

# Tarea Chica 1

Línea de comandos y Git  
Profesores Vicente Domínguez y Luis Ramírez  
Enunciado: 12 de Agosto de 2019

---

## Indicaciones

- Fecha de Entrega: Lunes 19 de Agosto de 2019 hasta las 23:59
  - Se debe entregar la tarea en el repositorio asignado a cada uno por Github Classroom.
  - Cada hora de atraso descuenta 5 décimas de la nota que se obtenga.
  - La tarea es *individual*. La copia será sancionada con una nota 1,1 en el la tarea, además de las sanciones disciplinarias correspondientes.
- 

## Objetivo

- Adquirir conocimiento sobre el uso de comandos básicos del **shell (intérprete de órdenes)** llamado **Bash (Bourne Again Shell)**, el que se usa en sistemas operativos **UNIX** para ejecutar instrucciones ingresadas en una **interfaz de línea de comandos**.
- Utilizando el conocimiento anterior, crear los comandos necesarios para poder jugar DCCpoly.
- Hacer uso de **Git** y **GitHub** para registrar su progreso.

## Utilidad

Usar una interfaz de línea de comandos con Bash como intérprete de los comandos (e.g. **Git Bash**) permite crear, mover y eliminar archivos y carpetas. Además, permite modificar nombres de archivos y sus contenidos, realizando acciones como ordenar

los contenidos del archivo o modificarlos y, la verdad, permite hacer virtualmente lo que sea, pues Bash es un intérprete del **lenguaje de programación Bash**.

## Parte I (0.5 puntos)

Primero, se debe realizar este **tutorial en codecademy de forma obligatoria**. Su usuario en esta plataforma deberá ser igual al nombre de su repositorio de GitHub (en caso de estar ocupado pueden utilizar su usuario UC o avisarnos oportunamente, lo mismo si ya tienen una cuenta).

Deberán poner en el readme el link a su perfil de codecademy para certificar que completaron el tutorial.

## Parte II (4 puntos)

Dado que, luego de hacer la Parte I de su tarea han quedado motivadísimos con seguir poniendo en práctica sus conocimientos sobre comandos en Bash, ha llegado la hora de ponerlos a prueba con el novedoso juego del departamento: **DCCpoly**.

### DCCpoly

¿Has soñado siempre con salir de tu carrera? ¿Estás aún atrapado con ramos que no logras pasar? ¿Quieres conocer un poco más sobre el maravilloso mundo del departamento de computación? Te presentamos DCCpoly, el juego que puede hacer tus sueños realidad.

DCCpoly es un juego de mesa basado en la aprobación de tus ramos y tu posterior labor como ayudante, completamente inspirado en los cursos y personajes del DCC. El objetivo del juego es aprobar todos los ramos con los créditos que tienes disponible (Aquel jugador que aprueba todos los cursos del tablero primero, gana el juego).

Los jugadores mueven sus respectivas fichas por turnos en sentido horario alrededor de un tablero, basándose en la puntuación obtenida en un dado y caen en cursos que pueden tomar y aprobar <sup>1</sup> si cuentan con los créditos suficientes y si por supuesto deciden invertirlos en este. Posteriormente pueden ser ayudantes de un curso si caen nuevamente en la casilla correspondiente a este y si *pagan* el precio por ser ayudantes (que sale en la carta del curso). Los ayudantes *cobran créditos*<sup>2</sup> a los jugadores que

---

<sup>1</sup>Como simplificación, si toman un curso, este se aprueba inmediatamente

<sup>2</sup>Esta también es otra simplificación del juego, los ayudantes no les quitamos sus créditos

caigan en los cursos de su dominio .

## Piezas del juego

Las *piezas* del juego podrán encontrarlas en la carpeta DCCpoly de su repositorio.

1. **Tablero** de 8x8, compuesto por el punto de partida BatiChrist, cursos del DCC, eventos del DCC y casillas de destino (tablero.txt)
2. **Cartas de cursos** que describen el nombre y sigla del curso, créditos que cuesta tomarlos/aprobarlos (Créditos), cuántos créditos cuesta ser ayudante (Precio Ayudantía) y cuántos créditos paga (Pago ayudantía). Estas cartas son las que el DCC <sup>3</sup> te entrega cuando tomas/apruebas un curso a cambio de créditos (Archivos txt en la carpeta Cursos)
3. **Cartas de Eventos** que describen los eventos más conocidos del DCC y que para efectos del juego tienen alguna consecuencia en el avance de tu carrera. <sup>4</sup> (Archivos txt en la carpeta Eventos)
4. **Cartas de Destino** que representan situaciones que podrían pasarte en tu vida universitaria dentro del DCC (Archivos txt en la carpeta Destino)
5. **Jugadores**, corresponden a la representación del jugador y los cursos que posee (Cada Jugador tiene una carpeta dentro de la carpeta Jugadores donde se tiene el archivo txt con su posición actual, los créditos totales que posee y las siglas de los cursos que ha aprobado, además de la carpeta Ayudantías donde están los archivos correspondientes a las cartas de cursos en que es ayudante)
6. **Dado**. DCCpoly se juega con un sólo dado

