Fun with project

Outline

Outline

Allegro install

- Windows:
  - https://drive.google.com/file/d/1OQBL_ChhbuOfK_qGZtIvIzfh11LlxWz4/view?usp=sharing
- Mac:
  - https://goo.gl/vpTA9t
  - https://hackmd.io/@kerwintsai/SkRTk6kCS
- Allegro download:
  - https://github.com/liballeg/allegro5/releases
- Allegra reference:
  - https://liballeg.org/a5docs/trunk/

Outline

Main.cpp

```
GameWindow *TowerGame= new GameWindow();

TowerGame->game_play();
```

Initial the game

Start the game

GameWindow.cpp

~GameWindow()

```
al_init_primitives_addon();
al_init_font_addon(); // initialize the font addon
al_init_ttf_addon(); // initialize the ttf (True Type Font) addon
al_init_image_addon(); // initialize the image addon
al_init_acodec_addon(); // initialize acodec addon

al_install_keyboard(); // install keyboard event
al_install_mouse();    // install mouse event
al_install_audio();    // install audio event
```

Initialize the basic function of allegro, install event and register event.

```
al_register_event_source(event_queue, al_get_display_event_source(display));
al_register_event_source(event_queue, al_get_keyboard_event_source());
al_register_event_source(event_queue, al_get_mouse_event_source());

al_register_event_source(event_queue, al_get_timer_event_source(timer));
al_register_event_source(event_queue, al_get_timer_event_source(monster_pro));
```

GameWindow.cpp

~initial the game

Four steps to use the function in allegro

GameWindow.cpp

~start the game

```
GameWindow::game_play()
{
    int msg;

    srand(time(NULL));

    msg = -1;
    game_reset();
    game_begin();

    while(msg != GAME_EXIT)
    {
        msg = game_run();
    }

    show_err_msg(msg);
}
```

Clear components you display or used.
Stop the sound and timer

GameWindow.cpp

~start the game

```
GameWindow::game_play()
{
    int msg;

    srand(time(NULL));

    msg = -1;
    game_reset();
    game_begin();

    while(msg != GAME_EXIT)
    {
        msg = game_run();
    }

    show_err_msg(msg);
}
```

Start timer, disply sound and draw picture on screen

GameWindow.cpp

~start the game

```
GameWindow::game_play()
{
    int msg;

    srand(time(NULL));

    msg = -1;
    game_reset();
    game_begin();

    while(msg != GAME_EXIT)
    {
        msg = game_run();
    }

    show_err_msg(msg);
}
```

Use while loop to keep running the game.
Process the event, update game component, and draw picture

GameWindow.cpp

~game_reset()

```
GameWindow::game_reset() {
    // reset game and begin
    for(auto&& child : towerSet) {
        delete child;
    }
    towerSet.clear();
    monsterSet.clear();
    selectedTower = -1;
    lastClicked = -1;
    Coin_Inc_Count = 0;
    Monster_Pro_Count = 0;
    mute = false;
    redraw = false;
    menu->Reset();
    // stop sample instance
    al_stop_sample_instance(backgroundSound);
    al_stop_sample_instance(startSound);
    // stop timer
    al_stop_timer(timer);
    al_stop_timer(monster_pro);
}
```

You can clear all the component in your game in this function.
If you want your game can restart then this function will be important

GameWindow.cpp

~game_begin()

```
GameWindow::game_begin() {
    printf(">>> Start Level[%d]\n", level->getLevel());
    draw_running_map();

    al_play_sample_instance(startSound);
    while(al_get_sample_instance_playing(startSound));
    al_play_sample_instance(backgroundSound);

    al_start_timer(timer);
    al_start_timer(monster_pro);
}
```

Start your game here, you can draw the basic item first like start menu.
Start timer then you can run your game.

