รายชื่อสมาชิก

กลุ่ม Dawning

นายโชคอนันต์ ภักศร 6630250079

นายรัชตพล พ่วงงามพันธ์ 6630250419

นายวุฒิภัทร ศรีสอาด 6630250460

นายสุวัฒน์ สันติมาลัย 6630250494

```cpp
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <fstream>
#include <random>
#include <cstdlib>

using namespace std;

void clearScreen(){
    cout << "\033[2J\033[H";
}

class Word{
    private:
        vector<string> words;
        string secretWord;

    public:
        void putWord(string word){
            words.push_back(word);
        }

        vector<string> getWords(){
            return words;
        }

        string getSecretWord(){
            return secretWord;
        }

        void randomWord();

};

void Word::randomWord() {
    if (words.empty()){
        secretWord = "";
    }

    random_device rd;
    mt19937 generator(rd());

    uniform_int_distribution<int> distribution(0, words.size() - 1);
    int random_index = distribution(generator);

    secretWord = words[random_index];
}

class Name{
    public:
        string name;
};

class Player: public Name{
    public:
        int lives;
};
```

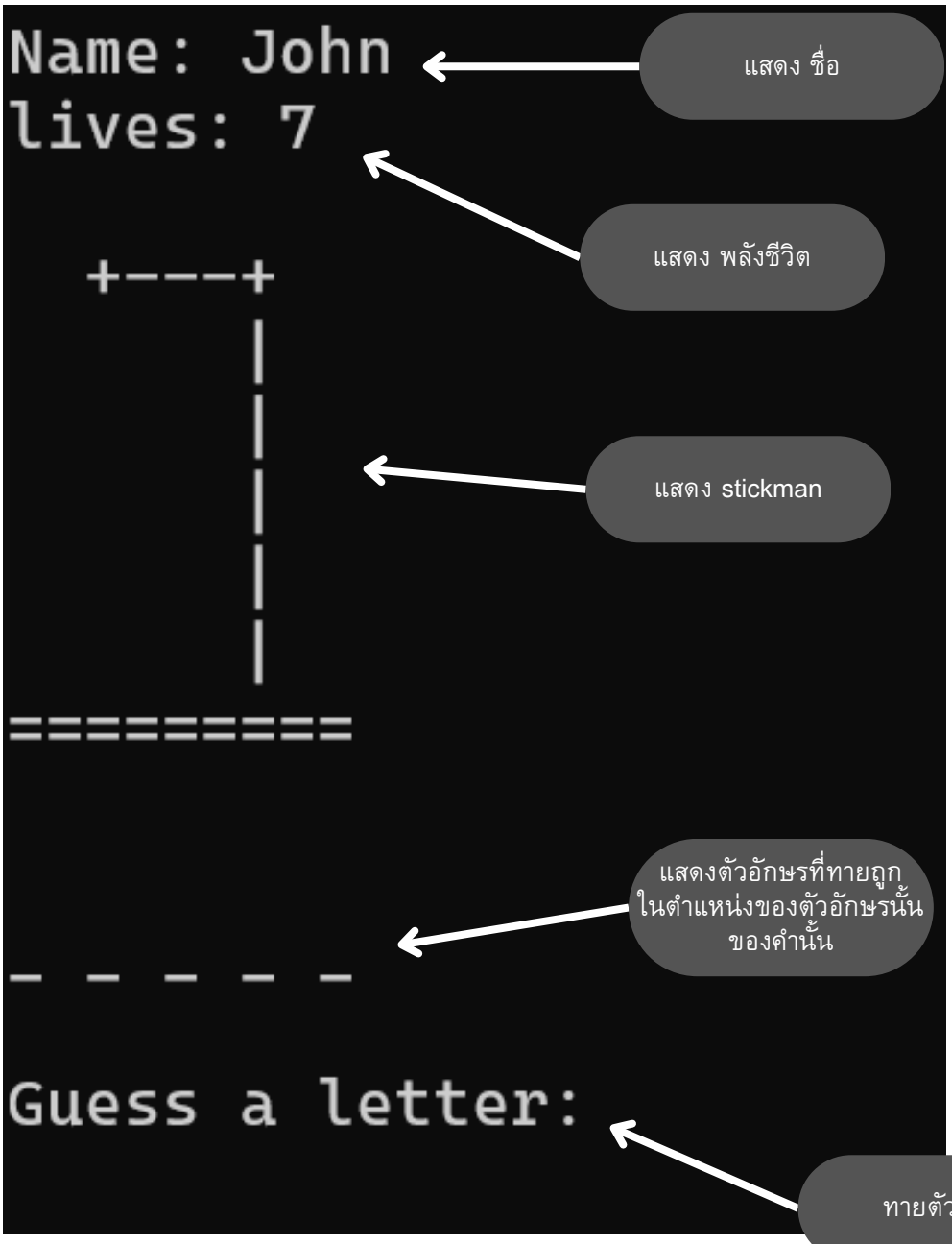
```
60
61  class HangmanGame : public Player{
62      private:
63          vector<char> guessedWord;
64          string secretWord;
65          vector<string> stickman;
66
67      public:
68          HangmanGame(string w){
69              secretWord = w;
70              lives = 7;
71              int wordLenght = secretWord.length();
72              guessedWord = vector<char>(wordLenght, '_');
73          };
74
75          HangmanGame(string w, int l){
76              secretWord = w;
77              lives = l;
78              int wordLenght = secretWord.length();
79              guessedWord = vector<char>(wordLenght, '_');
80          };
81
82          void putName(string n){
83              if (n != ""){
84                  name = n;
85              }else{
86                  name = "Anonymous";
87              }
88          }
89
90          bool isGameOver(){
91              return lives == 0 || secretWord == string(guessedWord.begin(), guessedWord.end());
92          }
93
94          void guessLetter(char letter){
95              letter = tolower(letter);
96              bool found = false;
97
98              for (int i = 0; i < secretWord.size(); ++i){
99                  if (secretWord[i] == letter){
100                     guessedWord[i] = letter;
101                     found = true;
102                 }
103             }
104
105             if (!found){
106                 --lives;
107             }
108         }
109
110         void putState(vector<string> stages){
111             stickman = stages;
112         }
113
114         void displayStickman(){
115             cout << stickman[lives] << endl;
116         }
117
118         void play();
119 };
120
```