## Detalles del juego y consideraciones

DCCpoly pueden jugarlo máximo 4 personas. <sup>5</sup>

Cada jugador inicia la partida con 50 créditos en el punto de partida del tablero (BatiChrist)

---

<sup>3</sup>Más conocido como Banco, en otros juegos similares

<sup>4</sup>Te invitamos a informarte sobre estos eventos y animarte a que participes en ellos :)

<sup>5</sup>Obviamente el mínimo son 2, pero recuerda que en la realidad, la competencia no es contra el otro, sino que contra tí mismo

Si en algún momento tienes que pagar algo y no dispones de los créditos para hacerlo, pierdes el juego.

Cada vez que un jugador pasa por el BatiChrist se le conceden 10 créditos.

Las posiciones en el tablero son un número entre 0 y 27.

Cada jugador lanza el dado según su turno y avanza en el tablero la cantidad de posiciones indicada por este.

**Cursos:** si caes en un casillero de tipo curso puede pasar que:

- Si no has tomado/aprobado el curso, puedes pagar los créditos correspondientes a la aprobación del curso para aprobarlo y añadirlo a tu lista de cursos aprobados. Si no pagas estos créditos, no pasa nada.
- Si caíste en un casillero en el que ya has aprobado el curso (en un turno anterior) puedes pagar los créditos para tomar la ayudantía del curso, lo que permitirá que ganes créditos si alguien más cae en ella. Si alguien ya tomó la ayudantía del curso no puedes adquirirla.
- Si alguien es ayudante de este curso, entonces debes pagarle los créditos correspondientes al ayudante (Lo que sale en *Pago Ayudantía* en la tarjeta), esto es independiente de si tomas/apruebas el curso o no.

**Eventos:** si caes en un casillero de eventos, revisa la tarjeta de evento correspondiente y sigue las instrucciones. Los posibles eventos son:

- DCCercano
- Fonda de Don Yadrán
- Programatón
- Pasantía en Google
- Tarreo DCC
- DCCalabaza
- DCCerro

**Destino:** si caes en un casillero de destino, revisa la tarjeta de evento correspondiente y sigue las instrucciones

## Tu misión

Debes mostrar el/los comandos que puedan realizar lo pedido en las siguientes instrucciones:

1. **[0.1 puntos]** Lanzar el dado: Debes escribir un comando en bash que te permita generar un número aleatorio entre 1 y 6. <sup>6</sup>
2. **[0.1 puntos]** Cambiar los créditos actuales del jugador  $i$  a la cantidad de créditos  $c$ .
3. **[0.1 puntos]** Mover al jugador  $i$  a una posición  $j$  del tablero.
4. **[0.2 puntos]** Otorgarle una ayudantía al jugador  $i$ . Para esto tienes que mover la tarjeta del curso  $k$  que se encuentra en la carpeta *cursos* de la carpeta *DCC* a la carpeta *ayudantías* del jugador  $i$ .
5. **[0.2 puntos]** Aprobarle un curso al jugador  $i$ . Para esto debes agregar al archivo *info.txt* del jugador  $i$  el curso  $k$  que desees aprobarle.
6. **[0.2 puntos]** Imagina que eres el jugador  $i_1$  y que luego de lanzar el dado caes en un casillero del curso  $k$ , ubicado en la posición  $j$ , del cual el jugador  $i_2$  es ayudante, tu no has aprobado este curso aún. Suponga que Créditos de ese curso es  $c$ , que Precio Ayudantía es  $p$ , que Pago Ayudantía es  $a$ , que los créditos actuales del jugador 1 son  $cr_1$ , los créditos actuales del jugador 3 son  $cr_3$  y que desees aprobar el curso. Realiza los pasos de acuerdo a las reglas del juego.
7. **[0.2 puntos]** Imagina que eres el jugador  $i_1$  y luego de lanzar el dado caes en un casillero del curso  $k$ , ubicado en la posición  $j$ , del cual el jugador  $i_2$  es ayudante, esta vez tu ya habías aprobado el curso. Suponga que Créditos de ese curso es  $c$ , que Precio Ayudantía es  $p$ , que Pago Ayudantía es  $a$ , que los créditos actuales del jugador 1 son  $cr_1$ , los créditos actuales del jugador 3 son  $cr_3$ . Realiza los pasos de acuerdo a las reglas del juego.
8. **[0.4 puntos]** El jugador  $i$  cae en el casillero DCCercano. Muestra el comando con el cual puedes visualizar lo que dice la tarjeta (e01.txt en la carpeta *eventos* de *DCC*). Además muestra los comandos de las instrucciones correspondientes a la tarjeta. Suponga que los créditos actuales del jugador  $i$  son  $cr_i$ .
9. **[0.4 puntos]** El jugador  $i$  cae en el casillero Programatón. Muestra el comando con el cual puedes visualizar lo que dice la tarjeta (e03.txt en la carpeta *eventos* de *DCC*). Además muestra los comandos de las instrucciones correspondientes

