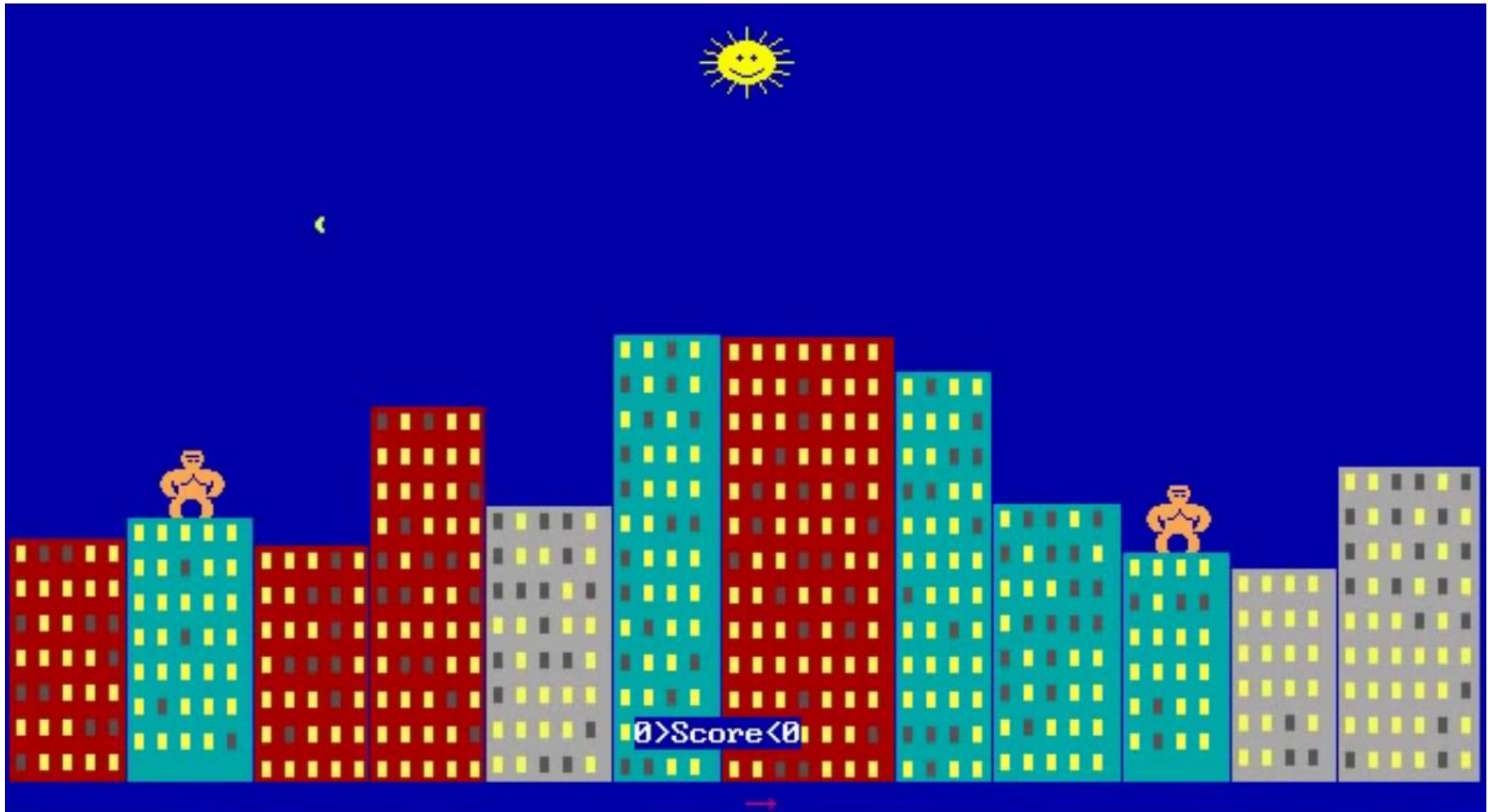


Stage Scuole Superiori

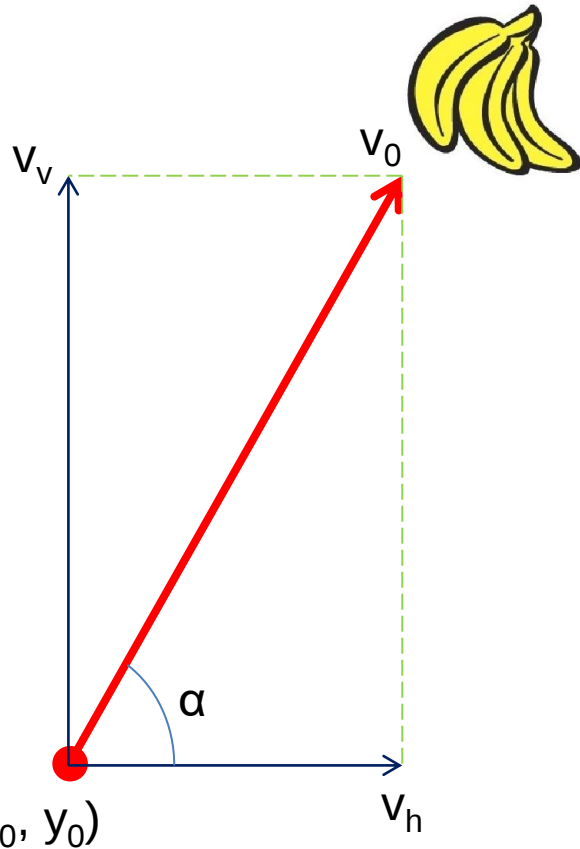
Videogame: Gorillas

Giorgio Leonardi

Idea di partenza:



Problema 1: traiettoria banana



$$V_h = V_0 \cos(\alpha)$$
$$V_v = V_0 \sin(\alpha)$$

$$X = X(t) = X_0 + (V_0 \cos(\alpha))t$$

$$Y = Y(t) = Y_0 + (V_0 \sin(\alpha))t - \frac{1}{2}gt^2$$



Blocchi per traiettoria della banana

- **BANANA X (x_0 , v_0 , α , t)**
 - return $x_0 + (v_0 \cdot \cos(\alpha) * t)$
- **BANANA Y (y_0 , v_0 , α , t)**
 - return $y_0 + (v_0 \cdot \sin(\alpha) * t) - (9.8 * t * t) / 2$

