

Tarea N°1 - ELO 329 2018-1:
Descuentos, Recomendaciones y Aspectos
Destacados

Pilar Arancibia - Jesús Márquez

Índice

I	Recomendaciones Generales	2
	I-A Respecto al Código de la Tarea	2
	I-B Respecto a la Entrega y Documentación de la Tarea	2
II	Grupo 1 : 201630003-1 201430028-K 201530008-9 201330057-K	2
	II-A Descuentos	2
	II-B Recomendaciones	2
	II-C Aspectos Destacados	2
III	Grupo 2 : 201630019-8 201630042-2 201630016-3	2
	III-A Descuentos	2
	III-B Recomendaciones	3
	III-C Aspectos Destacados	3
IV	Grupo 3 : 201321051-1 201321009-0 201304179-5	3
	IV-A Descuentos	3
	IV-B Recomendaciones	4
	IV-C Aspectos Destacados	4
V	Grupo 4 : 201530022-4 201530025-9 201530014-3	4
	V-A Descuentos	4
	V-B Recomendaciones	4
VI	Grupo 5 : 201630006-6 201630039-2 201630002-3	5
	VI-A Descuentos	5
	VI-B Recomendaciones	5
	VI-C Aspectos Destacados	5
VII	Grupo 6 : 201504100-8 201504126-1 201504012-5	5
	VII-A Descuentos	5
	VII-B Recomendaciones	5
	VII-C Aspectos Destacados	6
VIII	Grupo 7 : 201303041-6 201303044-0 201104505-K	6
	VIII-A Descuentos	6
	VIII-B Recomendaciones	6
IX	Grupo 8 : 201573070-9 201890059-1 201790233-7	6
	IX-A Descuentos	6
	IX-B Recomendaciones	6
	IX-C Aspectos Destacados	7
X	Grupo 9 : 201421030-2 201321019-8 201404136-5	7
	X-A Descuentos	7
	X-B Recomendaciones	7
	X-C Aspectos Destacados	7
XI	Grupo 10 : 201404021-0	7
	XI-A Descuentos	7
	XI-B Recomendaciones	8
	XI-C Aspectos Destacados	8
XII	Grupo 11 : 201103032-K 201204122-8 201604009-9	8
	XII-A Descuentos	8
	XII-B Recomendaciones	8
	XII-C Aspectos Destacados	8
XIII	Grupo 12 : 201330015-4 201421003-5 201321074-0	8

	XIII-A	Descuentos	8
	XIII-B	Recomendaciones	8
	XIII-C	Aspectos Destacados	9
XIV	Grupo 13 : 201321034-1 201321007-4 201321056-2		9
	XIV-A	Descuentos	9
	XIV-B	Recomendaciones	9
	XIV-C	Aspectos Destacados	9
XV	Grupo 14 : 201321043-0 201730038-8 201704005-K		9
	XV-A	Descuentos	9
	XV-B	Recomendaciones	9
XVI	Grupo 15 : 201203014-5 201304292-9		10
	XVI-A	Descuentos	10
	XVI-B	Recomendaciones	10
XVII	Grupo 16 : 201351025-6 201123008-6 201890108-3		10
	XVII-A	Descuentos	10
	XVII-B	Recomendaciones	10
XVIII	Grupo 17 : 201604062-5 201630034-1		11
	XVIII-A	Descuentos	11
	XVIII-B	Recomendaciones	11
XIX	Grupo 18 : 201530013-5 201630005-8 201621058-K		11
	XIX-A	Descuentos	11
	XIX-B	Recomendaciones	11
	XIX-C	Aspectos Destacados	11
XX	Grupo 19 : 201630012-0 201630015-5 201630004-K		12
	XX-A	Descuentos	12
	XX-B	Recomendaciones	12
	XX-C	Aspectos Destacados	12
XXI	Grupo 20 : 201430040-9 201430052-2 201430035-2		12
	XXI-A	Descuentos	12
	XXI-B	Recomendaciones	12
	XXI-C	Aspectos Destacados	12
XXII	Grupo 21 : 201030013-7 201030003-K		12
	XXII-A	Descuentos	12
	XXII-B	Recomendaciones	13
	XXII-C	Aspectos Destacados	13
XXIII	Grupo 22 : 201121020-4 201121032-8		13
	XXIII-A	Descuentos	13
	XXIII-B	Recomendaciones	13
	XXIII-C	Aspectos Destacados	13
XXIV	Grupo 23 : 201273065-1 201790232-9 201430022-0		13
	XXIV-A	Descuentos	13
	XXIV-B	Recomendaciones	14
XXV	Grupo 24 : 201103007-9 201604147-8		14
	XXV-A	Descuentos	14
	XXV-B	Recomendaciones	14
	XXV-C	Aspectos Destacados	14

