title=%EA%B8%80%EA%BC%B4_%EB%9E%98%EC%8A%A4%ED%84%B0%EB%9D%BC%EC%9D%B4%EC%A0%9C%EC%9D%B4%EC%85%98&action=edit&redlink=1) 엔진을 제공한다. 래스터라이제이션(Rasterisation)은 컴퓨터 과학에서 벡터 그래픽 이미지를 비디오 디스플레이나 프린터 등의 래스터 디바이스에 출력하기 위해 래스터 이미지로 변환하는 것을 의미한다.[14] 단순하면서도 사용하기 쉽고 통일된 UI를 제공하여 [글꼴](https://ko.wikipedia.org/wiki/%EA%B8%80%EA%BC%B4) 파일의 내용에 접근할 수 있으므로 이러한 작업들을 매우 단순하게 만들어 준다. Free Type라이브러리는 영리 목적이든 아니든 간에 어떠한 종류의 프로젝트에서도 사용할 수 있다. 따라서 Free Type은높은 품질의 텍스트를 요구하는 현대의 [비디오 게임](https://ko.wikipedia.org/wiki/%EB%B9%84%EB%94%94%EC%98%A4_%EA%B2%8C%EC%9E%84)을 위한 래스터라이제이션 엔진으로 널리 채용되고 있다.[15]

Image Saving and Loading

FreeImage : FreeImage는 PNG, BMP, JPEG, TIFF 와 같은 인기 있는 그래픽 이미지 형식을 멀티미디어 어플리케이션에서 지원하고자 하는 개발자들을 위한 오픈 소스 라이브러리 프로젝트다. FreeImage는 사용하기 쉽고, 빠르고, 멀티스레딩이 안전하며, 모든 32비트 또는 64비트 버전의 Windows, Linux, macOS와 호환된다. ANSI C 인터페이스 덕분에 FreeImage는 C, C++, VB, C#, Delphi, Java 등 다양한 언어와 Perl, Python, PHP, TCL 또는 Ruby와 같은 공통 스크립트 언어에서도 사용할 수 있다.  
라이브러리는 WIN32 또는 WIN64 C/C++ 컴파일러에 대해 연결할 수 있는 이진 DLL 배포와 소스 배포의 두 가지 버전으로 제공된다. 마이크로소프트 VS용 워크스페이스 파일.Net 2013 및 VS.넷 2017이 제공되며 리눅스, 민GW, 맥 OS X용 makefile도 제공된다.[16]

Video playback and grabbing

Quicktime : QuickTime은 [애플](https://ko.wikipedia.org/wiki/%EC%95%A0%ED%94%8C)이 개발한 멀티미디어 프레임워크로 [디지털 영상](https://ko.wikipedia.org/wiki/%EB%94%94%EC%A7%80%ED%84%B8_%EC%98%81%EC%83%81), [미디어 클립](https://ko.wikipedia.org/w/index.php?title=%EB%AF%B8%EB%94%94%EC%96%B4_%ED%81%B4%EB%A6%BD&action=edit&redlink=1), 소리, 텍스트, 애니메이션, 음악, 몇 가지 종류의 [퀵타임 VR](https://ko.wikipedia.org/w/index.php?title=%ED%80%B5%ED%83%80%EC%9E%84_VR&action=edit&redlink=1)을 다룰 수 있다. [클래식 맥 OS](https://ko.wikipedia.org/wiki/%EB%A7%A5_OS), [OS X](https://ko.wikipedia.org/wiki/OS_X), Microsoft Windows 운영 체제를 지원하며, [아이튠스](https://ko.wikipedia.org/wiki/%EC%95%84%EC%9D%B4%ED%8A%A0%EC%8A%A4), QuickTime Player를 포함한 소프트웨어 꾸러미에서 필수적인 지원을 제공한다.[17]