---

<sup>6</sup>Este comando puedes encontrarlo si vas a la ayudantía correspondiente o puedes simplemente googlearlo

a la tarjeta. Suponga que se quiere mover al curso  $k$  que aún no has aprobado, ubicado en la posición  $j$  y que tienes créditos suficientes para tomarlo, por lo que decides hacerlo. Créditos de ese curso es  $c$  y los créditos actuales del jugador  $i$  son  $cr_i$

10. **[0.4 puntos]** El jugador  $i$  cae en el casillero Tarreo DCC. Muestra el comando con el cual puedes visualizar lo que dice la tarjeta (e05.txt en la carpeta *eventos* de DCC). Además muestra los comandos de las instrucciones correspondientes a la tarjeta. Suponga que al lanzar el dado sale un número menor a 2 y que devuelves la ayudantía del curso  $k$  llamado *curso\_k.txt*.
11. **[0.4 puntos]** El jugador  $i$  cae en uno de los casilleros de destino. Muestra el comando con el cual puedes visualizar lo que dice la tarjeta (d01.txt en la carpeta *destino* de DCC). Además muestra los comandos de las instrucciones correspondientes a la tarjeta. Suponga que deseas remover el curso aprobado  $k$ , llamado *curso\_k.txt* del cual además eres ayudante, .
12. **[0.4 puntos]** El jugador  $i$  cae en uno de los casilleros de destino. Muestra el comando con el cual puedes visualizar lo que dice la tarjeta (d03.txt en la carpeta *destino* de DCC). Además muestra los comandos de las instrucciones correspondientes a la tarjeta.
13. **[0.4 puntos]** Para cada jugador, lista todos los cursos en los que es ayudante.
14. **[0.3 puntos]** Para cada jugador ve su estado actual en el juego (su posición créditos actuales y cursos aprobados).
15. **[0.2 puntos]** Como eres curioso y han quedado algunas tarjetas sin leer, muestra el comando con el cual puedes visualizar lo que dicen las siguientes tarjetas:
  - e02.txt en la carpeta *eventos* de DCC
  - e04.txt en la carpeta *eventos* de DCC
  - e06.txt en la carpeta *eventos* de DCC
  - e07.txt en la carpeta *eventos* de DCC
  - d02.txt en la carpeta *destino* de DCC
  - d04.txt en la carpeta *destino* de DCC
  - d05.txt en la carpeta *destino* de DCC

Recuerda que la posición, los créditos (actuales) y los aprobados (créditos aprobados) de cada jugador se encuentra en el archivo *info.txt* dentro de la carpeta Ji, dentro de la carpeta jugadores.

Este archivo se ve de la siguiente manera para un jugador en su etapa inicial:

```
Posición: 0
Créditos: 50
Aprobados:
```

Figura 1: Ejemplo jugador inicial.

Para un jugador que ha aprobado IIC1105 y IIC1103:

```
Posición: 0
Créditos: 50
Aprobados:
IIC1005
IIC1103
```

Figura 2: Ejemplo jugador con créditos aprobados.

Las modificaciones que hagas a los archivos **deben** ser a través de comandos por consola.

Para esta parte, deben subir a GitHub un **informe en formato PDF** donde, para cada instrucción enumerada anteriormente, se muestren los comandos en Bash que puedan ejecutarla, además de una explicación de por qué ese comando puede hacer lo pedido.

Para los comandos, ten presente que estos deben poder ser ejecutados **desde la carpeta DCCpoly**. Puedes ver la estructura de directorios al final de este documento.