I. RECOMENDACIONES GENERALES

I-A. *Respecto al Código de la Tarea*

-

I-B. *Respecto a la Entrega y Documentación de la Tarea*

-

II. GRUPO 1 : 201630003-1 201430028-K 201530008-9 201330057-K

II-A. *Descuentos*

- No se explica la relación entre las clases en las etapas ni en que consisten los módulos principales. **(-9 PUNTOS a Explicación)**
- El makefile de la etapa 4 y 5 contiene datos basura que hacen que éste no se ejecute. **(-1 PUNTO a Makefile)**
- El makefile de la etapa 5 contiene clase incompleta que no permite compilar adecuadamente la etapa. **(-1 PUNTO a Makefile)**
- El manejo del tiempo de simulación es incorrecto y esto lleva a que no se represente adecuadamente la simulación. **(-2 PUNTOS a Etapa 1)**
- Formato de archivos .pbm de salida no se respeta para laberintos pequeños. **(-1 PUNTO a Etapa 2)**
- Código no es robusto para tercera etapa, no genera rutas adecuadamente al cambiar la posición inicial, por ejemplo al dejarlo al medio sin murallas alrededor se queda estático, no opera ante configuración planteada en el código de ayuda. **(-8 PUNTO a Etapa 3)**
- Código lanza excepciones y no siempre genera el archivo .pbm de salida. **(-2 PUNTO a Etapa 3)**
- Código indica haber llegado a la salida del laberinto cuando hay una excepción, no llegando siempre a la salida. **(-5 PUNTO a Etapa 4)**

II-B. *Recomendaciones*

- Deben agregar a los dos ayudantes en el repositorio GIT de su tarea, dado que la elección de quién revisa es aleatoria.
- Evitar utilizar muchas veces consecutivas los mismos mensajes en los commits.
- Rotular adecuadamente el gráfico, con título y nombre de los ejes, además de utilizar una escala más adecuada.
- Inicializar todos los atributos de una clase en los constructores.
- No utilizar variables no inicializadas en métodos, y realizar métodos que sean concordantes a la clase, por ejemplo, si se trabaja con vectores como unidad de medición de distancia, no mostrar por pantalla valores de tiempo.
- No basta con explicar generalmente los métodos en los comentarios del códigos, en secciones complicadas, tal como la rotación e impresión del laberinto es recomendable señalar brevemente la idea.
- Tener más cuidado con copia baja, dado que al copiar referencias, al general la ruta también se altera el maze original y ello conlleva al mal funcionamiento de la etapa 3.
- Indicar claramente el marco de referencia de cada etapa. Ej. El valor de x aumenta hacia la derecha.
- No definir la misma clase dos veces en archivos distintos.

II-C. *Aspectos Destacados*

- Readme claro y preciso.
- En general, buen uso de GIT, se demuestra desarrollo iterativo y colaboración de todos los miembros.
- Se felicita por utilizar rotulado tipo JavaDoc.

III. GRUPO 2 : 201630019-8 201630042-2 201630016-3

III-A. *Descuentos*

- Existencia de múltiples archivos de documentación **(-2 Puntos en Descuentos)**.
- No se encuentra el gráfico pedido para la etapa 1 **(-2 Puntos a Gráfico-Etapa 1 en Documentación)**.
- No se incorpora diagrama de alto nivel para última etapa desarrollada. **(-5 PUNTOS a Diagrama UML)**
- La idea de la documentación era que ustedes expliquen brevemente el funcionamiento de su programa y las interacciones entre sus módulos, no generar una documentación automática con métodos como doxygen. **(-5 Puntos a Explicación en Documentación)**.
- No se documenta la estrategia utilizada para la etapa 5 **(-4 Puntos a Explicación en Documentación)**.

- No se abordaron los problemas y soluciones durante el desarrollo de la tarea en la documentación **(-9 Puntos a Problemas y Soluciones en Documentación)**.
- Robot da vuelta en si mismo por implementación de método para girar sin que haya detectado paredes. Al comentar esa implementación, el robot atraviesa paredes y sale del laberinto, generando una excepción **ArrayOutOfBoundsException** que no es manejada. **(-8 Puntos a Ruta generada es lógica en Etapa 3)**.
- Etapa 4 genera un error de compilación, y al manejar el error genera una excepción **NoSuchFileElementException** **(-15 en Etapa 4) (-2 a Errores en Cero Warnings, Cero Errores)**.
- Etapa 5 genera mismo error de compilación que en la Etapa 4, y al arreglarlo aparecen un sin fin de errores respecto a la clase **Pilot()**. **(-10 Puntos a Etapa 5) (-2 Puntos a Errores en Cero Warnings, Cero Errores)**.