GameWindow.cpp

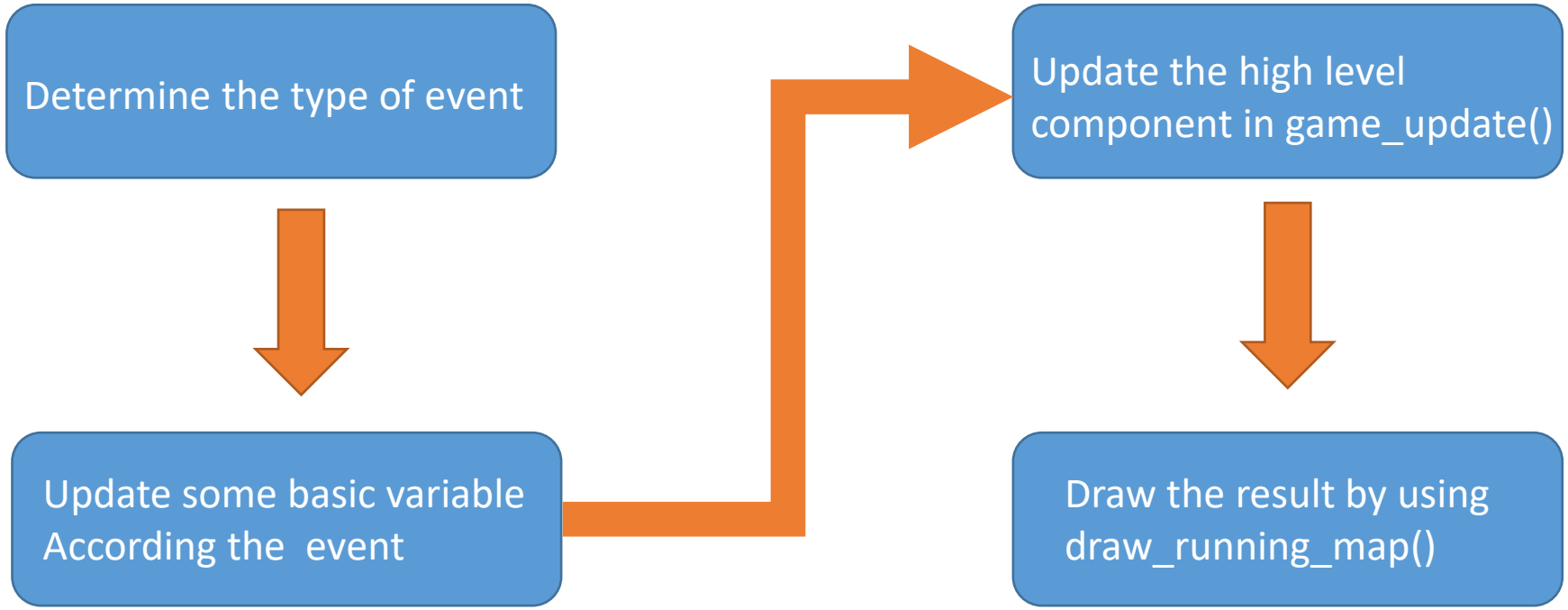
~game_run()

```
GameWindow::game_run() {
    int error = GAME_CONTINUE;

    if (!al_is_event_queue_empty(event_queue)) {

        error = process_event();
    }
    return error;
}
```

Process the event in event queue

GameWindow.cpp

~process_event ()



Outline

Todo 1

- In GameWindow.cpp
- In GameWindow::game_update() function

```
/*TODO:*/
/*Allow towers to detect if monster enters its range*/
/*Hint: Tower::DetectAttack*/
```

Todo 1

- In GameWindow.cpp
- In GameWindow::game_update() function

```
for( /* go through towerSet */ ) {
    for(/* go through monster Set */) {
        /* DetectAttack */
    }
}
```

Todo 2

- In GameWindow.cpp
- In GameWindow::game_update() function

```
/*TODO:*/
/*1. For each tower, traverse its attack set*/
/*2. If the monster collide with any attack, reduce the HP of the monster*/
/*3. Remember to set isDestroyed to "true" if monster is killed*/
/*Hint: Tower::TriggerAttack*/
```

Todo 2

- In GameWindow.cpp
- In GameWindow::game_update() function

```
for(/*go through towerSet*/){
    isDestroyed = /*TriggerAttack*/
}
```

Todo 3

- In GameWindow.cpp
- In GameWindow::game_update() function

```
/*TODO:*/
/*1. Update the attack set of each tower*/
/*Hint: Tower::UpdateAttack*/
```

Todo 3

- In GameWindow.cpp
- In GameWindow::game_update() function

```
for( /* go through towerSet */ ) {
    /* UpdateAttack */
}
```

Todo 4

- In GameWindow.cpp
- In GameWindow::process_event() function

```
case ALLEGRO_KEY_P:
    /*TODO: handle pause event here*/
    break;
```

Todo 4

- In GameWindow.cpp
- In GameWindow::process_event() function

```
al_get_timer_started(timer)
al_stop_timer(timer)
al_start_timer(timer)
```

Check if “timer” is started.
If yes return true otherwise return false

Stop timer

Start timer

Todo 5

- In Tower.cpp
- In Tower::TriggerAttack(Monster *monster) function

```
/*TODO:*/
/*1. Reduce the monster HP by the harm point*/
/*2. Erase and delete the attack from attack set*/
/*3. Return true if the monster's HP is reduced to zero*/
```