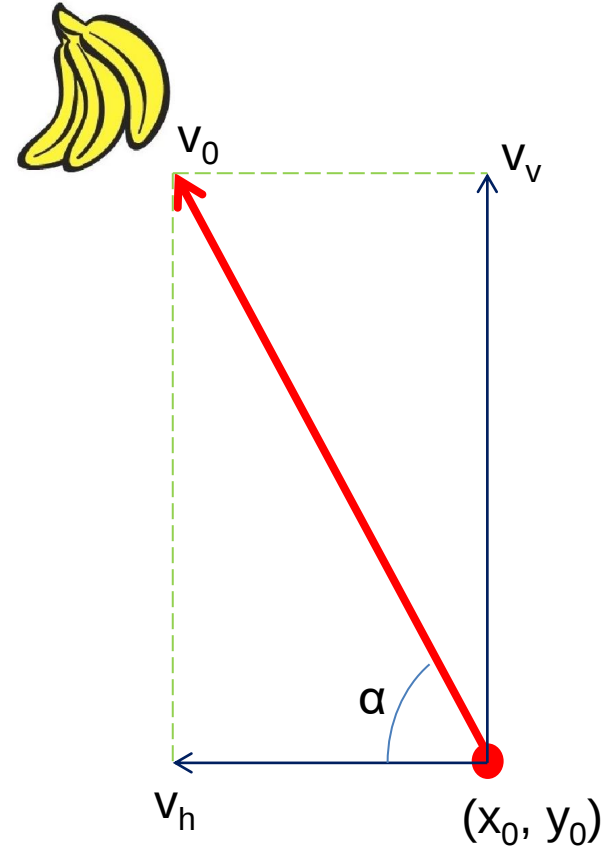


Problema 1: traiettoria opposta

$$V_h = V_0 \cos(\alpha)$$
$$V_v = V_0 \sin(\alpha)$$

$$X = X(t) = X_0 + (-V_0 \cos(\alpha))t$$

$$Y = Y(t) = Y_0 + (V_0 \sin(\alpha))t - \frac{1}{2}gt^2$$



Blocchi per ogni traiettoria

- *Ho in input la «direzione», che vale:*
 - *+1 se il tiro è da sinistra a destra*
 - *-1 se il tiro è da destra a sinistra*
- **BANANA X (x_0 , v_0 , α , t , direzione)**
 - **return $x_0 + (\text{direzione} * v_0 * \cos(\alpha)) * t$**
- **BANANA Y (y_0 , v_0 , α , t)**
 - **return $y_0 + (v_0 * \sin(\alpha) * t) - (9.8 * t * t) / 2$**



Primo elemento del gioco:

Lancio della banana

- **Sprite «banana»:** dati X_0 , Y_0 , V_0 , α e direzione
- **Quando ricevo: «lancia banana»:**
 - $T = 0$
 - Finchè non tocco un bordo esterno, o il bordo inferiore:
 - Calcolo $X = \text{BANANA } X(X_0, V_0, \alpha, T, \text{direzione})$
 - Calcolo $Y = \text{BANANA } Y(Y_0, V_0, \alpha, T)$
 - Mi posiziono a coordinate (X, Y)
 - Ruoto leggermente il costume, per animare il lancio della banana
 - $T = T + \Delta t$



Secondo elemento del gioco: il gorilla 1

- **Gorilla 1 è uno sprite con 2 costumi: una posa «normale» e una posa per il «lancio» della banana**
- **Quando il gorilla è attivato:**
 - **chiede velocità e angolo del lancio al giocatore 1**
 - **lancia la banana mandando un messaggio «lancia banana», indicando la direzione verso destra**
 - **Per tutta la durata del lancio, il gorilla assume il costume «lancio», per tornare «normale» quando la banana termina la sua corsa**



Gorilla 1

- Quando ricevo «gorilla 1»:
 - $V0$ = chiedo velocità all'utente
 - Alpha = chiedo angolo di tiro all'utente
 - $X0$ = posizione X del gorilla 1
 - $Y0$ = posizione Y del gorilla 1 + 60 pixel (il lancio parte da sopra la testa del gorilla)
 - direzione= +1 (lancio da SX a DX)
 - Passo al costume «lancio»
 - Invio il messaggio «lancia banana» e attendi la conclusione
 - Passo al costume «normale»



Gorilla 2

- **E' una copia del gorilla 1, con alcune differenze:**
 - **Quando lancia la banana, la direzione è da DX a SX**
 - **I due costumi sono ribaltati in orizzontale rispetto al gorilla 1**
 - **L'attivazione del gorilla 2 parte al ricevimento del messaggio «gorilla 2», non più «gorilla 1», che invece attiva l'altro gorilla**



Turni di gioco

- Il gioco è a turni, quindi i due gorilla si alternano. Abbiamo quindi bisogno di:
 - una variabile «tocca al gorilla», che indichi chi deve effettuare la prossima giocata
 - un meccanismo che alterni i giocatori:
 - cambiando di volta in volta il valore della variabile
 - Mandando, ad ogni turno, il messaggio che attiva il giocatore corretto



Turni di gioco

- Possiamo comandare i turni da uno script dello «stage», che è un componente super-partes del nostro software. Il gioco può iniziare indifferentemente dal gorilla 1 o dal gorilla 2
- Quando ricevo il messaggio «inizio gioco»:
 - «tocca al gorilla» è un numero a caso tra 1 e 2
 - Per sempre:
 - Se «tocca al gorilla» è 1, allora:
 - Invia a tutti «gorilla 1» e attendi la fine della giocata
 - «tocca al gorilla» = 2 (passo il gioco all'altro gorilla)
 - Altrimenti:
 - Invia a tutti «gorilla 2» e attendi la fine della giocata
 - «tocca al gorilla» = 1 (passo il gioco all'altro gorilla)



Vittoria!