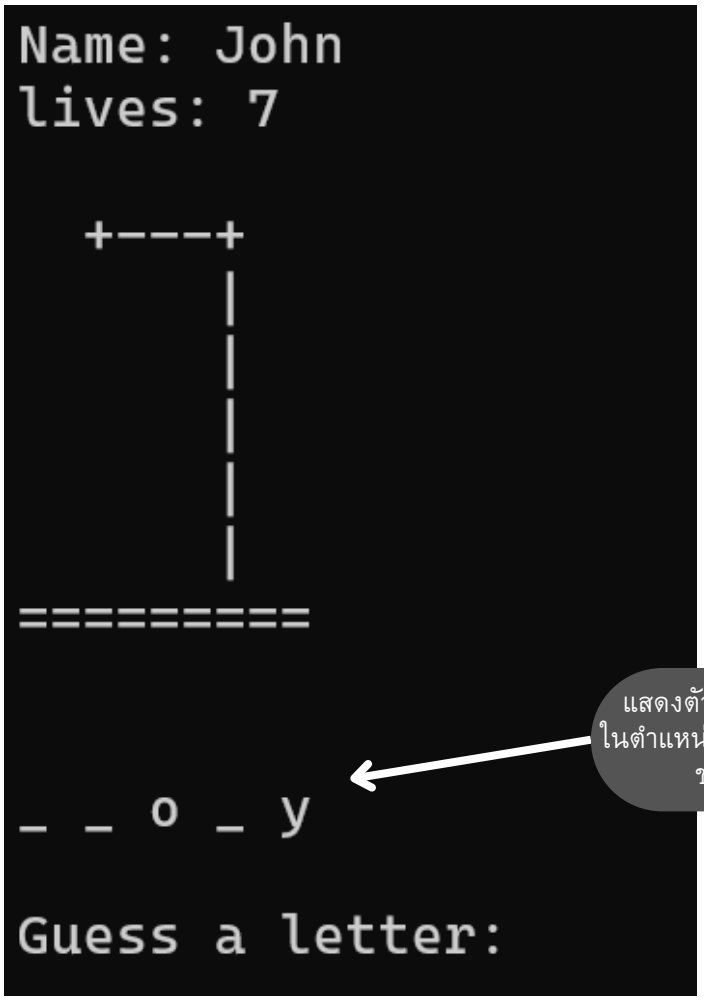
```
120
121 void HangmanGame::play(){
122     while (!isGameOver()){
123         clearScreen();
124         cout << "Name: " << name << endl;
125         cout << "lives: " << lives << endl;
126 
127         displayStickman();
128 
129         cout << endl;
130         for (char c : guessedWord){
131             cout << c << " ";
132         }
133 
134         cout << "\n\nGuess a letter: ";
135 
136         char guess;
137         cin >> guess;
138 
139         if (!isalpha(guess)){
140             cout << "Invalid input. Please enter a letter." << endl;
141             continue;
142         }
143 
144         guessLetter(guess);
145     }
146 
147     clearScreen();
148     cout << "Name: " << name << endl;
149     cout << "Your lives: " << lives << endl;
150     displayStickman();
151     cout << endl;
152     for (char c : guessedWord){
153         cout << c << " ";
154     }
155 
156     if (secretWord == string(guessedWord.begin(), guessedWord.end())){
157         cout << "\n\nCongratulations "<< name << " Win!!\n\n";
158         cout << "The correct word is '" << secretWord << "'\n" << endl;
159     }
160     else{
161         cout << "\n\n" << name << " You lose" << endl;
162         cout << "The correct word is '" << secretWord << "'\n" << endl;
163     }
164 }
165 
```