Todo 5

- In Tower.cpp
- In Tower::TriggerAttack(Monster *monster) function

Use something like this…..

```
monster->Subtract_HP( attack_set[i]->getHarmPoint() )
```

Outline

01 Introduction

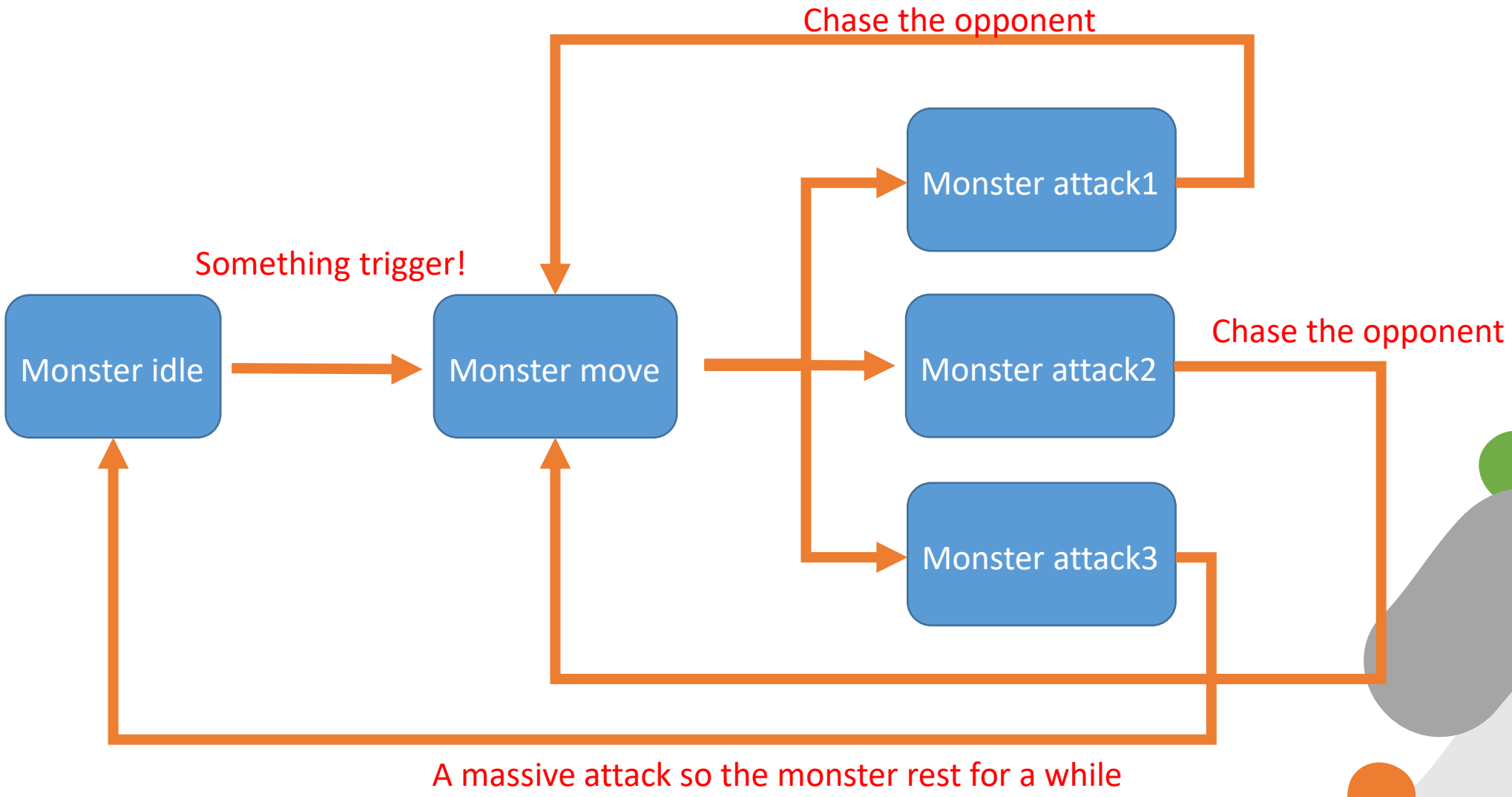
02 Trace code

03 Todo

04 Supplementary

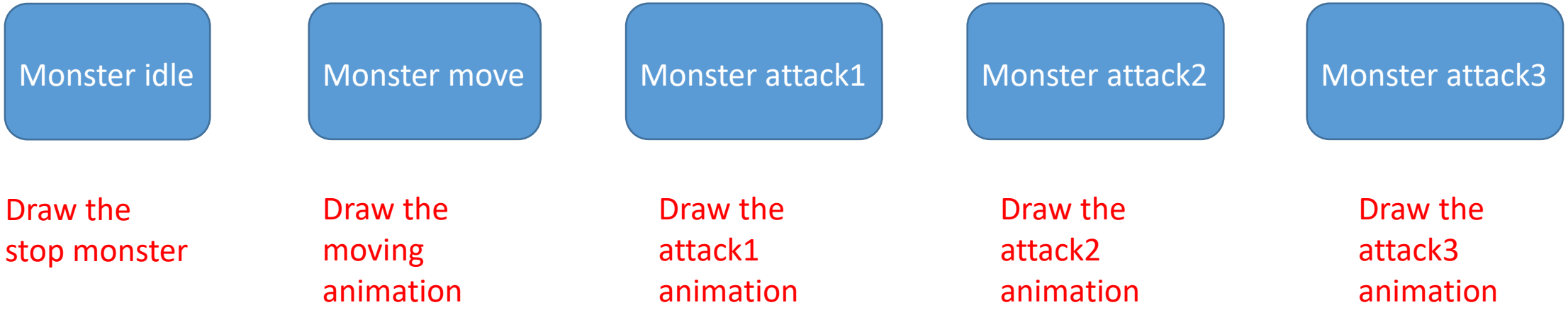
AI

~Use finite state machine!!!



AI

~Use finite state machine!!!



By using states, we can easy draw the animation of monster

Special effect sound

~Use finite state machine!!!

Monster idle	Monster move	Monster attack1	Monster attack2	Monster attack3
Display the sound of panting	Display the sound of moving	Display the sound of attack1	Display the sound of attack2	Display the sound of attack3

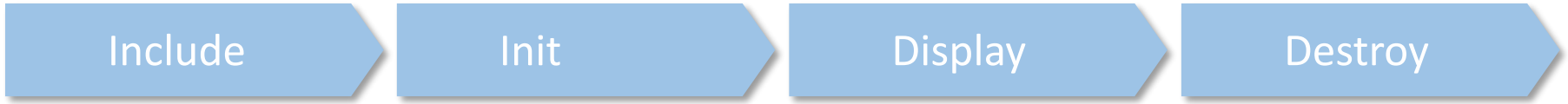
Just judge the state then use
al_play_sample(*ALLEGRO_SAMPLE)

Display video

- Use something like mp4 to jpg to transfer the video into images.
- Load the images into an array. You can use sprintf to manipulate the path of image
- Set a timer as the fps of your video.
- Set a event queue to get the timer event.
- Each time the timer trigger display the image on the screen
- Plus the index of array by 1 to display the next image.
- Then you get the effect of display video!

Display video

- Use allegro video addon!



Display video

- Use allegro video addon!

Include → Init → Display → Destroy

```
#include <allegro5/allegro_audio.h>
#include <allegro5/allegro_video.h>
```

Display video

- Use allegro video addon!

Include | Init | Display | Destroy

```
al_init_video_addon();
ALLEGRO_VIDEO *video = al_open_video(filename);
ALLEGRO_EVENT_SOURCE *temp = al_get_video_event_source(video);
al_register_event_source(queue, temp);
```