- **Quando la banana colpisce un gorilla:**
 - Il gorilla colpito esplode!
 - L'altro gorilla festeggia e vince!
- **Non importa chi ha lanciato la banana: un gorilla può lanciarsi la banana in testa e perdere**
- **Abbiamo bisogno di:**
 - **Modificare il codice della traiettoria della banana, per capire quando colpisce uno dei due gorilla**
 - Se la banana colpisce il gorilla 1, lancia un messaggio «esplodi 1». Se, invece, colpisce il gorilla 2, invia «esplodi 2»
 - **Gorilla 1:**
 - se riceve «esplodi 1», allora ha perso. Quindi parte un'animazione di esplosione e scompare
 - Se riceve «esplodi 2», allora ha vinto! Quindi parte l'animazione di una danza di vittoria e comunica la sua vittoria
 - **Gorilla 2: come il gorilla 1, ma rispondendo al contrario ai due messaggi**



Gorilla 1: vittoria o sconfitta

- Quando ricevo «esplodi 1»:
 - Passa al costume «planet 2»
 - Ingrandisci la dimensione
 - Riduci la dimensione per 3 volte, per simulare una sparizione
 - Nasconditi

- Quando ricevo «esplodi 2»:
 - Passa al costume «lancio»
 - Cambia per 3 o 4 volte il verso «destra-sinistra» per imitare una danza
 - Dire «Vittoria!»

Il gorilla 2 fa le stesse cose a messaggi invertiti

Aggiungiamo il vento

- Il vento modifica la traiettoria della banana e aggiunge un po' di imprevedibilità al gioco
- Il vento soffia solo in orizzontale, modificando la gittata del tiro (la velocità orizzontale), ma non l'altezza della parabola
- Abbiamo bisogno di:
 - Una variabile «vento», che indica la velocità del vento
 - Vento > 0 : il vento soffia da SX verso DX
 - Vento < 0 : il vento soffia da DX verso SX
 - Rendere visibile direzione e forza del vento ai giocatori
 - Modificare il blocchetto «BANANA X», per incorporare l'effetto del vento nella traiettoria orizzontale della banana