Con respecto al informe, puedes escribir los comandos en términos de las variables dadas o reemplazarlas por algún valor que estimes conveniente, considerando las restricciones correspondientes <sup>7</sup>. Cualquier decisión que tomes debe quedar clara en tu explicación del comando. Recuerda que si quieres probar que tus comandos funcionen como esperas, tendrás que reemplazar las variables por los valores necesarios.

---

<sup>7</sup>Por ejemplo que  $i$ , siendo el número de jugador puede ser un número entre 1 y 4, que las posiciones van entre 0 y 27, etc...

## Uso De GitHub (1.5 puntos)

A lo largo de toda la elaboración de su tarea, deberán usar Git y GitHub para reportar su progreso. Al término, deben tener **al menos** 2 commits. Uno que refleje que terminaron el tutorial de codecademy y otro que refleje que han terminado los comandos del DCCpoly.

Nótese que **está permitido** tener más commits que los indicados en el párrafo anterior, pero estos últimos (los 2 commits) son imprescindibles y otorgan los últimos 1.5 puntos. **Todos** los commits que realicen **deben** tener un mensaje descriptivo (e.g “Tutorial terminado”).

Recuerden siempre seguir los **tres** pasos necesarios para subir sus archivos a GitHub (**Git add, Git commit y Git push**), además de verificar que sus tareas se hayan subido correctamente a su repositorio.

## Entregables

En resumen, deben subir los siguientes archivos:

- Link de su perfil en codecademy en el readme del repositorio para certificar que completaron el tutorial
- Informe en formato pdf con el comando y explicación, para cada instrucción de la parte II

## Recomendaciones

- Antes de hacer preguntas en las issues, buscar en Google todo lo que se pueda relacionado a cualquier duda que se tenga. Investiguen los enlaces en magenta de este enunciado y, si es posible, ir más allá.
- Para la parte II es muy recomendable que investiguen sobre el comando *sed* que les será muy útil en muchas de las instrucciones. Por ejemplo, para editar una línea específica de un archivo o para eliminar una línea que contenga un cierto string.

## Penalizaciones

- Por no completar tutorial de Codecademy: **nota máxima 4**.
- Por tener al menos un commit con un mensaje poco descriptivo: **-0.5 puntos**.



## Estructura de directorios

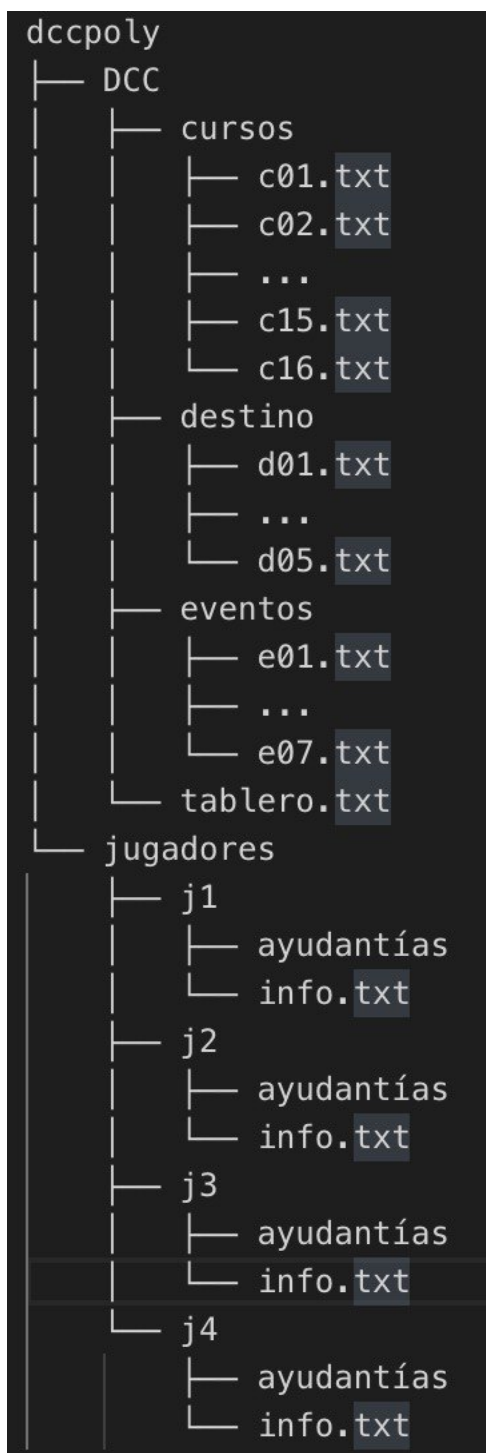


Figura 3: Estructura de directorios.