

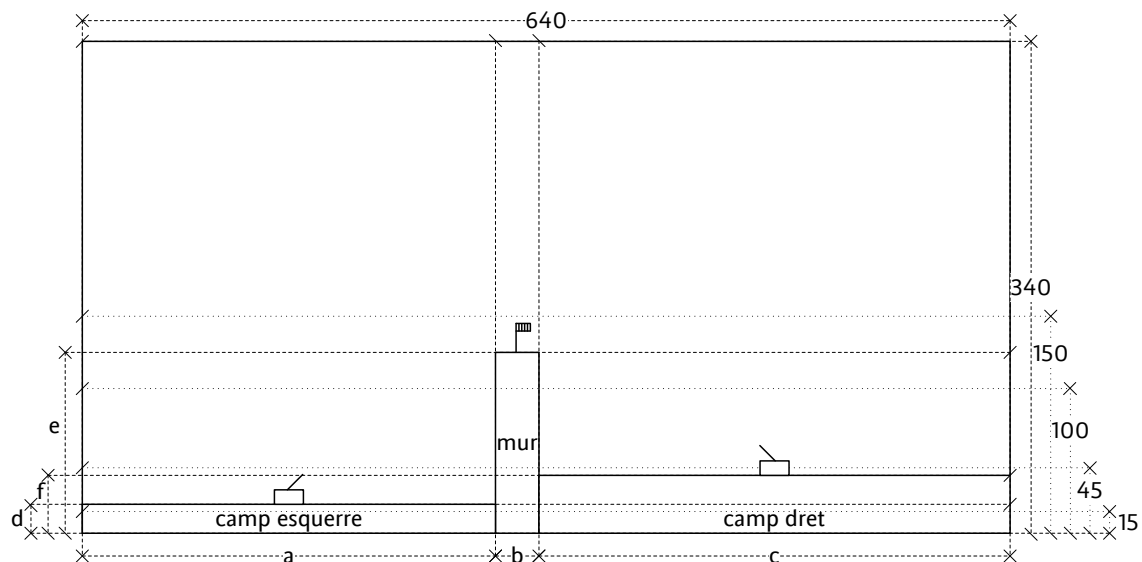
Enunciat de la pràctica final de LISP. Curs 2023-24

Descripció general

Utilitzarem una superfície de dibuix de 640×340 px. Hi podem trobar tres parts: un camp que ocupa aproximadament la meitat esquerra de la superfície; un mur enmig; i un altre camp que ocupa aproximadament la meitat dreta de la superfície. Les mides de cada part són aleatòries amb certes restriccions:

- ◆ El mur del mig té una amplada b d'entre 20 i 40 píxels i una altura e d'entre 100 i 150 píxels.
- ◆ Els camps es reparteixen horitzontalment (a i c) la resta dels 640 píxels, amb una diferència màxima de 40 píxels, i verticalment (d i f) entre 15 i 45 píxels.

Un exemple de distribució horitzontal seria la següent: un mur amb amplada $b = 30$ píxels; això deixa 610 píxels pels camps. Amb la diferència mínima (0 píxels), serien 305 píxels per a cadascun. Amb la diferència màxima de 40 píxels, això deixaria un camp esquerre d'amplada $a = 285$ píxels i un camp dret d'amplada $c = 325$ píxels (o al revés). O qualsevol valor entremig.



Els canons són rectangles de 20×10 píxels, amb una línia, que surt del mig de la part superior, orientada segons l'angle cap on apunten i tan llarga com la velocitat inicial dels seus disparos. Els canons inicialment comencen a un punt aleatori del terç central de cada camp (és a dir, si el camp esquerre té una amplada $a = 285$ píxels, el canó hauria de començar amb una posició horitzontal entre 95 i 190), apunten cap a 45° cap al mig de l'escena i disparen amb una velocitat inicial de 20 px/s.

A la part superior del mur s'hi ha de mostrar el pal d'una bandera de 20 píxels d'alt, que indica la força del vent (entre -5 i 5) segons la quantitat de línies que es pinten a la bandera i el sentit del vent segons cap on s'ha pintat la bandera. Si la força del vent és 0, es considera que no hi ha vent i s'ha de pintar el pal però no la bandera.

Totes les dades (p. ex.: posicions i mides de les parts del dibuix, posicions i orientacions dels canons, força del vent, etc) es guardaran com a propietats de símbols (p. ex.: escenari, cano1, cano2, etc).