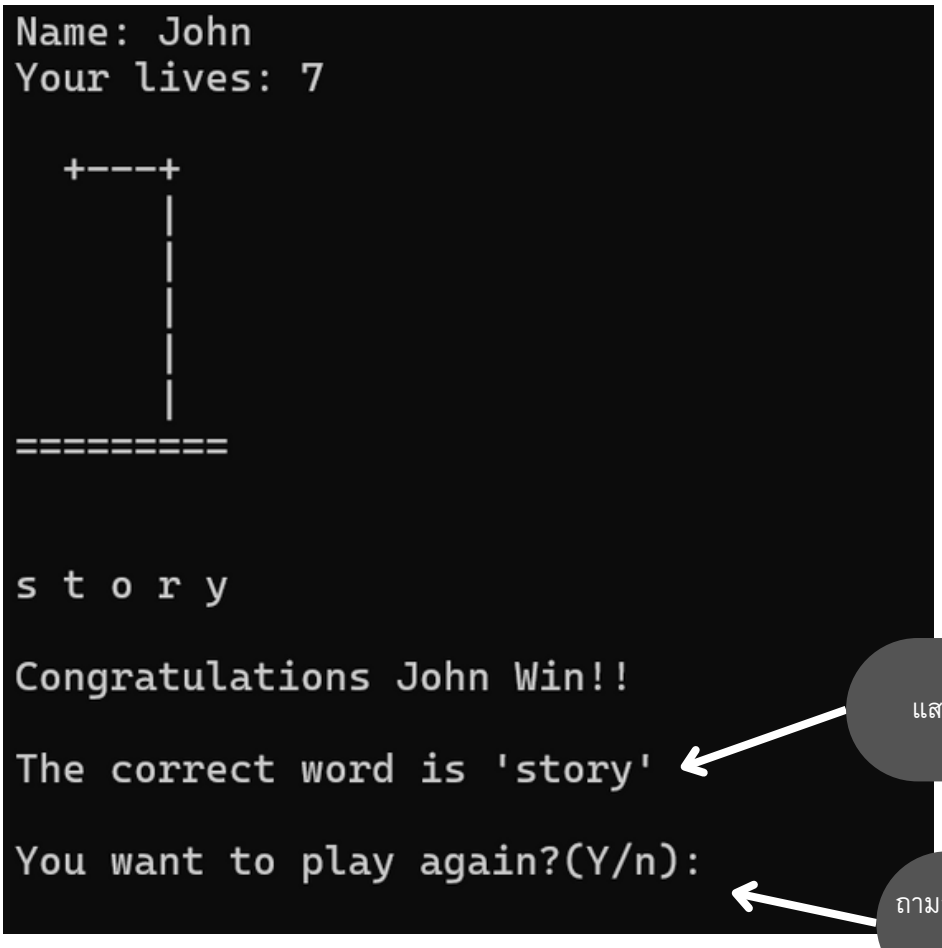
```cpp
int main(){
    clearScreen();

    string logo = R"(
     _
    | |
    | |__   __ _ _ __   __ _ _ __ ___   __ _ _ __
    | '_ \ / _` | '_ \ / _` | '_ ` _ \ / _` | '_ \
    | | | | (_| | | | | (_| | | | | | | (_| | | | |
    |_| |_|\__,_|_| |_|\__, |_| |_| |_|\__,_|_| |_|
                        __/ |
                       |____/
    )";

    Word word;

    ifstream secretWord("./secretWord.txt");
    ifstream Stickman("./Stickman.txt");

    if (!secretWord.is_open() || !Stickman.is_open()){
        cout << "Error opening file." << endl;
        return 1;
    };

    vector<string> stages;
    string stage;

    while (getline(Stickman, stage, '\'')) {
        if (!stage.empty()) {
            stages.push_back(stage);
        };
    };

    string w;
    while (getline(secretWord, w)){
        transform(w.begin(), w.end(), w.begin(), ::tolower);
        word.putWord(w);
    };

    secretWord.close();
    Stickman.close();