III-B. Recomendaciones

- Deben tener solo un archivo de documentación que contenga todas las etapas, a excepción de la etapa 5 la cuál se pide que se documente aparte.
- **No abusar de una documentación auto-generada.** Para la documentación se pedían diagramas UML, explicación breve de lo que hace el programa y cómo lo logra, el archivo excel junto con el gráfico de la etapa 1 y por lo menos 3 problemas que hayan tenido a lo largo del desarrollo y cómo los abordaron, algo parecido a un informe. Documentación auto generada debe ser complementaria a esto.
- Deben de ver y seguir cada uno de los pasos de la entrega de tareas que se encuentra en la plataforma informativa de tareas.
- Tener cuidado al momento de hacer un makefile con el nombre definido de un archivo de entrada. En la etapa 3 se define un archivo de nombre maze-in.pbm, pero en la carpeta del proyecto su pbm tenía el nombre de simple-maze.pbm, lo que hacía que no pudieran compilarse los archivos.
- Como se menciona en Descuentos, el robot quedaba girando infinitamente en sí mismo, pues cuando no se detectaban paredes ustedes lo hacían girar hacia la derecha. En ese momento el robot debiera de seguir un camino recto.
- Al comentar ese método el robot atravesaba las paredes debido a que no era capaz de percatarse si al doblar en una dirección concreta habría o no una pared también ahí.
- La posición del Robot en el código entregado estaba siempre fuera del laberinto de ejemplo, por lo que generaba una excepción de **ArrayOutOfBoundsException** inmediatamente. Tengan ojo en dejar listo para llegar y probar el funcionamiento de su código para sus siguientes entregas.
- Tengan ojo que el rango de los sensores no solo implica la posición de rango más lejana, sino todas las posiciones desde el sensor hacia el rango.
- Para la etapa 3 no era necesario un funcionamiento excepcional del robot, ni que fuera capaz de salir del laberinto, sino más bien el que sea capaz de moverse dentro de un laberinto sin problemas y de forma lógica.
- Cuando se trabaja con arreglos siempre es necesario manejar o restringir los casos en que se pueda desbordar al recorrerlo, para este caso, cuando el robot sale del laberinto genera una excepción debido a que se intenta acceder a índices indefinidos o inexistentes del arreglo.
- En la Etapa 4 DocumentedMaze debía de inicializar el atributo de su padre utilizando **super()**.
- En la Etapa 4 DocumentedMaze utiliza un método no creado llamado **read(Scanner s)**.
- Consideren para las próximas entregas que una etapa que no puede compilarse no debe entregarse. Documentar hasta donde hayan llegado de forma exitosa, no les baja puntaje respecto a errores de compilación y pueden optar a todo el puntaje de la documentación.
- Consideren para las próximas tareas una descripción de commits más específica. No "Actualización etapa X". Expliquen que actualización hicieron.
- Utilizar **Aragorn** para revisar su código antes de entregar, pues está dicho que ahí es donde los ayudantes compilan y ejecutan su solución.

III-C. Aspectos Destacados

- Excelente README.
- Un impecable orden y uso de comentarios en las primeras etapas, perfecto! **(+2 Puntos a Extra)**.
- Buena colaboración de todos los integrantes del equipo.

IV. GRUPO 3 : 201321051-1 201321009-0 201304179-5

IV-A. Descuentos

- No se entrega gráfico de la etapa 1. **(-2 PUNTOS a Gráfico-Etapa 1)**
- Código no es robusto para tercera etapa, no genera rutas adecuadamente al cambiar la posición inicial, no opera ante configuración planteada en el código de ayuda. **(-8 PUNTO a Etapa 3)**

- Código lanza excepciones y no siempre genera el archivo .pbm de salida. **(-2 PUNTOS a Etapa 3)**
- Código indica haber llegado a la salida del laberinto cuando se acerca a los límites de la imagen, no llegando siempre a la salida. **(-5 PUNTOS a Etapa 4)**
- En algunos casos el piloto toma curvas de manera que queda con la muralla a su izquierda. **(-2 PUNTOS a Etapa 4)**

IV-B. *Recomendaciones*

- Tratar de evitar al máximo mensajes de commits con información escasa o poco clara, como "Update Vector2D.java.º .Arreglado errores", tratar dar un poco más de contexto.
- No se pueden realizar modificaciones después del tag Final, para próxima entrega se considerará el estado del repositorio al momento de realizado el tag.
- No dejar atributos sin inicializar, ya que estos podrían tomar valores no deseados.
- Tener más cuidado con copia baja, dado que al copiar referencias, al general la ruta también se altera el maze original y ello conlleva al mal funcionamiento de la etapa 3.
- Indicar claramente el marco de referencia de cada etapa. Ej. El valor de x aumenta hacia la derecha.

IV-C. *Aspectos Destacados*

- En general, buen uso de GIT, se demuestra desarrollo iterativo y colaboración de todos los miembros.
- Buen readme, bastante detallado y explicativo.
- Muy buena documentación, breve pero concisa, detalla claramente las relaciones entre las clases utilizadas y las dificultades presentadas.
- Muy buena manera de abordar el tema de los giros con matrices de rotaciones.

V. GRUPO 4 : 201530022-4 201530025-9 201530014-3

V-A. *Descuentos*

- Existencia de múltiples archivos de documentación **(-2 Puntos en Descuentos)**.
- Existencia de múltiples archivos de README **(-2 Puntos en Descuentos)**.
- No se encuentra el gráfico pedido para la etapa 1 **(-2 Puntos a Gráfico-Etapa 1 en Documentación)**.
- Se abordan dos problemas y soluciones, el mínimo pedido por el profesor es tres **(-3 a Problemas y soluciones en Documentación)**.
- Formato presentado para la etapa 1 no es correcto. **(-3 Puntos a Formato de salida en Etapa 1)**.
- Salida de la etapa 1 no es robusto. **(-3 Puntos a Valores presentados son correctos en Etapa 1)**.
- No se realiza una copia profunda del Robot, pues este al desplazarse y marcar su recorrido también está creando nuevas paredes en el laberinto, haciendo que se encierre él mismo. **(-8 Puntos a Ruta generada es lógica en Etapa 3)**.
- Robot nunca sale del laberinto. **(-7 a Robot sale del laberinto en Etapa 4)**.
- Sin utilizar argumentos opcionales, el robot no sale del laberinto **(-1 a Sin argumentos opcionales en Etapa 4)**.
- Etapa 5 no terminada **(-10 a Etapa 5)**.