La interacció es farà a través de pulsacions de tecles del teclat, obtingudes a partir de (`get-key`). No obstant això, es recomana implementar també un bucle que avaluï l'entrada de text per cridar les funcions internes per així poder provar millor aquestes funcions individualment. Feis servir aquestes tecles:

Acció	Canó esquerre	Canó dret
Mou cap a l'esquerra	A	J
Mou cap a la dreta	D	L
Puja canó	W	I
Baixa canó	S	K
Menys potència	Q	O
Més potència	E	U
Dispara	F	H
Surt	ESC	

Es recomana definir aquestes funcions:

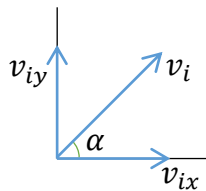
- ◆ (`inicia`)
 - ◇ Estableix les propietats necessàries per representar l'escenari i els seus elements. Comença el bucle principal del joc, que acaba en introduir una tecla específica o quan s'ha destruït un canó.
- ◆ (`bucle`)
 - ◇ Obté l'entrada de l'usuari, interpreta l'opció, du a terme l'acció desitjada, repinta l'escenari i torna a començar.
- ◆ (`pinta`)
 - ◇ Esborra la pantalla i torna a dibuixar l'escenari amb tots els seus elements segons les seves propietats.
- ◆ (`puja cano graus`) i (`baixa cano graus`)

- ◇ Permeten augmentar o disminuir l'orientació d'un canó en una quantitat de graus.
- ◆ (augmenta cano velocitat) i (disminueix cano velocitat)
 - ◇ Permeten augmentar o disminuir la velocitat inicial del dispar d'un canó en una quantitat de px/s.
- ◆ (dispara cano)
 - ◇ Donat un canó, calcula i dibuixa la trajectòria del projectil disparat, segons la posició del canó, l'angle del dispar i la seva velocitat inicial.
 - ◇ El càlcul s'atura quan el projectil pega a terra, al mur del mig, a un canó o si surt per un lateral.

Trajectòries

Com va estudiar Galileo Galilei, la trajectòria descrita per un projectil és una paràbola que és la combinació de dos moviments: un uniforme seguint l'eix x i un uniformement accelerat (acceleració constant $g = -9.8 \frac{px}{s}$) seguint l'eix y .

Donat un tir amb angle α i velocitat inicial v_i , podem obtenir les velocitats inicials horitzontal v_{ix} i vertical v_{iy} segons l'equació (1).



$$\begin{aligned} v_{ix} &= v_i \cdot \cos \alpha \\ v_{iy} &= v_i \cdot \sin \alpha \end{aligned} \quad (1)$$

A cada passa de la simulació, la velocitat es calcula segons l'equació (2) en funció de les velocitats anteriors (v_0) i del temps que ha passat (t); mentre que el moviment (x i y) de la bala es pot trobar segons l'equació (3) en funció d'una posició anterior (x_0 i y_0), d'aquestes velocitats (v_x i v_y) i del temps que ha passat (t). En els dos casos l'acceleració (a) de la gravetat (g) hi té efecte. És important decidir un interval de temps prou petit per calcular la trajectòria amb un detall suficient, però prou gros perquè pugui calcular-se recursivament sense desbordar la pila.

$$\begin{aligned} v_x &= v_{0x} \\ v_y &= v_{0y} + a \cdot t \end{aligned} \quad (2)$$

$$\begin{aligned} x &= x_0 + v_x \cdot t \\ y &= y_0 + v_y \cdot t + \frac{1}{2} \cdot a \cdot t^2 \end{aligned} \quad (3)$$

Per simular el vent, podeu canviar la fórmula de v_x de (2) per la següent: $v_x = v_{0x} + a \cdot t$, on a és la força del vent entre -5 i 5. Podeu mantenir les fórmules de (3).

Per saber si el projectil impacta amb algun element, teniu dues alternatives: la primera és comprovar si, a cada moment, la bala (x_b, y_b) es troba dins un rectangle (x_1, y_1, x_2, y_2) , però pot no ser gaire precís si la freqüència de càlcul de les posicions és baixa; la segona és obtenir la intersecció (x_i, y_i) entre dos segments, el darrer tros de la trajectòria $(x_{11}, y_{11}, x_{12}, y_{12})$ i el d'un dels costats del rectangle $(x_{21}, y_{21}, x_{22}, y_{22})$ (s'hauria de comprovar per cada costat del rectangle).