```

```
208     cout << logo << endl;
209
210     cout << "Hangman Game \n-> setting\n" << endl;
211
212     string config;
213     int lives;
214     cout << "Your default lives is 7 and maximum lives is 7" << endl;
215     cout << "You want to config your lives?(Y/n): ";
216     cin >> config;
217
218     if (config[0] == 'y' || config[0] == 'Y') {
219         cout << "Enter the number of lives: ";
220         cin >> lives;
221         if (lives > 7) {
222                 lives = 7;
223         }else if (lives < 1){
224             lives = 1;
225         }
226     }else{
227         lives = 7;
228     }
229
230     string name;
231     cout << "Input your name : ";
232     cin >> name;
233
234
235     while (true){
236         word.randomWord();
237         HangmanGame game(word.getSecretWord(), lives);
238         game.putName(name);
239         game.putState(stages);
240
241         game.play();
242
243         string replay;
244         cout << "You want to play again?(Y/n): ";
245         cin >> replay;
246
247         if (replay[0] == 'y' || replay[0] == 'Y'){
248             continue;;
249         }else{
250             break;
251         }
252     }
253
254     return 0;
```


Hangman Game
-> setting
Your default lives is 7 and maximum lives is 7
You want to config your lives?(Y/n): n
Input your name : John
เลือกว่าจะกำหนดค่าชีวิตหรือไม่
ตั้งชื่อ


Name: John
lives: 7
แสดง ชื่อ
แสดง พลังชีวิต
แสดง stickman
แสดงตัวอักษรที่ทายถูกในตำแหน่งของตัวอักษรนั้นของคำนั้น
Guess a letter:
ทายตัวอักษร


Name: John
lives: 7
+---+
|
|
|
|
|
|
==========
แสดงตัวอักษรที่ทายถูก ในตำแหน่งของตัวอักษรนั้น ของคำนั้น
_ _ o _ y
Guess a letter:


Name: John
Your lives: 7
+---+
|
|
|
|
|
|
=========
s t o r y
Congratulations John Win!!
แสดงคำที่ถูกต้อง
The correct word is 'story'
You want to play again?(Y/n):
ถามว่าต้องการเล่นอีกรอบ หรือไม่

อันนี้คือรูปแบบถ้าแพ้