

PacMan

Omer Giménez Mario G. Munzón Jordi Petit Salvador Roura

27 de gener de 2026

1 Regles del joc

Aquesta és una versió multi-jugador del PacMan, el clàssic joc d'arcade.

Resumint, cada jugador controla un pacman i diversos fantasmes que es mouen en un tauler rectangular que conté un laberint. A cada casella del tauler poden haver-hi cocos (dots) que aporten punts, cocos energètics (pills) que aporten punts i donen poder als pacmans, i bonus que aporten encara més punts i efectes especials. En mode normal, els pacmans no poden tocar els fantasmes perquè sinó moren. Però quan un pacman és poderós, aleshores corre al doble de velocitat i pot menjar els fantasmes enemics, els quals moren. De fet, els pacmans i els fantasmes no morren permanentment, sinó que després d'un cert temps, es regeneren. Quan el joc acaba, el jugador amb més punts guanya la partida.

A continuació es donen les regles del joc amb més detall:

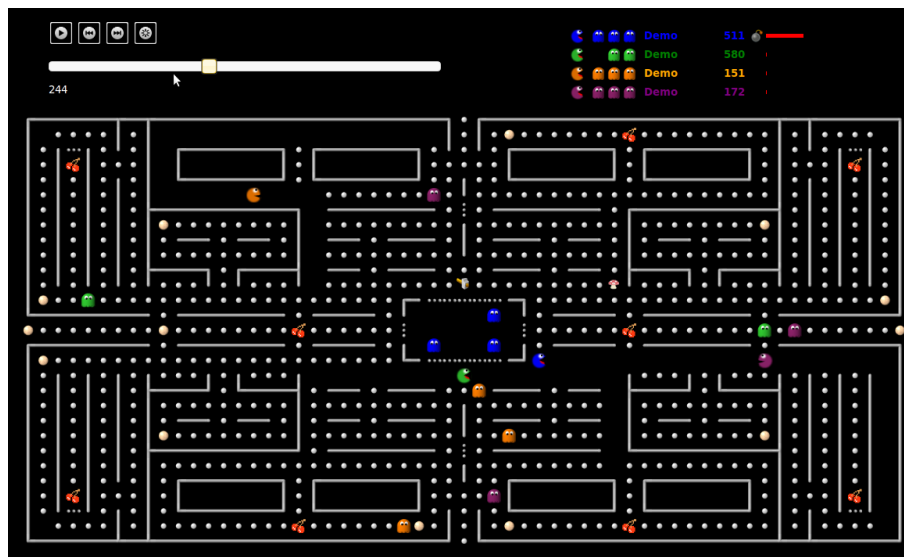
- Els següents paràmetres defineixen una partida:
 - nb_players: Nombre de jugadors de la partida.
 - nb_ghosts: Nombre de fantasmes per jugador.
 - nb_robots: Nombre de fantasmes i pacmans inicials en total, és a dir, $\text{nb_robots} = (1 + \text{nb_ghosts}) * \text{nb_players}$.
 - nb_rows: Nombre de files del tauler.
 - nb_cols: Nombre de columnes del tauler.
 - nb_rounds: Nombre de rondes de la partida.
 - regen_time: Temps de regeneració per als fantasmes i els pacmans.
 - power_time: Temps de poder dels pacmans.
- Cada casella del laberint pot estar buida o contenir:
 - Wall: Un mur.
 - Gate: Una porta.
 - Dot: Un coco.
 - Pill: Un coco energètic.
 - Bonus: Un coco de bonus.
 - Mushroom: Un bolet.

- Hammer: Un martell.
- A més, cada casella pot rebre la visita d'un (i només un) robot. Els robots possibles són:
 - PacMan: Un pacman normal.
 - PowerPacMan: Un pacman poderós.
 - Ghost: Un fantasma.

Cada robot pertany a un jugador i pot estar viu o mort temporalment.

- La partida comença amb la definició dels seus paràmetres, la descripció del laberint, les posicions inicials dels robots de cada jugador, i els marcadors a zero.
- Els pacmans es poden moure en horitzontal i en vertical a caselles adjacents que no siguin murs ni portes.
- Els fantasmes es poden moure en horitzontal i en vertical a caselles adjacents que no siguin murs, però sí que poden moure's per sobre de les portes.
- En una casella no hi poden haver dos robots alhora. Si en una casella es troben un fantasma i un pacman normal de diferents equips, el darrer morirà. Si en una casella es troben un fantasma i un pacman poderós de diferents equips, el primer morirà.
- Els pacmans normals i els fantasmes es mouen lentament, els pacmans poderosos ràpidament. Per fer-ho, els pacmans poderosos es poden moure a cada ronda, mentre que els pacmans normals i els fantasmes només es mouen cada dues rondes.
- A cada ronda, cada jugador sol·licita una acció, és a dir, tria —independentment dels altres jugadors— els moviments que vol aplicar als seus robots vius. Els moviments il·legals o repetits seran sempre ignorats. Per aplicar els moviments, a cada ronda, es decideixen els torns dels jugadors a l'atzar i s'apliquen els seus moviments en aquest ordre, tot respectant l'ordre relatiu dels moviments de cada jugador. Degut a les dependències entre els moviments dels robots, pot ser que no tots els moviments sol·licitats siguin concedits.
- És durant l'execució d'un moviment que es mouen els robots, morint-se si cal, i menjant cocos si és el cas. El marcador s'actualitza de forma convenient, tenint en compte els paràmetres següents:
 - pn_dot: Nombre de punts per menjar un coco.
 - pn_pill: Nombre de punts per menjar un coco energètic.
 - pn_bonus: Nombre de punts per menjar un coco de bonus, un bolet o un martell.