GStreamer : GStreamer는 [GObject](https://ko.wikipedia.org/wiki/GObject) 기반 [자료형 체계](https://ko.wikipedia.org/wiki/%EC%9E%90%EB%A3%8C%ED%98%95_%EC%B2%B4%EA%B3%84)와 더불어 [C 프로그래밍 언어](https://ko.wikipedia.org/wiki/C_%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D_%EC%96%B8%EC%96%B4)로 작성한 [파이프라인](https://ko.wikipedia.org/wiki/%ED%8C%8C%EC%9D%B4%ED%94%84%EB%9D%BC%EC%9D%B8) 기반 [멀티미디어 프레임워크](https://ko.wikipedia.org/wiki/%EB%A9%80%ED%8B%B0%EB%AF%B8%EB%94%94%EC%96%B4_%ED%94%84%EB%A0%88%EC%9E%84%EC%9B%8C%ED%81%AC)이다. GStreamer는 단순 [오디오](https://ko.wikipedia.org/w/index.php?title=%EC%98%A4%EB%94%94%EC%98%A4_%EC%A3%BC%ED%8C%8C%EC%88%98&action=edit&redlink=1) 재생, 오디오 및 비디오 재생, [녹음](https://ko.wikipedia.org/wiki/%EB%85%B9%EC%9D%8C), [스트리밍](https://ko.wikipedia.org/wiki/%EC%8A%A4%ED%8A%B8%EB%A6%AC%EB%B0%8D) 및 편집을 비롯하여 프로그래머가 다양한 미디어 관리 구성 요소를 만들 수 있게 해 준다. 파이프라인 디자인은 [영상 편집기](https://ko.wikipedia.org/wiki/%EC%98%81%EC%83%81_%ED%8E%B8%EC%A7%91), 스트리밍 미디어 브로드캐스터, [미디어 플레이어](https://ko.wikipedia.org/wiki/%EB%AF%B8%EB%94%94%EC%96%B4_%ED%94%8C%EB%A0%88%EC%9D%B4%EC%96%B4)와 같은 수많은 종류의 [멀티미디어](https://ko.wikipedia.org/wiki/%EB%A9%80%ED%8B%B0%EB%AF%B8%EB%94%94%EC%96%B4) 응용 프로그램을 만드는 역할을 한다. [18]

videoInput : Windows의 무료 동영상 캡쳐 라이브러리이다. 추가적인 소프트웨어를 설치할 필요 없으며, 세팅을 위한 팝업이 뜨지 않는다. 동시에 여러 캡처 디바이스에서 캡처를 할 수 있으며, 여러 영상 캡처 디바이스를 지원하다. 또한 속도도 빠르다.[19]

Variety of utilities

Poco : POCO C++ 라이브러리는 강력한 크로스 플랫폼 C++ 라이브러리로 데스크톱, 서버, 모바일 기기, IoT, 임베디드 시스템에서 실행되는 네트워크와 인터넷 기반 애플리케이션을 위한 라이브러리이다.[20]

Computer Vision

OpenCV : OpenCV(Open Source Computer Vision)은 인텔에서 개발한 실시간 [컴퓨터 비전](https://ko.wikipedia.org/wiki/%EC%BB%B4%ED%93%A8%ED%84%B0_%EB%B9%84%EC%A0%84)(물체를 인식하는 능력)을 목적으로 한 프로그래밍 [라이브러리](https://ko.wikipedia.org/wiki/%EB%9D%BC%EC%9D%B4%EB%B8%8C%EB%9F%AC%EB%A6%AC)로 실시간 이미지 처리에 중점을 두었다. [인텔](https://ko.wikipedia.org/wiki/%EC%9D%B8%ED%85%94) [CPU](https://ko.wikipedia.org/wiki/%EC%A4%91%EC%95%99_%EC%B2%98%EB%A6%AC_%EC%9E%A5%EC%B9%98)에서 사용되는 경우 속도의 향상을 볼 수 있는 [IPP](https://ko.wikipedia.org/wiki/IPP)(Intel Performance Primitives)를 지원한다. OpenCV는 Windows, Linux 등에서 사용 가능한 [크로스 플랫폼](https://ko.wikipedia.org/wiki/%ED%81%AC%EB%A1%9C%EC%8A%A4_%ED%94%8C%EB%9E%AB%ED%8F%BC)이며 [오픈소스](https://ko.wikipedia.org/wiki/%EC%98%A4%ED%94%88%EC%86%8C%EC%8A%A4) [BSD 허가서](https://ko.wikipedia.org/wiki/BSD_%ED%97%88%EA%B0%80%EC%84%9C) 하에서 무료로 사용할 수 있다. OpenCV는 [TensorFlow](https://ko.wikipedia.org/wiki/TensorFlow) , [Torch](https://ko.wikipedia.org/wiki/Torch) / [PyTorch](https://ko.wikipedia.org/wiki/PyTorch) 및 [Caffe](https://ko.wikipedia.org/w/index.php?title=Caffe&action=edit&redlink=1)의 [딥러닝](https://ko.wikipedia.org/wiki/%EB%94%A5%EB%9F%AC%EB%8B%9D) [프레임워크](https://ko.wikipedia.org/wiki/%ED%94%84%EB%A0%88%EC%9E%84%EC%9B%8C%ED%81%AC)를 지원한다.[21]