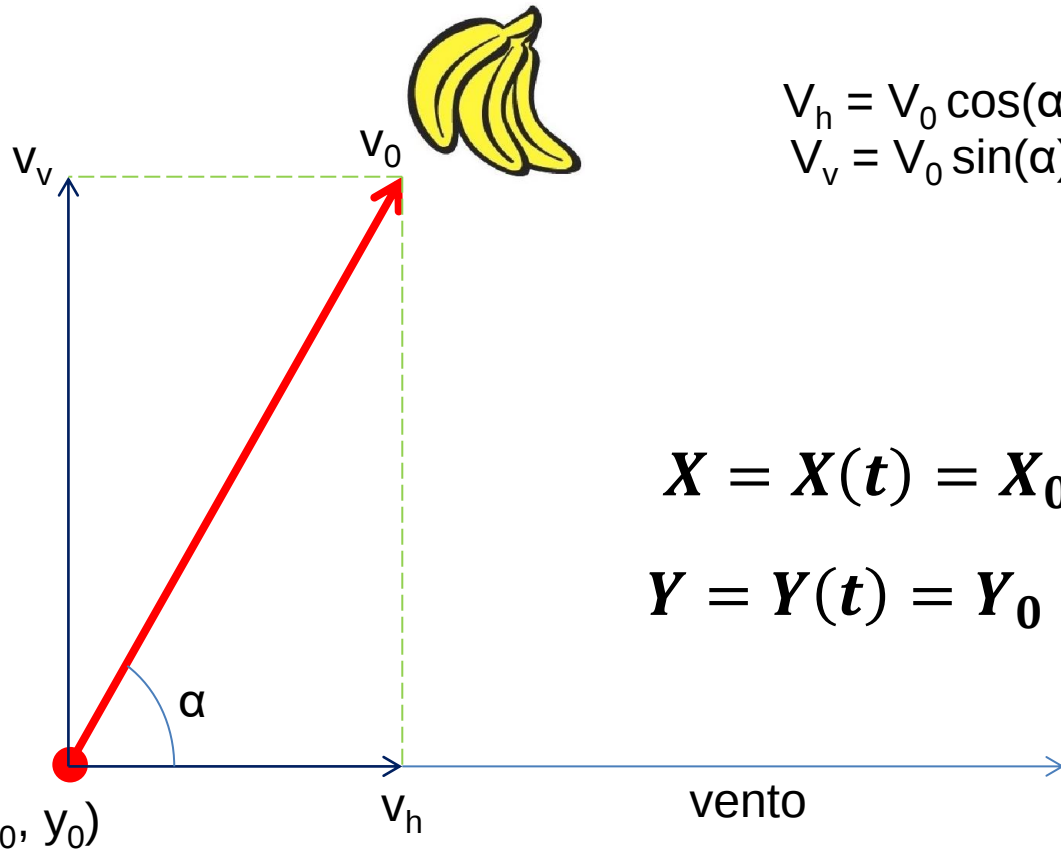


Sprite «vento»

- **Sprite a forma di freccia. Il suo sketch inizializza il vento a inizio gioco, e gira la freccia a seconda della direzione del vento**
- **Quando ricevo «inizio gioco»:**
 - **vento= numero casuale da -30 a 30 (ad esempio)**
 - **Se vento > 0, allora passa al costume «freccia a destra»**
 - **Altrimenti, se vento < 0, passa al costume «freccia a sinistra»**
 - **Altrimenti, passa al costume «no vento»**
- **Per rendere il gioco ancora più imprevedibile, il vento potrebbe cambiare ad ogni turno del gioco, invece di restare fisso per tutta la partita**



Traiettoria banana con vento



$$V_h = V_0 \cos(\alpha)$$
$$V_v = V_0 \sin(\alpha)$$

$$X = X(t) = X_0 + (\pm V_0 \cos(\alpha) + \text{vento})t$$
$$Y = Y(t) = Y_0 + (V_0 \sin(\alpha))t - \frac{1}{2}gt^2$$

- **BANANA X** (x_0 , v_0 , α , t , direzione, vento)
 - return $x_0 + (\text{direzione} * v_0 * \cos(\alpha) + \text{vento}) * t$



Grattacieli

- I due gorilla battagliaano in una giungla urbana, fatta di grattacieli di altezza casuale ad ogni partita.
- La banana ferma la sua corsa quando incontra un grattacielo
- Costruiamo i grattacieli partendo da un singolo sprite: «grattacieli», che rappresenta un singolo piano, che verrà clonato, impilandosi per ottenere un grattacielo di una certa altezza
- Finito di impilarsi per costruire un grattacielo, lo sprite si sposta di fianco per creare un altro grattacielo, sempre continuando a clonarsi
- Termineremo le clonazioni quando avremo riempito lo schermo di grattacieli