V-B. *Recomendaciones*

- Deben de hacer una sola documentación que contenga información acerca de todas las etapas y un README para cada tarea explicando la forma de compilar y ejecutar cada una de las etapas.
- Deben de ver y seguir cada uno de los pasos de la entrega de tareas que se encuentra en la plataforma informativa de tareas.
- Deben seguir lo pasos de la construcción de la tarea y resultados esperados dada por el profesor.
- Mantengan un orden en la salida, los print utilizados para el debugging no deben estar en una entrega final.
- Tanto al probar el código como al entregarlo es bueno tener una referencia clara de la salida que se requiere, por ejemplo, en la etapa 1 hubiera sido bueno que intentaran replicar el gráfico de ejemplo del profesor, para estar más seguros de que su solución funciona correctamente.
- Si tienen dudas con respecto al ramo, a la tarea o quedan pegados con un problema es bueno que acudan a profesores o ayudantes para que no pierdan mucho tiempo en el desarrollo de una tarea.
- Para las siguientes tareas el uso secuencial de commits será evaluado, consideren tener commits descriptivos y actualizados a cada cambio en su tarea. No subir commits con toda una etapa terminada.

VI. GRUPO 5 : 201630006-6 201630039-2 201630002-3

VI-A. Descuentos

- No se entrega gráfico de la etapa 1. **(-2 PUNTOS a Gráfico-Etapa 1)**
- No se incorpora diagrama de alto nivel para última etapa desarrollada. **(-5 PUNTOS a Diagramas UML)**
- No se explica la relación entre las clases principales. **(-5 PUNTOS a Explicación)**
- No se mencionan la cantidad solicitada de dificultades y no indica como fueron abordadas y/o solucionadas. **(-6 PUNTOS a Dificultades y Soluciones)**
- El robot no hace los giros correctamente para todos los casos, por ejemplo al avanzar en y, y doblar derecha y luego izquierda. **(-3 PUNTOS a Etapa 1)**
- Código no es robusto para tercera etapa, no genera rutas adecuadamente al cambiar la posición inicial, no opera ante configuración planteada en el código de ayuda. **(-8 PUNTO a Etapa 3)**
- Código genera el archivo .pbm de salida igual al archivo de entrada. **(-2 PUNTOS a Etapa 3)**
- No se llega a la salida del laberinto debido a que se termina antes la simulación. **(-7 PUNTOS a Etapa 4)**
- No se implementa adecuadamente la clase piloto, dado que existe momentos donde el vehículo decide atravesar los muros en vez de rodearlos. **(-3 PUNTOS a Etapa 4)**
- No se entregan archivos fuentes para etapa 5. **(-10 PUNTOS a Etapa 5)**
- Cómo no se entrega etapa 5, no se puede compilar **(-2 PUNTOS a Cero Warning, Cero Errores)**

VI-B. Recomendaciones

- No abusar de doxygen, para proyectos pequeños como esta tarea la cantidad de información que entrega es demasiada, por lo que dificulta encontrar la información a evaluar.
- No dejar atributos sin inicializar, ya que estos podrían tomar valores no deseados.
- Se les recomienda no generar más métodos de los solicitados o de los que ustedes estiman conveniente para llegar a su solución, ya que solo les termina por consumir más tiempo.
- Verificar la lógica integrada al código con respecto al momento de decidir si girar y hacia donde girar.
- No dejar carpetas con archivos ajenos a los solicitados ni archivos .class. Se puede realizar descuento por esto en tareas posteriores.

VI-C. Aspectos Destacados

- En general, buen uso de GIT, se demuestra desarrollo iterativo y colaboración de todos los miembros.
- Felicidades por la buena práctica de realizar los commits en inglés.
- Muchas felicitaciones por la cantidad de esfuerzo dedicado a comentar el código para doxygen, pero cuidado con dedicarle más tiempo a comentar que a programar la solución.

VII. GRUPO 6 : 201504100-8 201504126-1 201504012-5

VII-A. Descuentos

- No existe documentación **(-23 Puntos en Documentación)**.
- No se hace uso de un makefile para compilación de etapas **(-3 Puntos a Makefile)**.
- En la Etapa 2 no se utiliza bien el método para escribir la matriz resultante en un pbm, al arreglar este inconveniente la matriz resultante no cumple formato **(-7 Puntos a Etapa 2)**.
- **NullPointerException** no manejado no permite ejecutar el código, por lo que no se puede probar su funcionamiento. **(-15 Puntos a Etapa 3)**.
- Etapa 4 no realizada **(-15 Puntos a Etapa 4)**.
- Etapa 5 no realizada **(-10 Puntos a Etapa 5)**.

