

# Computer Graphics

## *Input Processing*

1. Read the OS messages received via SDL events and process key/mouse inputs. Create a class or struct that allows you to query the input during any given frame (similar to Unity).
  - a) Filter the events via their type to get the ones we need using **event.type**; relevant are:

```
SDL_EventType::SDL_EVENT_KEY_UP,  
SDL_EventType::SDL_EVENT_KEY_DOWN,  
SDL_EventType::SDL_EVENT_MOUSE_BUTTON_UP,  
SDL_EventType::SDL_EVENT_MOUSE_BUTTON_DOWN,  
SDL_EventType::SDL_EVENT_MOUSE_MOTION
```
  - b) Read the event data via **event.key**, **event.button** and **event.motion**
  - c) Store the input data in a way that it can be retrieved later (and clear it after every frame to prevent reading inputs from last frame!). This may best be achieved via **std::set** (allows storing a set of items) or **std::vector** (C++-style array that can grow). Feel free to use other solutions as well if they work better!
  - d) Provide a method for querying inputs, e.g. with an “input” object:

```
input.key_up(SDLK_Y);  
input.key_pressed(SDLK_A);  
input.key_down(SDLK_B);
```

Mouse button events should be handled similarly, while mouse motion will simply provide x and y positions or deltas.