

# C++ questions

## Q1

I valori finali sono i seguenti:

$a=5$  in quanto non è affetto da altre operazioni se non appunto quella appena scritta.

$b=8$  in quanto il risultato dell'operazione  $(a+2+1*0)$  è 7, attraverso l'operazione di incremento svolta da C, questo valore si eleva di 1.

$c=7$ , in quanto " $c=b++$ " copia il valore e successivamente esegue l'operazione di incremento. Nel caso in cui questa fosse stata " $c=++b$ ", invece, ci sarebbe stata prima l'incremento e poi la copia del valore, il risultato quindi sarebbe stato  $c=8$ .

## Q2

La funzione che compilerebbe nella funzione mostrata nella domanda sarebbe la prima:

Il primo è una "named structure" e consente di utilizzare lo struct come se fosse una vera e propria variabile.

Nel secondo caso invece, lo struct, vista la modalità nella quale viene dichiarata, la variabile viene dichiarata subito dopo lo struct, non si ha quindi la possibilità di trattarlo come se fosse una variabile.

Nell'immagine mostrata si assume un comportamento adatto al primo dei due esempi in quanto viene dichiarata la variabile di "tipo cat" col nome "newCat".

nel secondo caso sarebbe stato corretto scrivere:

```
void spawnCat(){  
    cat.weight = 5  
    cat.colour = Black;  
}
```

## Q3

Per istanziare una classe mediante Istanched(?)

## Q4

Per riuscire a sapere se due oggetti si stanno guardando c'è bisogno dei seguenti dati:

i due oggetti e la loro rotazione rispetto al mondo;

Il vettore della direzione verso la quale stanno guardando;

Nel caso in cui la differenza di rotazione sull'asse delle Y fra i due oggetti è di 180 gradi e il vettore scalare fra i due vettori rappresentanti la direzione verso la quale stanno guardando è pari a 0, allora i due si stanno guardando.

## Q5

Avrei svolto il binding del pulsante nella seguente maniera, inoltre avrei fatto riferimento alla classe che permette ad unreal di controllare gli Input:

```
void Esercizio::SetupPlayerInputComponent(class UInputComponent* PlayerInputComponent)
{
    PlayerInputComponent->BindAction("Jump", IE_Pressed, this, &ACharacter::Jump);
}
```