

**Celkový čas**  
4-6 hod

**Dovednosti:**

tvorba 2D postavy pro hru  
nastavení kostí a končetin  
animace postavy  
vytvoření skriptů postavy

Střední průmyslová škola elektrotechnická  
a Vyšší odborná škola Pardubice  
Karla IV. 13, Pardubice 530 02  
www.spse.cz reslova@spse.cz



# Project Erasmus+ CIT

Community of Information Technology

Art Graphic Tutorial

*Tvorba postavy pro 2D hru*

**C**

**SPŠE a VOŠ Pardubice, CZECH REPUBLIC**

**I**

**T**

International student practice in companies

## Cíl práce:

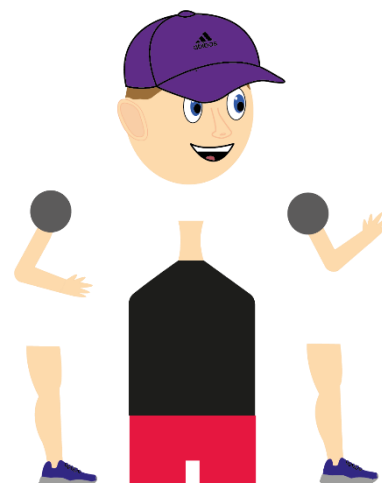
- Vytvoření 2D postavy pro platformovou hru
- Nastavení kostí a končetin postavy (a příslušné vztahy)
- Vytvoření 3 animací – idle, běh a skok
- Vytvoření skriptu pro 2D ovládání postavy
- Vytvoření skriptu pro změnu stavu postavy

## Použitý software:

- Inkscape (vektorová grafika)
- Adobe Illustrator (alternativa k Inkscape)
- Unity (tvorba hry)
- Visual Studio Code (programování skriptů – C#)

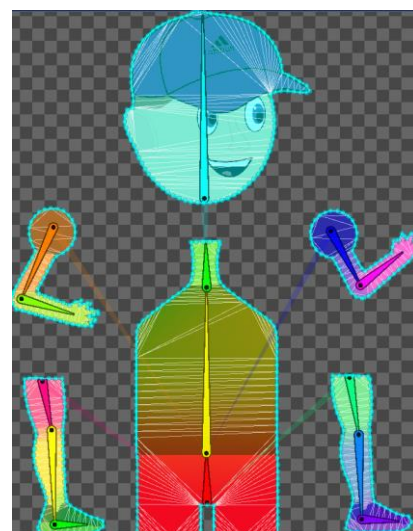
## Vytvoření postavy

- Prvním krokem je vytvoření všech částí postavy
- Postava se skládá z hlavy, trupu, rukou a nohou
- Končetiny jsou vytvořeny tak, aby s nimi bylo později možné plynule pohybovat při animaci
- Musíme také brát v potaz fakt, že pracujeme s 2D prostředím, takže postava musí být správně nasměrovaná
- Jednotlivé části těla při exportu vybereme samostatně (abychom poté mohli ve Sprite editoru nastavit kosti, končetiny a jejich vzájemné vztahy)



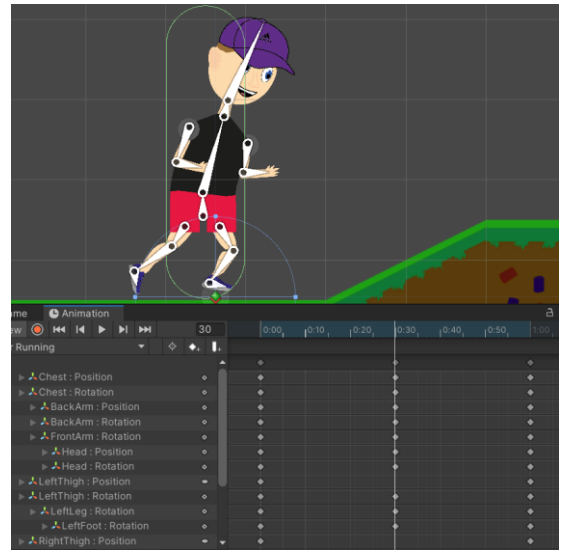
## Vytvoření kostí postavy

- Po importování PNG obrázku postavy je zapotřebí vytvořit kosti a přidat k nim končetiny
- K nastavení těchto vlastností využijeme Sprite Editor, který má mnoho užitečných nastavení
- Nejdříve vytvoříme základní kosti (defacto páteř), pomocí kterých budeme vytvářet vztahy na další části těla
- Poté navážeme vztah na hlavu, ruce a nohy pomocí relačních spojů
- Po vytvoření všech vztahů použijeme nastavení Auto Weights, které automaticky rozprostře váhu postavy



# Animace postavy

- Po vytvoření kostry a nastavení váhy postavy se přesuneme na animace
- Unity má svůj vlastní Animator, pomocí kterého vytvoříme 3 animace
- První je animace pro nečinný stav postavy (idle)
- Druhá je animace pro běh postavy
- Třetí je animace pro skok postavy
- Po vytvoření animace přejdeme do vztahového editoru, kde nastavíme podmínky přechodů mezi animacemi



## Programování skriptů postavy

- Po vytvoření animací nám zbývá vytvoření skriptů pro postavu
- Nejdříve napíšeme skript, který zajistí správný pohyb postavy ve 2D prostředí
- Zde například kontrolujeme, zda postava stojí na zemi nebo jestli například stojí
- Druhý skript je pro kontrolu stavu postavy, protože musíme sledovat několik hodnot
- Pokud se například postava objeví na úrovni  $Y > -10$ , tak to znamená okamžitou smrt
- Dále také monitorujeme změnu zdraví (healing/damage)

```
90     void FixedUpdate()
91     {
92         h = Input.GetAxis("Horizontal");
93         myRigidbody.velocity = new Vector2(h * moveSpeed, myRigidbody.velocity.y);
94
95         if(myRigidbody.velocity.x > 0 && transform.eulerAngles.y == 180)
96         {
97             transform.rotation = Quaternion.Euler(0, 0, 0);
98         }
99         else if (myRigidbody.velocity.x < 0 && transform.eulerAngles.y == 0)
100        {
101            transform.rotation = Quaternion.Euler(0, 180, 0);
102        }
103    }
104 }
```