VII-B. Recomendaciones

- A pesar de no haber alcanzado a terminar todas las etapas debieron de avanzar y completar la documentación hasta donde hayan podido llegar, para no perder puntos en ese ítem, pues aun así podían optar a todo el puntaje.
- Deben e comenzar a utilizar un makefile para la compilación y/o ejecución de sus tareas. En la página del profesor existe una plantilla.
- Para compilar les faltó colocar que es el nombre del archivo **.java**.
- Muy bien la forma en que hicieron que el robot girara, es más cercana a la realidad, pero para esta tarea no era muy necesario, un giro brusco (cambio de valores) era suficiente.

- Una técnica muy simple para encontrar problemas es ir comentando y des comentando las líneas de los métodos que se muestran en la consola, hasta encontrar dónde exactamente se encuentra el error.
- Para la siguiente tarea el uso de GIT será evaluado, por lo que se les recomienda aprender a utilizarla y leer los requisitos de su uso, pues influirá en el puntaje de las siguientes tareas.

VII-C. Aspectos Destacados

- README muy explicativo, es muy bueno que comenten los problemas que tuvieron si su etapa no está haciendo lo correcto.
- Muy ordenado y muy bien comentado hasta donde se llegó, excelente! (+2 Puntos en Extra).

VIII. GRUPO 7 : 201303041-6 201303044-0 201104505-K

VIII-A. Descuentos

- No se entrega gráfico obtenido de la etapa 1. **-2 PUNTOS a Gráfico-Etapa 1**
- No se entrega diagrama de alto nivel de la última etapa desarrollada. **-5 PUNTOS a Diagramas UML**
- No se explica la relación entre las clases en las etapas ni en que consisten los módulos principales. **(-9 PUNTOS a Explicación)**
- No se mencionan las dificultades a la que se encontraron durante el desarrollo de la tarea y las soluciones para abordarlas. **(-9 PUNTOS a Dificultades y Soluciones)**
- Cuando el robot gira, el tiempo de la simulación se repite una vez, lo que puede conllevar desfases a largo plazo. **(-2 PUNTOS a Etapa 1)**
- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen. **(-5 PUNTOS a Etapa 3)**

VIII-B. Recomendaciones

- Tener más cuidado a la próxima entrega con respecto a los archivos que se encuentran en el repositorio, para tareas siguientes no se aceptarán archivos entregados después del tag *final*.
- Se deben realizar una mayor cantidad de commits, el repositorio no refleja un trabajo cooperativo ni incremental. Puede generar descuento en futuras tareas.
- Todos los integrantes deben realizar una cantidad similar de commits. Puede generar descuento en futuras tareas.
- Dejar mensajes más informativos en los commits, sin repetir y menos ambiguos. Puede generar descuento en futuras tareas.
- Entregar archivo de documentación para tareas siguientes, recordar que tiene valor en las notas de éstas.
- Entregar solo un archivo readme donde se detalle como compilar y ejecutar cada etapa.
- No dejar atributos sin inicializar, ya que estos podrían tomar valores no deseados.

IX. GRUPO 8 : 201573070-9 201890059-1 201790233-7

IX-A. Descuentos

- No se manejan las excepciones cuando el robot está fuera del laberinto, al utilizar el ejemplo dado por el profesor, el programa arroja un **ArrayOutOfBoundsException (-5 Puntos en Etapa 3)**.
- Código de la etapa 4 no es robusto y muchas veces arroja un **ArrayOutOfBoundsException (-5 Puntos en Etapa 4)**.
- Etapa 5 arroja un **ArrayIndexOutOfBoundsException** para cualquier combinación de entrada opcional completa y sin ella, y no posiciona adecuadamente al robot en el laberinto. Además siempre que se genera utiliza la misma estrategia de la etapa 4. **(-10 a Etapa 5)**.

IX-B. Recomendaciones

- En el README debieron de explicar la compilación y uso de su makefile en vez de la explicación general de compilación de un código en java.
- Consideren para la próxima incluir sus archivos de prueba (en este caso, los archivos pbm de entrada), esto sólo para hacer más rápida la corrección.
- Prueben su código en **Aragorn** antes de entregarlo, errores o excepciones que no puedan arreglarse de inmediato les cuesta el puntaje completo de la etapa.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se

realizan al proyecto con este commit, intentar no poner `^actualización en X`, expliquen qué actualización hicieron!). En tareas siguientes esto podría generar descuentos.

- Cuando se trabaja con arreglos siempre es necesario manejar o restringir los casos en que se pueda desbordar al recorrerlo, para este caso, cuando el robot sale del laberinto genera una excepción debido a que se intenta acceder a índices indefinidos o inexistentes del arreglo.

IX-C. Aspectos Destacados

- Muy buen README.
- Muy buena documentación de su tarea en general, muchas felicidades!
- Muy interesante utilizar la dirección unitaria del robot!