Costruzione grattacieli

Dimensione di un piano: 50h x 20v



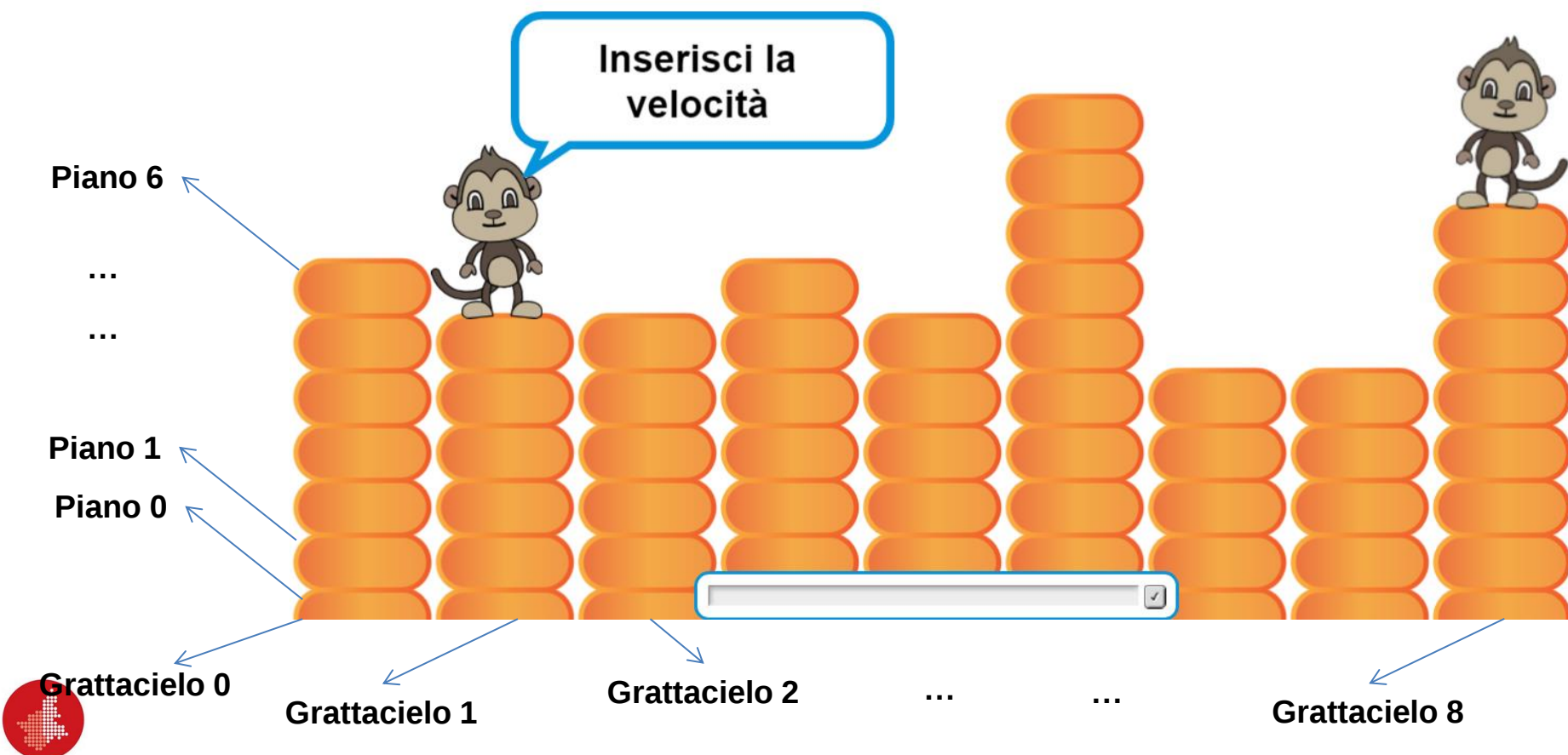
Posizione del p-esimo piano, del g-esimo grattacielo sullo schermo:

vento 20

Posizione verticale: $(p * 20) - 180$

Posizione orizzontale: $(g * 53) - 210$

La posizione orizzontale è un più «larga» per lasciare un minimo di spazio tra grattacieli



Costruzione grattacieli: algoritmi

POSIZIONE EDIFICIO (edificio g)

– Return ($g \cdot 53$) - 210

ALTEZZA PIANO (piano p)

– Return ($p \cdot 20$) - 180

- **Sprite «grattacieli»:** quando è premuta bandiera verde:
 - Mi porto in background e dimensione al 42%
 - Per codiceEdificio da 0 a 8: (per ogni edificio, numerati da 0 a 8)
 - altezzaEdificio= numero a caso tra 5 e 10 piani
 - Per codicePiano da 0 a altezzaEdificio: (per ogni piano, numerato da 0 all'altezza scelta)
 - Crea clone di «grattacieli»
 - Posizionati a $x = \text{POSIZIONE EDIFICIO}(\text{codiceEdificio})$ e $y = \text{ALTEZZA PIANO}(\text{codicePiano})$
 - Avverti «gorilla 1» che deve posizionarsi su un grattacielo: invia messaggio «piazza gorilla 1»



Posizione gorilla 1

- Il gorilla 1 deve scegliere uno dei tre grattacieli a sinistra e «sedervisi» sopra.
- Quando ricevo «piazza gorilla 1»:
 - grattacielo = numero a caso da 0 a 2 (codice)
 - $X = \text{grattacielo} * 56 - 220$ (lo sposto un po' a SX rispetto alla formula della posizione dei grattacieli)
 - $Y = 180$ (lo faccio partire dall'alto)
 - Finchè non sto toccando «grattacieli»:
 - Cambia y di -2 (scendo fino a toccare il grattacielo che ho sotto i piedi)
 - Passo al costume «normale»
 - Avverto il gorilla 2 che deve piazzarsi anche lui. Invia messaggio «piazza gorilla 2»