- pn_ghost: Nombre de punts per menjar un fantasma.
 - pn_pacman: Nombre de punts per menjar un pacman.
- Quan un pacman menja un bolet, el laberint es repobla de cocos.
 - Quan un pacman menja un martell, tots els fantasmes contraris moren. Però, compte!, la seva regeneració és molt ràpida, i els martells són prop de la gàbia.
 - Al final de cada ronda, s'actualitzen els comptadors de regeneració i de poder, permetent que els pacmans i els fantasmes morts revisquin i que els pacmans abandonin l'estat poderós. Els fantasmes sempre reviu en la seva gàbia, envoltada per murs i/o portes. Els pacmans sempre reviu en posicions buides i sense robots al voltant.
 - Com s'ha dit, algunes caselles són murs, pels quals no es pot passar. En particular, quasi tot el perímetre del tauler està envoltat de murs, excepte certes posicions simètriques que permeten passar per túnels hiperdimensionals (per exemple, un tunel pot permetre sortir per la banda inferior del tauler per anar a parar a la banda superior).
 - Si un jugador comet un error d'execució o esgota el seu temps de càlcul, el seu programa aborta, no juga més.
 - Guanya el jugador amb puntuació final més alta. El guanyador pot tenir tots els robots morts o, fins i tot, el programa abortat, això no es cap inconvenient.



2 Programació

En el web <https://pacman.jutge.org> trobareu el material necessari per programar el joc en C++. En concret, hi teniu tots els fitxers de suport, exemples de jugadors, i un visor de partides amb els quals podreu desenvolupar i provar els vostres jugadors.

Necessitareu **g++**, **make** i un navegador recent. El codi del joc és portable a qualsevol sistema Linux, Mac o Windows, suposant que hi instal·leu les eines adequades.

Els ordinadors de la FIB i de la FME tenen aquest software instal·lat.

Amb Debian o Ubuntu, amb una instrucció del tipus

```
sudo apt-get install build-essential chromium-browser
```

tindreu tot el que us cal.

Amb Mac, tindreu tot el que us cal si instal·leu XCode des de l'App Store.

Amb Windows, hauria de ser fàcil. Si algú fa el favor de dir-m'ho, ho publicaré.

2.1 Com fer un jugador

Per fer un jugador, copieu primer el fitxer **PlayerNull.cc** o **PlayerDemo.cc** a un fitxer **XXX.cc**, on **XXX** és un identificador de la vostra elecció. Trieu un identificador no ofensiu, que no hagi estat triat ja per un altre estudiant, i compost per, com a molt, 12 lletres, dígit i caràcters de subratllat; per exemple, **PacLady**.

A continuació, al fitxer **PacLady.cc** (o com l'hagueu anomenat) que acabeu de crear, heu de canviar la línia 15 per posar-hi el nom del vostre jugador (a l'exemple, `#define PLAYER_NAME PacLady`).

Finalment, heu d'implementar el vostre jugador tot completant la classe **PLAYER_NAME** que hereda les operacions de consulta del tauler (classe **Board**) i de creació d'accions (classe **Action**) a través de la classe base **Player**. El mètode **play()** es crida a cada ronda per transmetre-us l'estat actual del tauler i recollir les accions del vostre jugador. A la vostra classe, podeu afegir camps (variables) per recordar l'estat d'una ronda a l'altra, podeu afegir mètodes (o funcions) per descompondre el vostre programa, etc.

Fixeu-vos que la vostra classe ha de contenir una funció anomenada **factory()** que no heu de modificar, i que després de la classe també hi ha una crida per registrar el vostre jugador que tampoc heu de modificar.

2.2 Com executar i veure partides localment

1. Editant el **Makefile** podeu incorporar el fitxer **AIDummy-SISTEMA.o** (segons el sistema que tingueu) a la compilació per a poder usar el **Dummy** a les vostres partides. Podeu fer el mateix amb els fitxers **.o** d'altres jugadors.
2. Executeu **make all** per compilar tots els fitxers que calguin i crear l'executable **PacMan**.