3D model loading

Assimp : Open Asset Import Library(줄여서 Assimp)는 서로 다른 3D Asset File에 대한 공통적인 파일 형식에 대한 공통 API를 제공하는 cross-platform 3D 모델 가져오기(import) 라이브러리이다. C++로 쓰여져 있으며 C와 C++을 둘 다를 위한 인터페이스를 제공한다. C#, Python과의 바인딩도 프로젝트의 일부로 개발되었거나 다른 곳에서 사용 가능하다. Assimp에서 가져온 데이터는 간단한 계층적 자료 구조로 제공되며 구성 가능한 사후 처리 단계는 기능 세트를 증가시킨다.[22]

2-2. openFrameWorks로 만든 프로그램/프로젝트에는 어떤 것들이 있는지 하나 이상 소개하시오.

openFrameWorks는 C++를 기반으로 한 오픈 소스 프레임워크로서 창의적인 코딩을 위하여 디자인된 프레임워크이다. 기존의 라이브러리를 통합하여 손쉽게 이용 가능하도록 설계되어 있다. 그래픽에는 OpenGL이, 오디오 작동에는 rtAudio가, 글꼴에는 Free Type, 이미지 작업에는 FreeImage, 비디오 재생에는 QuickTime이 사용된다.[23]

openFrameWorks로 만들어진 프로그램은 openFrameWorks 공식 웹사이트에서 예시로 든 프로그램들이 있다. 첫 번째는 Puppet Parade로 Emily Gobeille와 Theo Watson이 만들었다. Puppet Pareade는 대화형 구조물로 어린이들이 그들의 팔을 이용해 그들 앞에 있는 벽에 있는 커다란 인형을 조종할 수 있다. 어린이들이 인형 옆에 있는 것처럼 인형을 조종할 수 있게 한다.

또 다른 예로는 Interactive Wall at UD로 University of Dayton(UD) 입학처에 있는 36피트 벽이다. 이 벽은 끊임없이 변화하는 그래픽 패턴이 생성되어 표시되는데 벽 앞에 사람의 존재에 따라 반응하기도 한다.[24]