Initialize video addon

Display video

- Use allegro video addon!

Include | Init | Display | Destroy

```
al_init_video_addon();
ALLEGRO_VIDEO *video = al_open_video(filename);
ALLEGRO_EVENT_SOURCE *temp = al_get_video_event_source(video);
al_register_event_source(queue, temp);
```

Open the video file
(Note: only ".ogv" type of video is avaliable)

Display video

- Use allegro video addon!

Include | Init | Display | Destroy

```
al_init_video_addon();
ALLEGRO_VIDEO *video = al_open_video(filename);
ALLEGRO_EVENT_SOURCE *temp = al_get_video_event_source(video);
al_register_event_source(queue, temp);
```

Get the event source of video and register it into event queue.

Display video

- Use allegro video addon!

Include → Init → Display → Destroy

```
al_reserve_samples(1);
al_start_video(video, al_get_default_mixer());
```

Reserve the sound channel for video and attach it into default mixer, then the video started

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
while(1){
    al_wait_for_event(queue, &event);
    if(event.type == ALLEGRO_EVENT_TIMER) {
        video_display(video);
    } else if( event.type == ALLEGRO_EVENT_DISPLAY_CLOSE ) {
        al_close_video(video);
        break;
    } else if( event.type == ALLEGRO_EVENT_VIDEO_FINISHED ) {
        break;
    }
}
```

The FPS of display the frame of video.

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
while(1){
    al_wait_for_event(queue, &event);
    if(event.type == ALLEGRO_EVENT_TIMER) {
        video_display(video);
    } else if( event.type == ALLEGRO_EVENT_DISPLAY_CLOSE ) {
        al_close_video(video);
        break;
    } else if( event.type == ALLEGRO_EVENT_VIDEO_FINISHED ) {
        break;
    }
}
```