Avaluació

S'ha de lliurar un fitxer de codi XLISP-PLUS “canons.lsp” amb les funcions necessàries per resoldre el problema descrit. El codi ha d'incloure el nom dels autors i ha d'estar ben formatat, amb un bon disseny funcional, amb els comentaris necessaris per documentar cada funció i un ús clar dels identificadors. El punt d'inici serà la funció `(inicia)`.

No s'acceptaran lliuraments que facin servir funcions que no haguem vist a classe com, per exemple, funcions d'iteració (amb *do*, *loop*, *dolist*, *dotimes*, etc), d'assignació destructiva (amb *set*, *setf*, *setq*, etc; l'assignació no destructiva està permesa), entre d'altres. Es permet accedir i modificar les llistes de propietats dels símbols escenari, cano1, cano2 i els que es considerin necessaris.

El joc és necessari que compleixi amb la part obligatòria. Si no, s'avaluarà amb una nota inferior a 5. La nota de qualsevol d'aquests apartats pot no arribar a la màxima si no es compleixen els requisits de format, disseny funcional, documentació o claredat.

A continuació es descriuen la part obligatòria i altres parts que es poden afegir opcionalment al programa. No és necessari implementar tots els altres apartats. La nota màxima és un 10.

- ◆ **Part obligatòria.** Nota màxima: 7.0 punts.
 - ◇ Generació de l'escenari segons les indicacions, excepte el vent.
 - ◇ Orientació dels canons, regulació de potència, dispar de bales amb la trajectòria sense vent.
 - ◇ El joc mostra la trajectòria de la bala disparada.
 - ◇ El joc acaba en pitjar ESC o en destruir un canó.
- ◆ **Disseny gràfic.** Nota màxima: 0.5 punts.
 - ◇ Mentre no simplifiqueu els requisits, podeu millorar la presentació dels gràfics (per exemple: color de les línies, omplir els polígons, etc).
- ◆ **Interfície.** Nota màxima: 0.5 punts.
 - ◇ Mentre no simplifiqueu els requisits, podeu millorar la interfície del joc (per exemple: un menú, barres d'ajuda, missatges d'informació, etc).
- ◆ **Vent.** Nota màxima: 0.5 punts.

- ◇ Generació de la bandera segons la força del vent.
- ◇ Aplicació de la força del vent a la trajectòria de les bales.
- ◆ **Moviment dels canons.** Nota màxima: 0.5 punts.
 - ◇ Moviment horitzontal dels canons, sempre dins els límits dels seus camps (teniu en compte que fan 20 píxels d'ample), que ha d'influir l'origen dels seus dispars i si les bales els toquen.
- ◆ **Animació de la trajectòria.** Nota màxima: 0.5 punts.
 - ◇ En calcular la trajectòria, el programa mostra de manera animada la trajectòria que segueix el projectil.
 - Potser vos farà falta una funció *sleep* (Figura 1).
- ◆ **Resistència dels canons.** Nota màxima: 0.5 punts.
 - ◇ Els canons tenen una quantitat de punts de resistència (per exemple, 3). En ser tocats per aquesta quantitat de bales, son destruïts i acaba la partida.
- ◆ **Torns.** Nota màxima: 1.0 punts.
 - ◇ El joc funciona per torns alternatius per a cada canó, començant aleatòriament per un o l'altre, fins que un dels canons és destruït.
- ◆ **IA.** Nota màxima: 1.0 punts.
 - ◇ Addicionalment a poder jugar dos humans per torns, el programa pot iniciar-se per jugar contra la màquina (per exemple amb la funció (*inicia-ia*)), que triarà moure, orientar, regular la potència i disparar el seu canó. La IA ha de poder calcular un moviment raonablement bo (idealment el millor) però podeu fer que no sempre el triï perquè no guanyi sempre.
- ◆ **Altres.**
 - ◇ Mentre no simplifiquen els requisits, podeu fer altres canvis. Si creieu que poden molestar, posau-los darrere una opció o feis servir una funció que els faci servir i una altra que no.

La data límit per fer el lliurament, que es pot fer en parelles, és el dia de l'examen de cada convocatòria. Aquest lliurament es farà a través d'Aula Digital, en un node habilitat especialment.

```
(defun sleep (seconds)
  "Espera la quantitat indicada de segons"
  ; Això és un bucle iteratiu. NO PODEU FER-NE SERVIR ENLLOC MÉS
  (do ((endtime (+ (get-internal-real-time)
                    (* seconds internal-time-units-per-second))))
    ((> (get-internal-real-time) endtime))))
```

Figura 1. Implementació de la funció *sleep*.