```
/*
    Name: Snake console game
    Author: nhan tran
    Date: 14/12/16 23:17
    Description: remember to include "NewGraphics.h" header file
*/

#include <iostream>
#include "NewGraphics.h"
#include <ctime>

using namespace std;

/* Macro definition */
#define NOWHERE   0
#define UP        1
#define DOWN      2
#define RIGHT     3
#define LEFT      4
#define WINDOWS_WIDTH  20
#define WINDOWS_HEIGHT 20
#define MAX_LENGTH     100
#define SPEED_LOW      200
#define SPEED_MEDIUM   100
#define SPEED_HIGH     50
#define SPEED_VERYHIGH 30

/* Structure definition */
typedef struct{
        int x[MAX_LENGTH];
        int y[MAX_LENGTH];
        int length;
}Snake;

typedef struct{
    int x;
    int y;
}Food;

/* Global variables */
int _lastDirection;
int _currDirection;
Snake _snake;
Food _food;
bool _gameOver;
int _score;

/* Function prototype */
void init();
void keyHit();
void drawSnake();
void drawFood();
void checkCollision();
void resetGame();
bool eatFood();

/* Main program */
int main(){

    init();
    while(!_gameOver){
        keyHit();
        drawSnake();
        checkCollision();
        Sleep(SPEED_MEDIUM);
    }
}

/* Function definition */

/* Ham: init()
 * Chuc nang: khoi tao game
 * Tham so: khong
 * Tra ve: khong
 */
void init(){
    ShowCur(false);
```

```
78      _gameOver = false;
79      _score = 0;
80      _lastDirection = NOWHERE;
81      _currDirection = NOWHERE;
82      _snake.x[0] = WINDOWS_WIDTH/2 - 2;      _snake.y[0] = WINDOWS_HEIGHT/2;
83      _snake.x[1] = WINDOWS_WIDTH/2 - 1;      _snake.y[1] = WINDOWS_HEIGHT/2;
84      _snake.x[2] = WINDOWS_WIDTH/2;          _snake.y[2] = WINDOWS_HEIGHT/2;
85      _snake.length = 3;
86
87      // Hien thi khung
88      for(int i=0; i<WINDOWS_WIDTH; i++){
89          gotoxy(i, 0);
90          cout << "?";
91      }
92      for(int i=0; i<=WINDOWS_WIDTH; i++){
93          gotoxy(i, WINDOWS_HEIGHT);
94          cout << "?";
95      }
96      for(int i=0; i<WINDOWS_HEIGHT; i++){
97          gotoxy(0, i);
98          cout << "?";
99      }
100     for(int i=0; i<WINDOWS_HEIGHT; i++){
101         gotoxy(WINDOWS_WIDTH, i);
102         cout << "?";
103     }
104
105     // Hien thi thong tin game
106     gotoxy(WINDOWS_WIDTH/2-7, WINDOWS_HEIGHT+2);
107     cout << "powered by SRC";
108
109     // Hien thi text diem so
110     gotoxy(WINDOWS_WIDTH+3, WINDOWS_HEIGHT/2);
111     cout << "Score: ";
112
113     // Hien thi con ran ban dau
114     for(int i=0; i<_snake.length; i++){
115         gotoxy(_snake.x[i], _snake.y[i]);
116         cout << "*";
117     }
118
119     // Hien thi thuc an ban dau
120     drawFood();
121 }
122
123 /* Ham: keyHit()
124  * Chuc nang: kiem tra nhan nut
125  * Tham so: khong
126  * Tra ve: khong
127  */
128 void keyHit(){
129     if(GetAsyncKeyState(VK_UP)){
130         if(_lastDirection!=DOWN) _currDirection = UP; // Kiem tra de tranh tinh trang
    chay nguoc lai
131     }
132     else if(GetAsyncKeyState(VK_DOWN)){
133         if(_lastDirection!=UP) _currDirection = DOWN;
134     }
135     else if(GetAsyncKeyState(VK_RIGHT)){
136         if(_lastDirection!=LEFT) _currDirection = RIGHT;
137     }
138     else if(GetAsyncKeyState(VK_LEFT)){
139         if(_lastDirection!=RIGHT) _currDirection = LEFT;
140     }
141 }
142
143 /* Ham: drawSnake()
144  * Chuc nang: ve con ran ra man hinh
145  * Tham so: khong
146  * Tra ve: khong
147  */
148 void drawSnake(){
149
150     if(_currDirection != NOWHERE){
151         // Xoa duoi ran
152         gotoxy(_snake.x[_snake.length-1], _snake.y[_snake.length-1]);
153         if(_snake.x[_snake.length-1]!=0  // Neu duoi chua cham tuong thi xoa
```

```
154             && _snake.x[_snake.length-1]!=WINDOWS_WIDTH
155             && _snake.y[_snake.length-1]!=0
156             && _snake.y[_snake.length-1]!=WINDOWS_HEIGHT){
157 
158                 if(eatFood()){
159                     _snake.length++; // Tang do dai len 1
160                     drawFood();
161                 }
162                 else{
163                     cout << " ";
164                 }
165             }
166             else{    // Neu duoi cham tuong thi khong xoa
167                 cout << "?";
168             }
169 
170             // Dich ran tu duoi len dau
171             for(int i=_snake.length-1; i>0; i--){
172                 _snake.x[i] = _snake.x[i-1];