If the display be closed, then close the video and break the loop

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
while(1){
    al_wait_for_event(queue, &event);
    if(event.type == ALLEGRO_EVENT_TIMER) {
        video_display(video);
    } else if( event.type == ALLEGRO_EVENT_DISPLAY_CLOSE ) {
        al_close_video(video);
        break;
    } else if( event.type == ALLEGRO_EVENT_VIDEO_FINISHED ) {
        break;
    }
}
```

If the video finished then break the loop

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
void video_display(ALLEGRO_VIDEO *video) {
    ALLEGRO_BITMAP *frame = al_get_video_frame(video);
    if (!frame)
        return;
    al_draw_scaled_bitmap(frame,
                          0, 0,
                          al_get_bitmap_width(frame),
                          al_get_bitmap_height(frame),
                          0, 0,
                          al_get_display_width(screen),
                          al_get_display_height(screen), 0);
    al_flip_display();
}
```

Get one frame from the video, frame may be none therefore you should determine the value before draw

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
void video_display(ALLEGRO_VIDEO *video) {
    ALLEGRO_BITMAP *frame = al_get_video_frame(video);
    if (!frame)
        return;
    al_draw_scaled_bitmap(frame,
                          0, 0,
                          al_get_bitmap_width(frame),
                          al_get_bitmap_height(frame),
                          0, 0,
                          al_get_display_width(screen),
                          al_get_display_height(screen), 0);
    al_flip_display();
}
```

The position of original bitmap you want to scale

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
void video_display(ALLEGRO_VIDEO *video) {
    ALLEGRO_BITMAP *frame = al_get_video_frame(video);
    if (!frame)
        return;
    al_draw_scaled_bitmap(frame,
                          0, 0,
                          al_get_bitmap_width(frame),
                          al_get_bitmap_height(frame),
                          0, 0,
                          al_get_display_width(screen),
                          al_get_display_height(screen), 0);
    al_flip_display();
}
```

The width/height of original bitmap you want to scale

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
void video_display(ALLEGRO_VIDEO *video) {
    ALLEGRO_BITMAP *frame = al_get_video_frame(video);
    if (!frame)
        return;
    al_draw_scaled_bitmap(frame,
                          0, 0,
                          al_get_bitmap_width(frame),
                          al_get_bitmap_height(frame),
                          0, 0,
                          al_get_display_width(screen),
                          al_get_display_height(screen), 0);
    al_flip_display();
}
```

The position of the result image on the display

Display video

- Use a loop to display the image of video

Include | Init | Display | Destroy

```
void video_display(ALLEGRO_VIDEO *video) {
    ALLEGRO_BITMAP *frame = al_get_video_frame(video);
    if (!frame)
        return;
    al_draw_scaled_bitmap(frame,
                          0, 0,
                          al_get_bitmap_width(frame),
                          al_get_bitmap_height(frame),
                          0, 0,
                          al_get_display_width(screen),
                          al_get_display_height(screen), 0);
    al_flip_display();
}
```

The width/height of the result image on the display

Display video

- Use a loop to display the image of video

Include → Init → **Display** → Destroy

```
void video_display(ALLEGRO_VIDEO *video) {
    ALLEGRO_BITMAP *frame = al_get_video_frame(video);
    if (!frame)
        return;
    al_draw_scaled_bitmap(frame,
                          0, 0,
                          al_get_bitmap_width(frame),
                          al_get_bitmap_height(frame),
                          0, 0,
                          al_get_display_width(screen),
                          al_get_display_height(screen), 0);
    al_flip_display();
}
```

The same flag of draw bitmap, control the flip

Display video

Include | Init | Display | Destroy

```
al_draw_scaled_bitmap(frame,
                      0, 0,
                      al_get_bitmap_width(frame),
                      al_get_bitmap_height(frame),
                      0, 0,
                      al_get_display_width(screen),
                      al_get_display_height(screen), 0);
```


