

Hra SNAKE

Vysvetlenie:

Kód vytvorí jednoduchú hru s hadíkom. Vykreslí sa mapa 20x20, hadík a ovocie, ktoré bude meniť svoje súradnice po zjedení. Hadík sa bude jedením ovocia zväčšovať, pričom s každým zjedením ovocia sa pripočíta 10 bodov, ktoré sa budú vypisovať pod mapou. Hra sa skončí, keď hadík narazí sám do seba, alebo narazí do okrajov mapy, alebo keď užívateľ stlačí klávesu X.

Návod na spustenie:

Hra sa vo VS Code terminály nezobrazuje správne, preto ju len skompilujem cez Run Build Task. Následne stlačíme CTRL+SHIFT+C a otvoríme cmd vo Windowse, kde kód spustíme príkazom ***nazov_programu.exe***.

Postup:

Do priloženého kódu dopíšeme základné nastavenia a funkcie pre vykresľovanie, pre spracovanie vstupu z klávesnice a pre nastavenia logiky hry.

1. Zahrnieme potrebné knižnice a zdefinujeme základné globálne premenné/konštanty, ktoré budeme potrebovať v celom kóde. Porozmýšľajte čo si potrebujeme zdefinovať.

```
#include <iostream>
#include <conio.h>
#include <windows.h>

bool gameOver;

//premenne a konstanty

int flag;
```

Aby sme zjednodušili logiku medzi smerom pohybu a vstupom z klávesnice, použijeme premennú *flag*. Jej číselná hodnota bude určovať konkrétny smer.

2. Vo funkcií Setup si zdefinujeme počiatočné nastavenia ako napr. začiatočné súradnice hadíka. Znovu porozmýšľajte čo je potrebné nastaviť.
3. Do funkcie Draw doplňte kód, ktorý vykreslí mapu, hadíka a ovocie. Pre ohraničenie mapy použite znak '#', pre začiatok hadíka 'O', pre dopĺňajúci sa chvost 'o', pre ovocie 'F' a pre vnútro mapy ' ' (medzeru).
Na začiatku funkcie použijeme funkciu *system("cls")*. To spôsobí, že sa mapa bude všetko neustále vykresľovať nanovo. Po dokončení funkcie Draw skúste, ako funguje hra bez použitia *system("cls")*.
4. Do funkcie Input zaznamenáme klávesy na ovládanie hadíka. Pre pohyb do smerov použite klávesy WSAD a klávesu X pre ukončenie hry. Použijeme funkcie *_kbhit()*, ktorá kontroluje či bola stlačená klávesnica a funkciu *_getch()*, ktorá zaznamená stlačenú klávesu. Viac o fungovaní týchto funkcií nájdete na tomto [linku](#).

Tip: pre priradenie klávesy k smeru použite namiesto podmienok *switch*. Switch má jednoduchý zápis a vykonáva sa rýchlejšie. Používa sa len s celými číslami, znakmi alebo enumeráciami. Pozri si tento [link](#) a zisti viac.

5. Vo funkcií Logic už máme vložený kód, vďaka ktorému chvost hadíka nasleduje hlavu. Skúste sa zamyslieť nad týmto kódom. Ďalej si doplníme do funkcie switch pohyb hadíka podľa vybraného smeru. Následne zdefinujeme čo sa stane, keď hadík trafí okraj mapy alebo svoj chvost. Nakoniec spravíme pripočítavanie skóre a vytvorenie nového ovocia, keď ho hadík zje.
6. Nakoniec zavoláme všetky funkcie v hlavnej funkcií main.

Zdrojový kód:

```
#include <iostream>
#include <conio.h>
#include <windows.h>

bool gameOver;

//premenne a konstanty

int flag;

void Setup();
void Draw();
void Input();
void Logic();

int main()
{
    Setup();
    while (!gameOver)
    {
        Draw();
        Input();
        Logic();
        Sleep(10);
    }

    return 0;
}

void Setup()
{
    //premenne a konstanty pre zakladne nastavenia
}

void Draw()
```

```

{
    system("cls");

    for () {
        for () {
            //vykreslenie hranic mapy, hadika, ovocia, a vnutra mapy
        }
    }
    std::cout << std::endl;

}

std::cout << std::endl;

//vypisanie skore
}

void Input()
{
    //priradenie WSAD klaves k smerom UP, DOWN, LEFT, RIGHT a klavesy X
    //pre ukoncenie hry

    if (_kbhit()) {

        switch (_getch())
        {
            case klavesa1:
                //prikaz 1
                break;
            case klavesa2:
                //prikaz 2
                break;
            case klavesa3:
                //prikaz 3
                break;
            case klavesa4:
                //prikaz 4
                break;
            case klavesa5:
                //prikaz 5
                break;
        }
    }
}

void Logic()
{

```

```

int prevX = tailX[0];
int prevY = tailY[0];
int prev2X, prev2Y;
tailX[0] = x;
tailY[0] = y;

for (int i = 1; i < nTail; i++) {      //kod, aby chvost nasledoval hlavu
    prev2X = tailX[i];
    prev2Y = tailY[i];

    tailX[i] = prevX;
    tailY[i] = prevY;

    prevX = prev2X;
    prevY = prev2Y;
}

switch (flag)
{
case smer1:
    //prikaz 1
    break;
case smer2:
    //prikaz 2
    break;
case smer3:
    //prikaz 3
    break;
case smer4:
    //prikaz 4
    break;
default:
    break;
}

//co sa stane, ked hadik narazi do hranic mapy

for () {
    //co sa stane, ked hadik narazi do svojho chvosta
}

//co sa stane, ked hadik zje ovocie
}

```

BONUS: Skúste podmienku s ukončením hry po náraze do hraníc upraviť tak, aby hadík vedel prechádzať cez hranice mapy, teda aby sa objavil na druhej strane.

Zdroj: [youtube](#) a [stránka](#)