X. GRUPO 9 : 201421030-2 201321019-8 201404136-5

X-A. Descuentos

- Readme no menciona adecuadamente como compilar y ejecutar la tarea. **(-2 PUNTOS a Readme)**
- Makefile no esta bien redactado, llama a una clase no creada cuando se utiliza make run. **(-1 PUNTOS a Makefile)**
- No se explica la relación entre las clases en las etapas ni en que consisten los módulos principales. **(-9 PUNTOS a Explicación)**
- No se mencionan las soluciones para abordar las dificultades que se encontraron durante el desarrollo de la tarea. **(-4 PUNTOS a Dificultades y Soluciones)**
- Los datos de salida de la simulación no responden a cambios de velocidad del robot. **(-3 PUNTOS a Etapa 1)**
- Si se entrega un laberinto de tamaño distinto al laberinto de ejemplo en etapa 2, la imagen resultante se ve distorsionada. **(-3 PUNTOS a Etapa 2)**
- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen. **(-5 PUNTOS a Etapa 3)**
- En la cuarta etapa no se implementa adecuadamente la clase navegante, dado que solo puede tener como dirección inicial hacia abajo. **(-5 PUNTOS a Etapa 4)**
- Código no es robusto para quinta etapa, no se logra ejecutar sin que arroje un Out of Bounds Exception. **(-5 PUNTOS a Etapa 5)**
- Se hace uso de hardcoding en todas las etapas. **(2 PUNTOS a Descuentos)**

X-B. Recomendaciones

- Para siguientes tareas recordar mencionar como se solucionaron las dificultades encontradas y explicar más en la explicación de los módulos de la última etapa desarrollada.
- Rotular los gráficos en general y acostumbrarse a utilizar una escala más acorde a los datos mostrados.
- Mantener la formalidad en los comentarios.
- No hacer Hardcoding.
- Utilizar nombres más representativos en los archivos de las etapas, no poner el main en la Clase Stage4 para la etapa 5.

X-C. Aspectos Destacados

- En general, buen uso de GIT, se demuestra desarrollo iterativo y colaboración de todos los miembros.

XI. GRUPO 10 : 201404021-0

XI-A. Descuentos

- No agrega **tag Final** a su último commit **(-5 Puntos en Descuentos)**.
- Exceso de archivos basura **(-2 Puntos en Descuentos)**.
- No se entrega diagrama de alto nivel de la última etapa desarrollada. **(-5 Puntos a Diagrama UML)**
- README no explica cómo compilar y ejecutar cada una de las etapas. **(-2 Puntos a README)**.
- No hace uso de Makefiles **(-3 Puntos a Makefile)**.
- No presenta dificultades y soluciones **(-9 Puntos a Dificultades y Soluciones)**.
- No se realiza la Etapa 4 **(-15 Puntos en Etapa 4)**. No se realiza la Etapa 5 **(-10 Puntos en Etapa 5)**.
- Uso de **hardcoding** en Etapa1 **(-2 Puntos en Descuentos)**.
- Se revisa el código dentro de la carpeta `Etapa 1`, y genera un error de compilación **local variable timer is accessed from within inner class; needs to be declared final**. La etapa no puede corroborarse **(-10 Puntos en Etapa 1) (-2 Puntos a Errores en Cero Warnings, Cero Errores)**.
- Se revisa el código dentro de la carpeta `Etapa 3`, y genera un inmenso error de compilación, haciendo imposible su revisión **(-15 Puntos en Etapa 3) (-2 Puntos a Errores en Cero Warnings, Cero Errores)**.

XI-B. *Recomendaciones*

- Leer el reglamento de entrega de tareas en la plataforma informativa de tareas.
- Utilizar los códigos de ayuda del profesor para hacer su tarea. Se pierde tiempo de más haciendo todas las clases desde cero y no usando el método del profesor.
- Los códigos deben estar en sus respectivas carpetas, ahí es donde se revisará su funcionamiento.
- Probar su código en **Aragorn**, pues es dicho que es ahí donde los ayudantes revisan su funcionamiento.
- No agregar cráteres especiales si no es está utilizando un makefile para cambiar la convención.
- Hacer uso de un **Makefile** para la compilación de sus etapas.
- No colocar código fuera de las carpetas.
- Utilizar el **tag Final** Para el último commit de su proyecto, sólo por esta tarea se hace descuento, para las siguientes se toma como tarea no entregada y se evalúa con nota 0.
- El README se utiliza para mostrar de qué manera se compila y se ejecuta su código.
- No hacer uso de **hardcoding**.
- Los comentarios de los commits deben detallar cuál fue el cambio que se hizo.
- Borrar el debugging antes de entregar.

XI-C. *Aspectos Destacados*

- Hacer las clases desde cero y pensar en una propia solución y ejecución del código es un trabajo extra que de todas maneras se valora (**+2 Puntos en Extra**).

XII. GRUPO 11 : 201103032-K 201204122-8 201604009-9

XII-A. *Descuentos*

- No se entrega gráfico obtenido de la etapa 1. **-2 PUNTOS a Gráfico-Etapa 1**
- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen. **(-5 PUNTOS a Etapa 3)**
- Código de la cuarta etapa no siempre llega a la salida, debido a que algunas veces queda *Out of Bounds*. **(-5 PUNTO a Etapa 4)**
- Código no es robusto para quinta etapa, no se logra ejecutar sin que arroje un Out of Bounds Exception o entre en un loop y no termine de ejecutarse la rutina. **(-5 PUNTOS a Etapa 5)**

XII-B. *Recomendaciones*

- De lo visto en el avance realizado en GIT, se recomienda para las siguientes tareas ser más colaborativos, que no solo una persona escriba los commits, que se suban actualizaciones de forma más gradual y en general, dar mensajes más descriptivos e informativos en los commits.
- Ser más cuidadosos al copiar y pegar fragmentos de información, como en el readme.
- No dejar atributos sin inicializar, ya que estos podrían tomar valores no deseados.