```
al_draw_scaled_bitmap(frame,
                      100, 0,
                      al_get_bitmap_width(frame)-100,
                      al_get_bitmap_height(frame),
                      0, 0,
                      al_get_display_width(screen),
                      al_get_display_height(screen), 0);
```

Display video

Include → Init → Display → Destroy

- Close the video will automatically destroy the video.

Display GIF

- This addon is not official
- You need to include the file in your project
- https://allegro5.org/algif5/

algif5

About

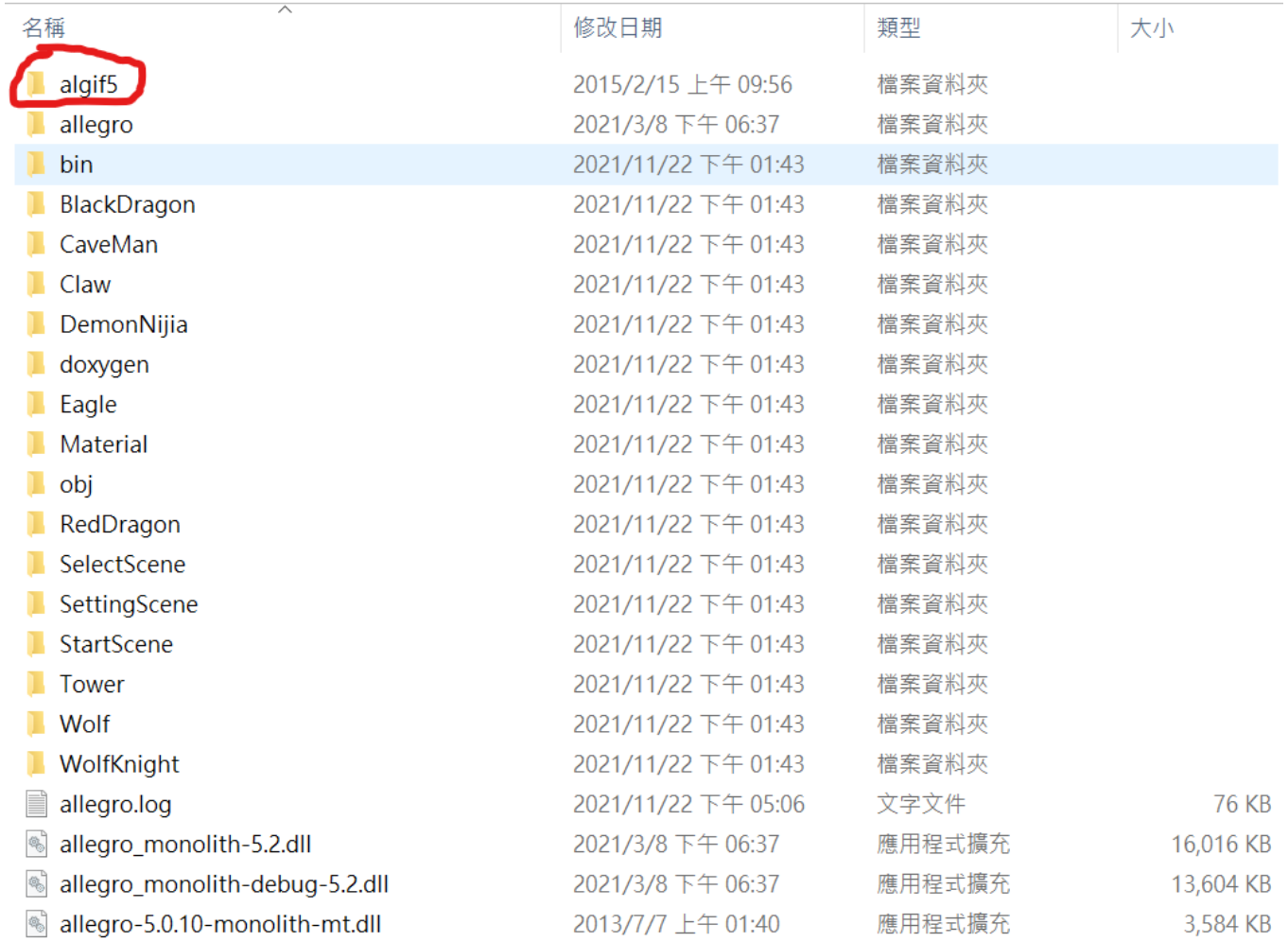
algif5 is a gif loading library for Allegro 5. Homepage, Github.

Download

algif5.zip

Display GIF

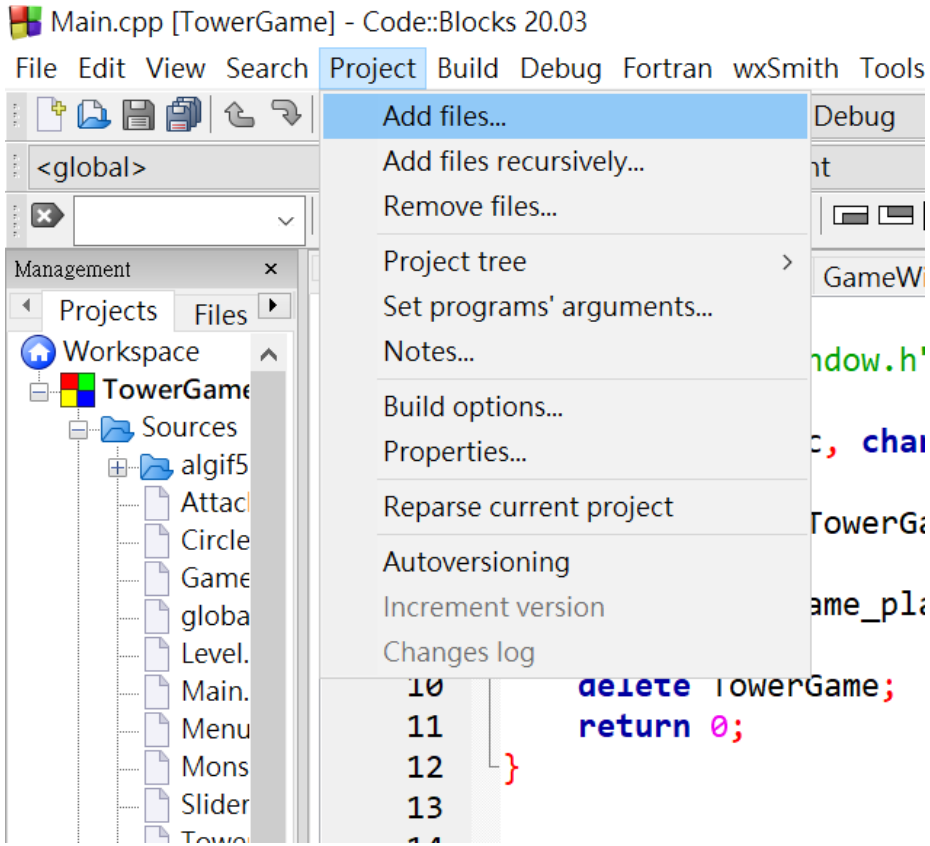
- Download the zip, unzip it and put it into your final project folder

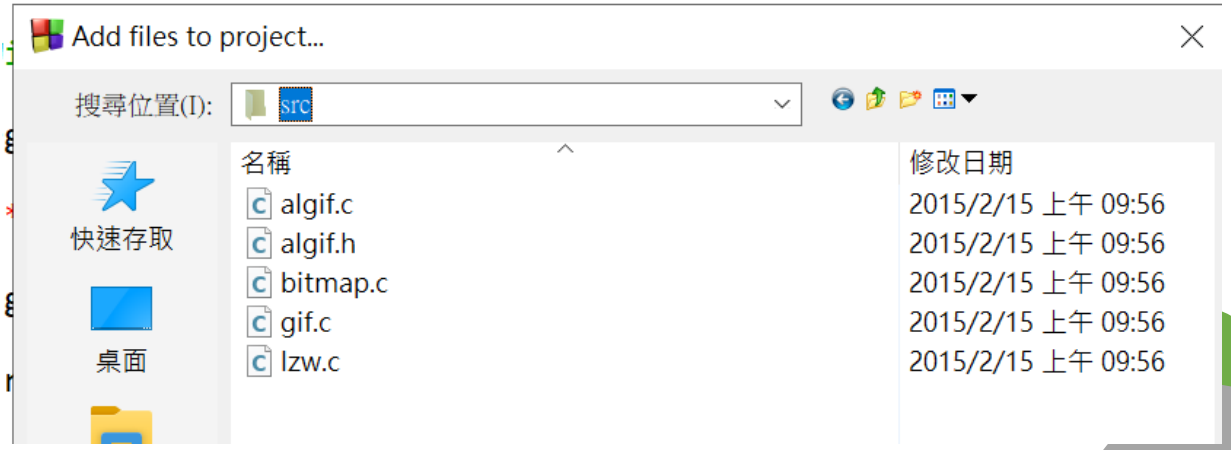


名稱	修改日期	類型	大小
algif5	2015/2/15 上午 09:56	檔案資料夾	
allegro	2021/3/8 下午 06:37	檔案資料夾	
bin	2021/11/22 下午 01:43	檔案資料夾	
BlackDragon	2021/11/22 下午 01:43	檔案資料夾	