3. Disputeu la partida amb una comanda com ara `./Game Demo PacLady -i demo.cnf -o partida.pac`. Aquí, el primer jugador serà `Demo`, el segon `PacLady` i la partida començarà amb la descripció `demo.cnf` i el resultat quedarà a `partida.pac`.
4. Visualitzeu la partida executant `./viewer.sh partida.pac`, que buscarà un navegador i l'obrirà.

Podeu obtenir la llista de paràmetres del programa `PacMan` amb `./Game --help`. En particular `./Game --list` us llistarà els noms dels jugadors inclosos.

Si us cal, podeu netejar el vostre directori de fitxers executables i fitxers objectes amb la comanda `make clean`.

2.3 Restriccions

Els codis dels jugadors que vulgueu enviar al web del joc han de complir certes limitacions:

- Tot el codi del vostre jugador s'ha de trobar en un sol fitxer i seguir les convencions donades.
- No podeu fer servir variables globals (utilitzeu camps).
- Només podeu usar les llibreries estàndard de C++, com ara `vector`, `map`, etc.
- No podeu obrir fitxers, ni fer altres crides al sistema operatiu (threads, forks, etcètera).
- Si us cal, podeu utilitzar `cout` i `cerr`, però usar `cin` no té sentit. Però compte, escriure missatges consumeix temps.
- En execució local, el sistema no controla jugadors que abortin, que triguin massa, o que interfereixin amb els contraris. En l'execució al servidor, sí.

2.4 Estructures de dades

Per saber com consultar el tauler, doneu un cop d'ull a les operacions públiques de la class `Board` del fitxer `Board.hh`. Per saber com donar les accions, doneu un cop d'ull a les operacions públiques de la class `Action` del fitxer `Action.hh`. Fixeu-vos que només podeu utilitzar les operacions públiques d'aquestes classes.

3 Consells

Us recomanem seguir els consells següents:

- Estudieu la part pública de les classes lliurades. No us preocupeu per les parts privades ni la implementació.

- Sembla que el visor funciona millor amb Chrome que amb Firefox. Però, compte, per raons de seguretat, per veure partides locals, cal obrir Chrome amb la opció `--allow-file-access-from-files`, que permet llegir fitxers del vostre disc.
- Comenceu amb estratègies molt senzilles, que siguin fàcils de programar i de depurar, que és exactament allò que necessiteu al principi.
- Programeu procediments senzills i útils, i assegureu-vos que funcionin correctament.
- No us arrisqueu a ser eliminats abans de començar. Aconseguiu que el web us accepti un jugador tan ràpidament com us sigui possible. Un jugador acceptat representa tenir assegurada part de la nota de la pràctica!
- Conserveu diverses versions antigues (però que funcionin correctament) dels vostres jugadors, i no les toqueu per res.
- Compileu sovint, testejeu sovint. És molt més fàcil trobar i corregir els errors quan s'han canviat unes poques línies de codi que quan s'acumulen molts canvis de cop.
- Feu servir `cerrs` per posar xivatos, i `asserts` per comprovar que el codi fa el que toca. Però comenteu els `cerrs` abans d'enviar el jugador al web, ja que alenteixen els programes.
- Per testejar un jugador nou, enfronteu-lo contra uns altres jugadors que no treguin cap missatge. Així només apareixeran per pantalla els missatges del nou jugador.
- Un error de segmentació en el vostre codi aturarà el simulador, i no podreu veure què ha provocat l'error del jugador. Apreneu a usar el `valgrind` si no sou capaços de corregir els errors amb `cerrs`.
- Poseu les opcions de debug al Makefile per ajudar-vos a trobar possibles errors.
- No us arrisqueu a quedar eliminats per culpa d'un jugador massa lent. Doneu-vos un bon marge de seguretat. En particular, eviteu tant com pugueu l'escriptura de missatges.
- Feu competir els vostres jugadors contra rivals diferents, i estudeu les partides. Encara que no podreu veure els codis dels altres estudiants, si sou capaços de deduir les seves tàctiques i us semblen útils, podeu mirar d'imitar-les, defensar-vos-en o millorar-les.
- No deixeu el vostre codi a ningú, ni tan sols d'una versió antiga. Un cop feta la competició s'analitzaran els programes de cada ronda amb programes detectors de plagi (JPlag, per exemple). També es comparan

els jugadors d'aquest curs amb els jugadors de cursos anteriors. Passar-vos els fitxers `.so` té menys risc, però es poden fer servir per endevinar la vostra estratègia.

- Per evitar disgustos, feu servir el web del joc per jugar partides amb els vostres amics. Però no confieu massa en els vostres amics...
- Tingueu en compte que podeu lliurar nous jugadors en qualsevol moment (excepte durant la fase final).
- No espereu fins al darrer moment per enviar les vostres solucions. Els demés també ho fan i la cua del Jutge no dona l'abast.
- Insistim: Feu codis senzills. Compileu sovint, testejeu sovint. O afronteu-ne les conseqüències.