XII-C. *Aspectos Destacados*

- Una documentación y un readme bastante adecuados, breves pero concisos.

XIII. GRUPO 12 : 201330015-4 201421003-5 201321074-0

XIII-A. *Descuentos*

- No se encuentra el gráfico pedido para la etapa 1 **(-2 Puntos a Gráfico-Etapa 1 en Documentación)**.
- No se explican los módulos ni la interacción entre ellos **(-5 Puntos a Explicación en Documentación)**.
- No se explica la estrategia utilizada en la etapa 5 **(-4 Puntos a Explicación en Documentación)**.

XIII-B. *Recomendaciones*

- El la documentación se debe de poner una breve explicación de los módulos y su interacción e informar lo que se estime conveniente para entender su solución, además de que el profesor pedía que se explicara la estrategia utilizada para la etapa 5 en la documentación.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se realizan al proyecto con este commit). En tareas siguientes esto podría generar descuentos.

XIII-C. Aspectos Destacados

- Excelente forma de explicar sus dificultades y sus soluciones, muy completo, se nota una dedicación. **(+2 Puntos a Extra)**.
- Excelente README.
- Muy buen manejo de errores.
- Un orden y un trabajo impecable, muchas felicitaciones!

XIV. GRUPO 13 : 201321034-1 201321007-4 201321056-2

XIV-A. Descuentos

- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen y se detiene cuando se encuentra con una pared mientras se mueve verticalmente. **(-5 PUNTOS a Etapa 3)**
- Código de la cuarta etapa no siempre llega a la salida, debido a que algunas veces queda *Out of Bounds* o en un loop. **(-5 PUNTO a Etapa 4)**

XIV-B. Recomendaciones

- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits, y darle un poco más de contexto e información a los mensajes de éstos. En tareas siguientes esto podría generar descuentos.
- No dejar atributos sin inicializar, ya que estos podrían tomar valores no deseados.

XIV-C. Aspectos Destacados

- En general, el que trabajó en el repositorio desarrolló una buena labor, sólo faltó un poquito más de contexto en los mensajes de los commits, explicar mejor que cambios se realizaron.
- Excelente documentación, muy detallada y agradable a la vista, demuestra de gran manera la labor desarrollada durante el transcurso de la tarea. **(2 PUNTOS a Puntos Extras)**.

XV. GRUPO 14 : 201321043-0 201730038-8 201704005-K

XV-A. Descuentos

- Archivos png se consideran archivos basura **(-2 Puntos a Descuentos)**.
- No se encuentra el gráfico pedido para la etapa 1 **(-2 Puntos a Gráfico-Etapa 1 en Documentación)**.
- Se encuentran archivos png de los diagramas de alto nivel pedidos, pero no están contenidos en una documentación formal. **(-2 Puntos a Diagramas UML en Documentación)**.
- README no explica cómo ejecutar cada etapa **(-1 Punto a README)**.
- No se encuentra el archivo de documentación **(-18 Puntos en Documentación)**.
- En la Etapa 3 el robot al encontrarse con una pared comienza a girar infinitamente en el mismo lugar **(-5 Puntos a La ruta generada es lógica en Etapa 3)**.
- En la Etapa 3 llamaban a métodos como si fueran atributos, por lo que generaba un error de compilación que tuvo que ser arreglado para poder ver el funcionamiento del código. **(-2 Puntos a Errores en Cero Warnings, Cero Errores)**.
- El robot de la Etapa 5 no obedece a las entradas opcionales, solo el robot de la estrategia anterior los atiende **(-3 Puntos en Etapa 5)**.

XV-B. Recomendaciones

- Los diagramas UML deben de estar integrados en una documentación, no estar como archivos de imagen sueltos en una carpeta.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se realizan al proyecto con este commit, intentar no poner *^actualización en X*, expliquen qué actualización hicieron!). En tareas siguientes esto podría generar descuentos.
- Leer las reglas de entrega de tareas en la plataforma informativa de tareas.

XVI. GRUPO 15 : 201203014-5 201304292-9

XVI-A. Descuentos

- No se entrega gráfico obtenido de la etapa 1. **-2 PUNTOS a Gráfico-Etapa 1**
- No se entrega diagrama de alto nivel de la última etapa desarrollada. **-5 PUNTOS a Diagramas UML**
- No se entregan makefile para compilar y ejecutar las etapas desarrolladas. **(-3 PUNTO a Makefile)**
- No se explica la relación entre las clases en las etapas ni en que consisten los módulos principales. **(-9 PUNTOS a Explicación)**
- No se mencionan las dificultades que se encontraron durante el desarrollo de la tarea ni las soluciones para éstas. **(-9 PUNTOS a Dificultades y Soluciones)**
- Los valores de salida no se ajustan adecuadamente al cambiar el valor de la velocidad el robot. **(-3 PUNTOS a Etapa 1)**
- La ruta no se guarda adecuadamente dado que se realiza una copia baja del laberinto, por lo que al avanzar el robot se generan nuevos muros en el laberinto original. **(-3 PUNTOS a Etapa 3)**
- Código de la cuarta etapa no siempre llega a la salida, debido a que algunas veces queda *Out of Bounds* o en un loop. **(-5 PUNTOS a Etapa 4)**
- No se entrega código fuente para la etapa 5. **(-10 PUNTOS a Etapa 5)**
- Cómo no se entrega etapa 5, no se puede compilar. **(-2 PUNTOS a Cero Warning, Cero Errores)**
- Se hace uso de hardcoding en la etapa 1. **(2 PUNTOS a Descuentos)**
-