CaveMan	2021/11/22 下午 01:43	檔案資料夾	
Claw	2021/11/22 下午 01:43	檔案資料夾	
DemonNijia	2021/11/22 下午 01:43	檔案資料夾	
doxygen	2021/11/22 下午 01:43	檔案資料夾	
Eagle	2021/11/22 下午 01:43	檔案資料夾	
Material	2021/11/22 下午 01:43	檔案資料夾	
obj	2021/11/22 下午 01:43	檔案資料夾	
RedDragon	2021/11/22 下午 01:43	檔案資料夾	
SelectScene	2021/11/22 下午 01:43	檔案資料夾	
SettingScene	2021/11/22 下午 01:43	檔案資料夾	
StartScene	2021/11/22 下午 01:43	檔案資料夾	
Tower	2021/11/22 下午 01:43	檔案資料夾	
Wolf	2021/11/22 下午 01:43	檔案資料夾	
WolfKnight	2021/11/22 下午 01:43	檔案資料夾	
allegro.log	2021/11/22 下午 05:06	文字文件	76 KB
allegro_monolith-5.2.dll	2021/3/8 下午 06:37	應用程式擴充	16,016 KB
allegro_monolith-debug-5.2.dll	2021/3/8 下午 06:37	應用程式擴充	13,604 KB
allegro-5.0.10-monolith-mt.dll	2013/7/7 上午 01:40	應用程式擴充	3,584 KB

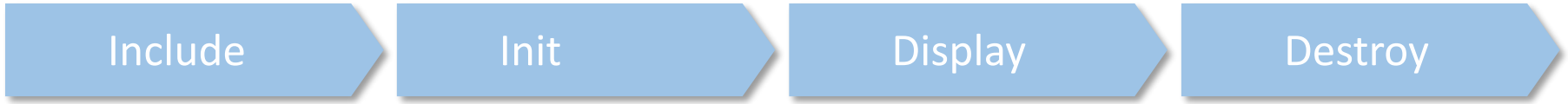
Display GIF

- Use add file to include algif5
- Choose all files under algif5/src/





Display GIF



- Use the library of algif5!

Display GIF

Include → Init → Display → Destroy

- Include should have the path to algif.h