173                 _snake.y[i] = _snake.y[i-1];
174             }
175 
176             // Dich dau ran
177             if(    _snake.x[0]!=0   // Kiem tra neu dau ran chua va cham voi tuong
178                && _snake.x[0]!=WINDOWS_WIDTH
179                && _snake.y[0]!=0
180                && _snake.y[0]!=WINDOWS_HEIGHT){
181                 /*
182                 if(_lastDirection!=LEFT && _currDirection==RIGHT){
183                     _snake.x[0]++;
184                 }
185                 else if(_lastDirection!=RIGHT && _currDirection==LEFT){
186                     _snake.x[0]--;
187                 }
188                 else if(_lastDirection!=DOWN && _currDirection==UP){
189                     _snake.y[0]--;
190                 }
191                 else if(_lastDirection!=UP && _currDirection==DOWN){
192                     _snake.y[0]++;
193                 }
194                 */
195                 if(_currDirection==RIGHT){
196                     _snake.x[0]++;
197                 }
198                 else if(_currDirection==LEFT){
199                     _snake.x[0]--;
200                 }
201                 else if(_currDirection==UP){
202                     _snake.y[0]--;
203                 }
204                 else if(_currDirection==DOWN){
205                     _snake.y[0]++;
206                 }
207             }
208             else{  // Neu dau ran va cham voi tuong
209                 if(_snake.x[0]==0){
210                     _snake.x[0] = WINDOWS_WIDTH-1;
211                 }
212                 else if(_snake.x[0]==WINDOWS_WIDTH){
213                     _snake.x[0] = 1;
214                 }
215                 else if(_snake.y[0]==0){
216                     _snake.y[0] = WINDOWS_HEIGHT-1;
217                 }
218                 else if(_snake.y[0]==WINDOWS_HEIGHT){
219                     _snake.y[0] = 1;
220                 }
221             }
222             // Thay doi _lastDirection
223             _lastDirection = _currDirection;
224 
225             // In dau ran moi
226             gotoxy(_snake.x[0], _snake.y[0]);
227             cout << "*";
228         }
229 
230         // Hien thi diem so hien tai
```

```
231         gotoxy(WINDOWS_WIDTH+10, WINDOWS_HEIGHT/2);
232         cout << _score;
233     }
234
235     /* Ham: drawFood()
236      * Chuc nang: ve thuc an ra man hinh
237      * Tham so: khong
238      * Tra ve: khong
239      */
240     void drawFood(){
241         srand(time(NULL));
242         _food.x = rand() % (WINDOWS_WIDTH-2)  + 1;
243         _food.y = rand() % (WINDOWS_HEIGHT-2) + 1;
244
245         gotoxy(_food.x, _food.y);
246         cout << (char)3; // Trai tim
247     }
248
249     /* Ham: checkCollision()
250      * Chuc nang: kiem tra su va cham cua dau ran voi than ran
251      * Tham so: khong
252      * Tra ve: khong
253      */
254     void checkCollision(){
255         for(int i=1; i<_snake.length; i++){
256             if(_snake.x[i] == _snake.x[0]
257             && _snake.y[i] == _snake.y[0]){
258                 _gameOver = true;
259                 resetGame();
260             }
261         }
262     }
263
264     /* Ham: resetGame()
265      * Chuc nang: cho nguoi dung nhan phim SPACE de choi lai tu dau
266      * Tham so: khong
267      * Tra ve: khong
268      */
269     void resetGame(){
270         clearScreen(); // Xoa man hinh
271         gotoxy(0, 1);
272         cout << "GAME OVER";
273         gotoxy(0, 3);
274         cout << "Final score: " << _score;
275         gotoxy(0, 5);
276         cout << "Press ENTER to play again, ESC to escape!";
277         while(1){
278             if(GetAsyncKeyState(VK_RETURN)){
279                 clearScreen();
280                 _gameOver = false;
281                 init();
282                 break;
283             }
284             if(GetAsyncKeyState(VK_ESCAPE)){
285                 break;
286             }
287         }
288     }
289
290     /* Ham: eatFood()
291      * Chuc nang: kiem tra ran da an thuc an hay chua
292      * Tham so: khong
293      * Tra ve: TRUE or FALSE
294      */
295     bool eatFood(){
296         if(_snake.x[0]==_food.x && _snake.y[0]==_food.y){
297             _score++;
298             return true;
299         }
300         return false;
301     }
302
```