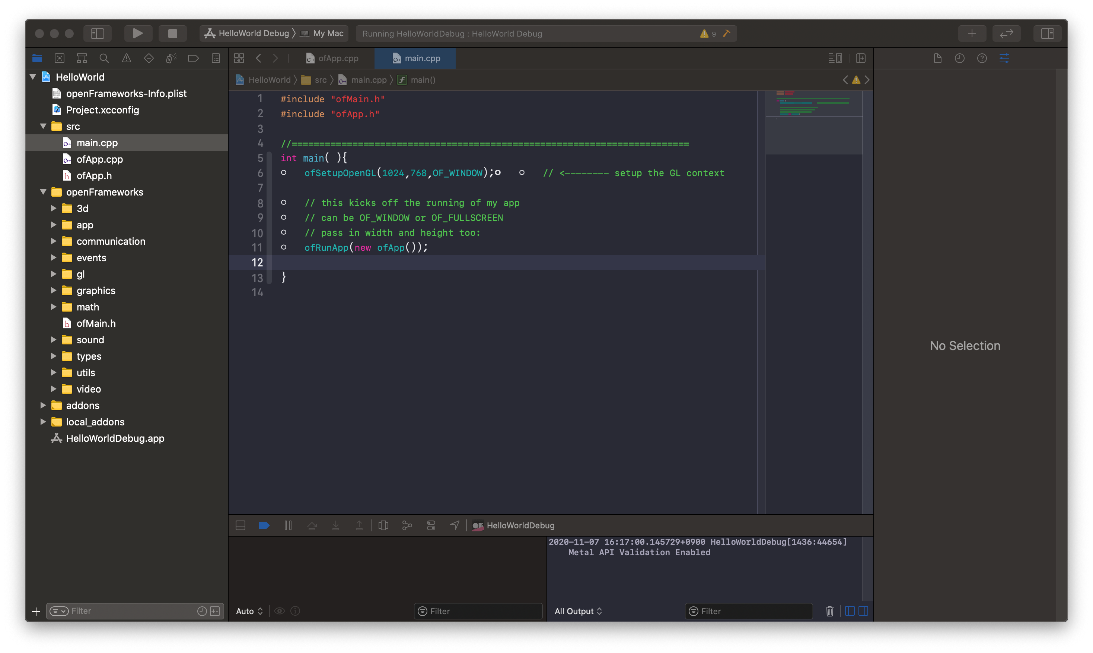
위에서 예시로 든 프로젝트들은 openFrameWorks로 제작 가능한 범주를 보여준다.

2-3. openFrameWorks를 다운 받아 Hello World!를 출력하는 프로그램을 간단하게 작성해본 뒤, 실행 결과를 스크린샷으로 첨부하시오.

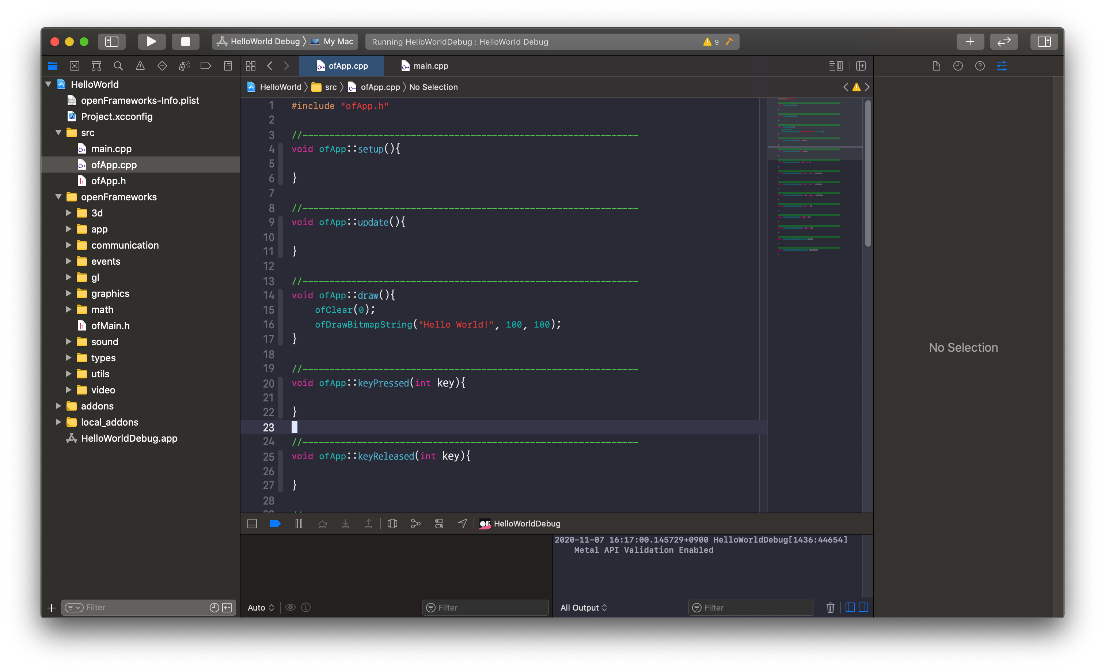
강의 자료를 따라서 openFramework에서 제공하는 프레임워크를 다운받고 Xcode IDE로 openFrameworks 프로젝트를 만든다. 해당 프로젝트 이름은 HelloWord라고 했다.