Posizione gorilla 2

- Il gorilla 2 deve scegliere uno dei tre grattacieli a destra e «sedervisi» sopra.
- Quando ricevo «piazza gorilla 2»:
 - grattacielo = numero a caso da 0 a 2 (codice)
 - $X = 210 - \text{grattacielo} * 56$ (ribalto la formula di prima, perché il grattacielo è a destra!)
 - $Y = 180$ (lo faccio partire dall'alto)
 - Finchè non sto toccando «grattacieli»:
 - Cambia y di -2 (scendo fino a toccare il grattacielo che ho sotto i piedi)
 - Passo al costume «normale»
 - I gorilla sono piazzati e pronti: faccio partire il gioco. Invia messaggio «inizio gioco»



Il gioco è quasi finito!

**In questa versione si può giocare
con un amico**



Modifica 1: Gioca a 1 giocatore

- Modifichiamo il gorilla 2 per farlo giocare da solo quando viene chiamato:
- Il gorilla 2 si tiene due variabili private: «mioV0» e «mioAlpha», che mantiene per memorizzare la sua mossa, e per adattarla nel caso sbagliasse
- Inizia con $\text{mioV0} = 45 \text{ m/s}$ e con $\text{mioAlpha} = 45 \text{ gradi}$, e prova a tirare così
- Se il tiro è sbagliato, uso questa strategia standard:
 - se il tiro è corto, provo ad aumentare la velocità di 20 m/s e anche l'angolo (perché potrei aver colpito il grattacielo che ho davanti!)
 - Se il tiro è lungo, allora l'angolo potrebbe andare bene e diminuisco solo la velocità (di soli 7 m/s per inserire asimmetria nella strategia, altrimenti potrei finire continuamente lungo/corto/lungo/corto senza mai avvicinarmi al bersaglio)



Questa strategia è in un nuovo blocco privato «aggiusta la mira»

Richieste a voi

- Il gioco così completato si presta a un certo numero di modifiche per migliorarlo, che vi chiediamo per ottenere le 3 ore di Alternanza come frequenza di questa attività di laboratorio
- Punto 1: Il gioco si può giocare su diversi pianeti, modificando la forza di gravità: inserire la possibilità di poter giocare su diversi pianeti (Marte, la Luna, ecc..)
 - cercate i valori di forza di gravità dei pianeti che volete offrire nel gioco e date la possibilità di scegliere su che pianeta giocare, prima di iniziare la partita vera e propria
- Punto 2: il vento non deve soffiare solo in orizzontale, ma in una direzione diagonale (creando un vettore simile a quello del lancio della banana).
 - Scegliete una direzione qualsiasi del vento (forza e angolo), e modificate anche il blocchetto della posizione verticale (BANANA Y) per modificare il tragitto della banana nel modo giusto
- Punto 3: il blocchetto «aggiusta la mira» fa giocare da solo il gorilla 2, ma secondo una strategia molto grezza. Pensate da voi una strategia per rendere il gorilla 2 più «intelligente» e aggiustare meglio la mira
 - magari non guardate solo se la banana cade prima o dopo il gorilla 1, ma potete usare la distanza tra banana e gorilla per aggiustare meglio velocità e angolo
 - Se la banana finisce la traiettoria troppo vicino al gorilla 2, forse ha preso in pieno un grattacielo alto nei paraggi, per cui conviene alzare di più la mira
 - ... ecc ecc. Pensate a una strategia vostra, e inseritela nel blocchetto «aggiusta la mira»



Consegna

- **Consegnate il gioco finale, completo della vostra soluzione alle 3 modifiche richieste, con una relazione non lunga, ma che contiene la spiegazione delle vostre soluzioni**
- **La consegna va fatta nello slot che abbiamo messo a disposizione insieme ai video e alle slide del progetto 2**