```
#include "algif5/src/algif.h"
```

Display GIF

Include | Init | Display | Destroy

- Simply load the gif by algif_load_animation().
- Make sure the path of your gif is correct.

```
ALGIF_ANIMATION *gif = algif_load_animation("meme.gif");
```

Display GIF

Include | Init | Display | Destroy

- The original algif lack of the start time variable
- We can declare a gif struct

```
struct GIF{
    double start_time = 0;
    ALGIF_ANIMATION *gif = NULL;
    double get_time(double time) {
        if(start_time == 0) start_time = time;
        return time - start_time;
    }
} obj;
```

Display GIF

Include | Init | Display | Destroy

- Use GIF.get_time() to get the actual start time of gif
- al_get_time(): the time(seconds) from the initialization of allegro
- algif_get_bitmap() will return the frame that related to the time

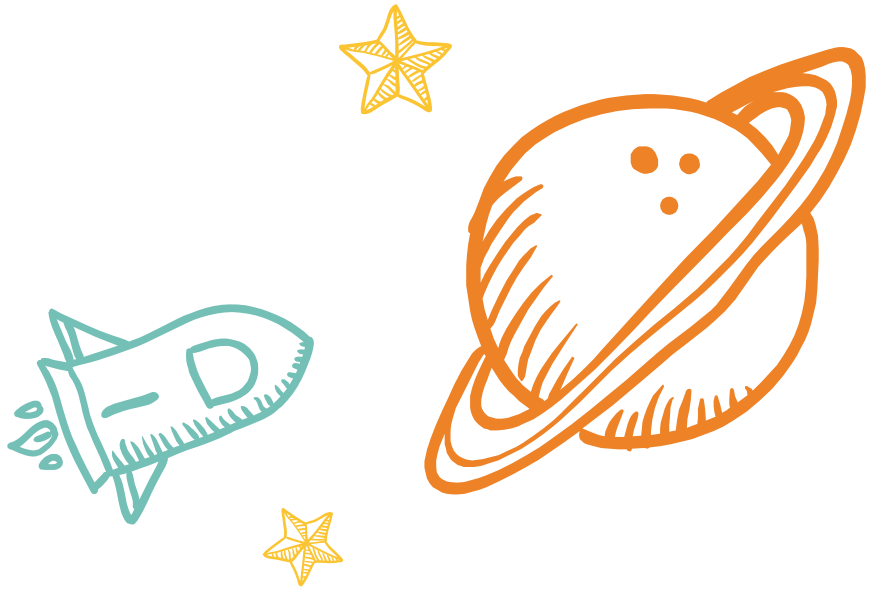
```
ALLEGRO_BITMAP *frame = algif_get_bitmap( obj.gif, obj.get_time( al_get_time() ) );
```

Display GIF

Include → Init → Display → **Destroy**

- Use algif_destroy_animation() to destroy the gif

```
algif_destroy_animation(obj.gif);
```



The End~