프로젝트를 만들면 src 폴더에 기본적으로 main.cpp, ofApp.cpp 소스 파일과 ofApp.h 헤더파일이 있다.

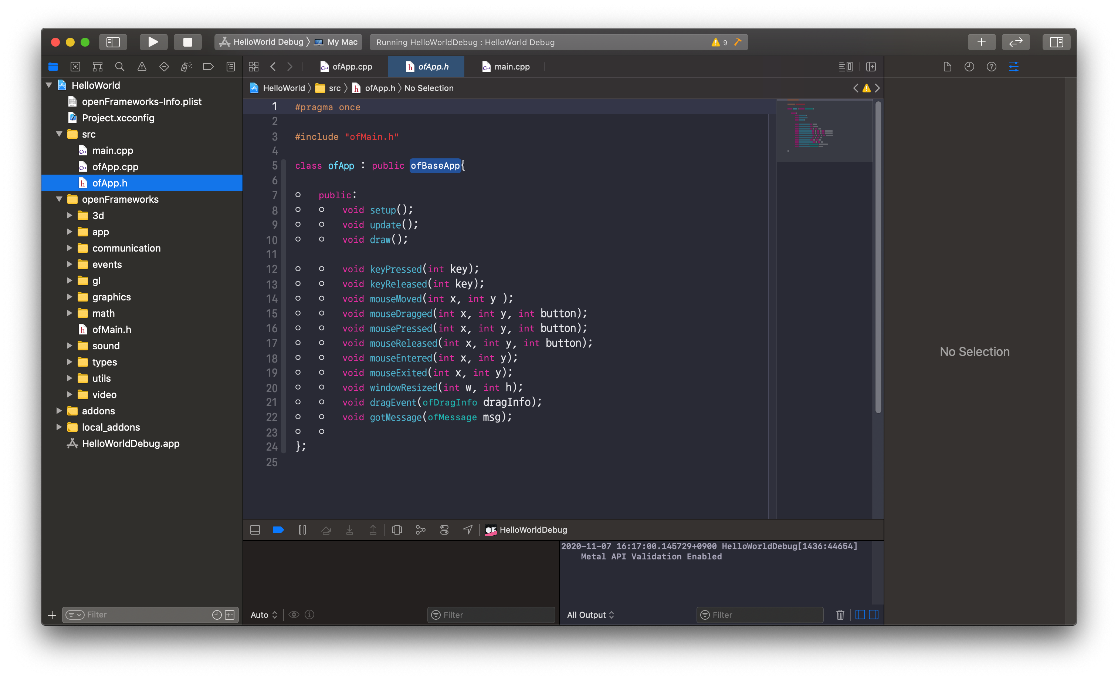
main.cpp



ofApp.cpp



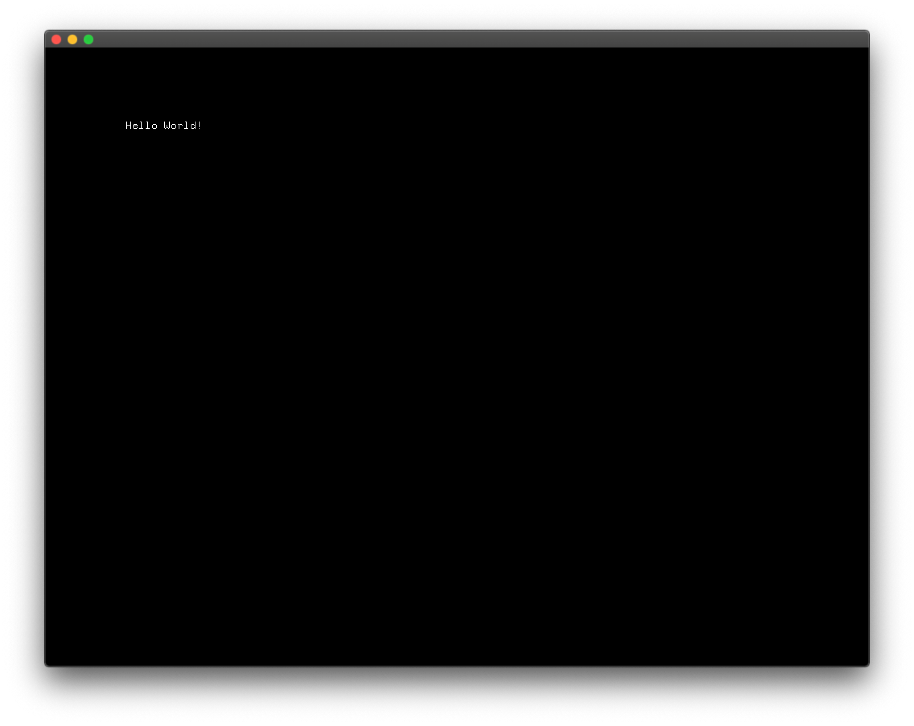
ofApp.h



화면에 Hello World!를 출력하기 위해 ofApp.cpp 파일의 draw 함수를 변경한다.(기본적으로는 비어있다.)

|  |
| --- |
| void ofApp::draw() {  ofClear(0);  ofDrawBitmapString(“Hello World!”, 100, 100);  } |

ofApp.cpp의 draw 함수를 변경 후 실행하면 다음과 같이 검은 화면에 하얀색 글씨로 Hello World!가 출력된다.



ofDrawBitmapString 함수의 두 번째, 세 번째 인자를 바꾸면 출력 위치를 변경할 수 있다. 위의 예에서는 왼쪽 위인 원점 (0,0)을 기준으로 (100, 100)에서 출력된 것이다.

3. 실습 방법

강의 자료 16p – 39p의 내용을 간단하게 정리하시오.

강의 자료 16p – 39p에서 사용된 함수들을 정리해보면 다음과 같다.

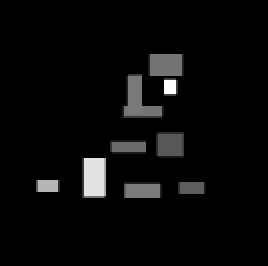
|  |
| --- |
| void ofApp::Draw() 함수에서 호출할 함수  색상 설정 함수  ofBackground(0) : 출력 화면의 색상을 설정한다. Grayscale(흑백)으로 0은 검은색, 255는 흰색이다. 강의자료에서는 0으로 설정하여 출력 화면이 검은 색으로 출력된다.  ofSetColor(255) : 출력 화면 위에 그릴 도형의 색을 설정한다. 강의 자료에서는 255로 흰색으로 도형이 그려진다.  도형 그리기 함수  ofDrawRectangle(50, 50, 100, 100) : 왼쪽 두 개 인자는 직사각형의 왼쪽 위 꼭짓점의 좌표를, 오른쪽 두 개의 인자는 직사각형의 가로길이, 세로길이를 의미한다. 왼쪽 꼭짓점이 (50, 50)에 위치하고 가로 길이와 세로 길이가 모두 100인 정사각형이 하얀색으로 그려진다.  ofDrawCircle(250, 100, 50) : 왼쪽 두 개의 인자는 원의 중심을, 오른쪽 하나의 인자는 원의 반지름을 의미한다. 중심이 (250, 100)에 위치하고 반지름이 50인 원이 하얀색으로 그려진다.  ofDrawEllipse(400, 100, 80, 100) : 왼쪽 두 개의 인자는 타원의 중심을, 오른쪽 두 개의 인자는 각각 타원의 가로축 길이, 세로축 길이이다. 중심이 (400, 100)이고 가로축이 80, 세로축이 100인 타원이 하얀색으로 그려진다.  ofDrawTriangle(500, 150, 550, 50, 600, 150) : 왼쪽으로부터 두 개씩 첫 번째 꼭짓점의 좌표, 두 번째 꼭짓점의 좌표, 세 번쨰 꼭짓점의 좌표를 나타낸다. 세 점 (500, 150), (550, 50), (600, 150)을 잇는 하얀색 삼각형이 그려진다.  ofDrawLine(700, 50, 700, 150) : 왼쪽 두개의 인자는 선분의 시작점의 좌표, 나머지 두 개의 인자는 선분의 끝점의 좌표이다. 두 점 (700, 50)과 (700, 150)을 잇는 하얀색 선분이 그려진다.  ofFill() vs ofNoFill()  기본적으로 위의 도형 그리기 함수를 호출하면 색이 채워진 도형이 그려진다. ofFill 함수가 생략되어 있으면 자동으로 색이 채워진 채 그려진다. 만약 ofFill 함수를 생략하고 ofNoFill 함수를 호출 후 도형 그리기 함수를 호출하면 해당 도형은 속이 채워지지 않은 채 외곽선만 그려진다.  마우스 관련 함수  ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT) : ofGetMousePressed는 마우스 버튼이 눌렸는지 확인하는 함수이다. 인자로 주어진 버튼이 눌렸을 경우 true를, 그렇지 않을 경우 false를 반환한다. OF\_MOUSE\_BUTTON\_LEFT는 마우스의 왼쪽 버튼을 의미한다.  ofGetMouseX(), ofGetMouseY() : 현재 마우스 커서가 위치한 지점의 X좌표와 Y좌표를 각각 반환한다.  ofSetRectMode(OF\_RECTMODE\_CENTER) : ofDrawRectangle 함수로 직사각형을 그릴 때, 앞의 두 인자를 왼쪽 위 꼭짓점의 좌표가 아니라 직사각형의 중심의 좌표로 한다.  ofRandom(50, 255) : 앞의 인자부터 뒤의 인자 사이의 값을 반환한다.(양 끝값 포함) 50부터 255 사이의 값을 반환한다.  void ofApp::setup() 함수에서 호출하는 함수  ofSetBackgroundAuto(false) : 해당 함수를 호출하지 않으면 기본적으로 draw 함수가 호출될 때마다 화면이 지워진다. 해당 함수를 false 인자를 줘서 호출하면 draw 함수가 호출되어도 기존에 그렸던 도형이 지워지지 않는다.  ofSetFrameRate(60) 초당 60 프레임을 처리하도록 설정한다. |

위의 함수들을 이용하여 여러 개의 폭발하는 모양을 만들 수 있다.

draw 함수에 기존의 코드를 지우고 다음 코드를 추가한다.

|  |
| --- |
| if(ofGetMousePressed(OF\_MOUSE\_BUTTON\_LEFT)) {  ofSetRectMode(OF\_RECTMODE\_CENTER):  int numRects = 10;  for(int r = 0; r < numRects; r++) {  ofSetColor(ofRandom(50, 255));  float width = ofRandom(5, 20);  float height = ofRandom(5, 20);  float xOffset = ofRandom(-40, 40);  float yOffset = ofRandom(-40, 40);  ofDrawRectangle(ofGetMouseX() + xOffset, ofGetMouseY() + yOffset, width, height);  }  } |

위의 코드를 추가 후 실행해보면 마우스 클릭 한 번당 10개의 직사각형이 생겨난다. 10개의 직사각형은 각각 색상, 가로 길이, 세로 길이, 중심의 위치가 랜덤하게 정해진다. 중심의 위치는 클릭 시 마우스 커서 위치를 기준으로 x축 방향으로 xOffset, y축 방향으로 yOffset 떨어진 곳에 위치한다. setup 함수에서 ofSetFrameRate(1)을 호출하여 1초당 1프레임씩 처리하도록 하면 쉽게 확인 가능하다.



한 번 클릭하니 10개의 랜덤한 크기, 랜덤한 위치, 랜덤한 색상의 직사각형이 그려진 것을 알 수 있다. 원활한 폭발하는 이미지를 그리기 위해 초당 프레임을 60으로 설정한다.

ofSetFrameRate(60)

다음과 같이 폭발하는 이미지를 얻을 수 있다.



ofColor 객체를 생성하면 색상을 정의할 수 있다. ofColor myOrange(255, 132, 0) 처럼 RGB 값만 주어 불투명한 색을 만들 수도 있고 ofColor myBlue(0, 0, 255, 50) 처럼 RGBA(alpha) 값을 주어 투명한 색을 만들 수도 있다.

ofVec2f 객체를 생성해 벡터를 2차원 벡터를 만들 수 있다.

update 함수는 non-drawing 작업들을 위한 것으로 한 frame 당 한 번 불려지도록 설계된 함수이다.

4. 기타

openFrameworks처럼 여러 라이브러리들을 통합하여 제공하는 것을 프레임워크(Framework)라고 한다.

5. 참고 문헌

[1] openFrameworks, “about”, https://openframeworks.cc/about

[2] 위키백과 “OpenGL”, https://ko.wikipedia.org/wiki/OpenGL

[3] 위키백과 “OpenGL Extension Wrangler Library”,

https://en.wikipedia.org/wiki/OpenGL\_Extension\_Wrangler\_Library

[4] 위키백과 “OpenGL 유틸리티 툴킷”,

https://ko.wikipedia.org/wiki/OpenGL\_%EC%9C%A0%ED%8B%B8%EB%A6%AC%ED%8B%B0\_%ED%88%B4%ED%82%B7

[5] libtess, https://www.npmjs.com/package/libtess

[6] cairographics.org, https://www.cairographics.org/

[7] The RtAudio Home Page, http://www.music.mcgill.ca/~gary/rtaudio/

[8] Port Audio, http://www.portaudio.com/

[9] 위키백과 “OpenAL”, https://ko.wikipedia.org/wiki/OpenAL

[10] openal.org, https://www.openal.org/

[11] 위키백과 “고속 푸리에 변환”,

https://ko.wikipedia.org/wiki/%EA%B3%A0%EC%86%8D\_%ED%91%B8%EB%A6%AC%EC%97%90\_%EB%B3%80%ED%99%98

[12] Kiss FFT, https://directory.fsf.org/wiki/Kiss\_FFT

[13] 위키백과 “FMOD”, https://ko.wikipedia.org/wiki/FMOD

[14] 위키백과 “래스터화”,

https://ko.wikipedia.org/wiki/%EB%9E%98%EC%8A%A4%ED%84%B0%ED%99%94

[15] 위키백과 “프리타입”,

https://ko.wikipedia.org/wiki/%ED%94%84%EB%A6%AC%ED%83%80%EC%9E%85

[16] FreeImage, https://freeimage.sourceforge.io/index.html

[17] 위키백과 “퀵타임”, https://ko.wikipedia.org/wiki/%ED%80%B5%ED%83%80%EC%9E%84

[18] 위키백과 “GStreamer”, https://ko.wikipedia.org/wiki/GStreamer

[19] videoInput, http://www.muonics.net/school/spring05/videoInput/

[20] POCO C++ Libraries, https://pocoproject.org/index.html

[21] 위키백과 “OpenCV”, https://ko.wikipedia.org/wiki/OpenCV

[22] 위키백과 “Open Asset Import Library”,

https://en.wikipedia.org/wiki/Open\_Asset\_Import\_Library

[23] 위키백과 “오픈프레임웍스”

https://ko.wikipedia.org/wiki/%EC%98%A4%ED%94%88%ED%94%84%EB%A0%88%EC%9E%84%EC%9B%8D%EC%8A%A4#%ED%8A%B9%EC%A7%95

# [24] ofTutorials - Introduction

http://openframeworks.kr/tutorials/introduction/000\_introduction.html