XVI-B. Recomendaciones

- Respetar formato del nombre del repositorio para siguientes tareas. En tareas siguientes esto podría generar descuentos.
- Tratar de realizar el trabajo de forma más distribuida durante el período asignado para la tarea.
- Dar mensajes más significativo a los commits, no repetir mensajes y en general, dar más contexto a los cambios realizados.
- Tratar de tener todos los archivos en orden al momento de realizar tag *final*, ya que archivos subidos después de éste no serán considerados en tareas siguientes.
- No dejar en los comentarios de los códigos información irrelevante para el entendimiento de éstos.
- No dejar atributos sin inicializar, ya que estos podrían tomar valores no deseados.
- Utilizar apropiadamente los atributos de la clase, con el fin de evitar pasar un objeto como argumento cada vez que se utiliza una función.
- Tener más cuidado con los sentidos de giro, en la etapa 2 los laberintos están haciendo giros en sentido negativo.
- Respetar formato de carpetas contenedoras para la tarea. Esto podría generar descuento en tareas futuras.
- Generar un sólo archivo readme para todas las etapas, no es necesario generar uno para cada etapa.

XVII. GRUPO 16 : 201351025-6 201123008-6 201890108-3

XVII-A. Descuentos

- No se encuentra el gráfico pedido para la etapa 1 **(-2 Puntos a Gráfico-Etapa 1 en Documentación)**.
- No se encuentra el archivo de documentación **(-23 Puntos en Documentación)**.
- README no explica cómo compilar y ejecutar el código **(-2 Puntos a README en Documentación)**.
- Archivos .class en los directorios. **(-2 Puntos en Descuentos)**.
- En la Etapa 3 ocurre un **Null Pointer Exception** no manejado que no permite la ejecución de la etapa **(-15 Puntos en Etapa 3)**.

XVII-B. Recomendaciones

- A pesar de no haber logrado terminar con todas las etapas, debieron de documentar hasta donde llegaron para no perder todos los puntajes de la documentación.
- El README debe de explicar la forma de compilar y ejecutar su código.
- Cada día de atraso restan 5 puntos a la nota final de la tarea, pero consideren que un día más de trabajo puede hacerles ganar más puntaje del que perderán debido al descuento de atraso.
- Los archivos .class no deben incluirse en la entrega de la tarea, pues se generan al compilar su código.
- En el README se debe colocar la forma en que se compila y se ejecuta cada etapa.
- Consideren que al instanciar nuevamente una clase MyWorld se está creando un objeto diferente a la que se está utilizando en el Stage3.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se realizan al proyecto con este commit y no subir toda una etapa en un solo commit). En tareas siguientes esto podría generar descuentos.

XVIII. GRUPO 17 : 201604062-5 201630034-1

XVIII-A. Descuentos

- No se entrega ningún elemento asociado a la documentación. (-30 PUNTOS a Total Documentación)
- Los valores de salida no se ajustan adecuadamente al cambiar el valor de la velocidad el robot y no siempre respeta los giros indicados. (-5 PUNTOS a Etapa 1)
- Formato de archivos .pbm de salida no se respeta para laberintos con tamaño distinto a 70. (-4 PUNTOS a Etapa 2)
- No se puede evaluar etapa 3 porque falta definir un par de métodos en código fuente y por ende, no compila. (-15 PUNTOS a Etapa 3)
- No se entrega código fuente para la etapa 4. (-15 PUNTOS a Etapa 4)
- No se entrega código fuente para la etapa 5. (-10 PUNTOS a Etapa 5)
- Tres etapas no compilan. (-6 PUNTOS a Cero Warning, Cero Errores)
- Se incorporan muchas carpetas innecesarias y archivos basura. (1 PUNTO a Descuentos)
- Se hace uso de hardcoding en la etapa 1 y 2. (2 PUNTOS a Descuentos)

XVIII-B. Recomendaciones

- Recordar el uso de GIT para las próximas tareas, ya que se evalúa y tiene su respectiva ponderación en la nota final de éstas.
- Respetar formato de carpetas contenedoras para la tarea. Esto podría generar descuento en tareas futuras.
- El readme debe ir con información de como compilar las etapas y como ejecutarlas.
- Si no entienden bien alguna clase o alguna interacción de la tarea, acudir a los profesores y/o ayudantes, no esperen a tener una mala nota para preocuparse.

XIX. GRUPO 18 : 201530013-5 201630005-8 201621058-K

XIX-A. Descuentos

- Descuento por atraso en entrega. (-5 Puntos en Descuentos).
- No coloca **tag Final** al último commit (-5 Puntos en Descuentos).
- Múltiples archivos README (-2 Puntos en Descuentos).
- No se encuentra el gráfico pedido para la etapa 1 (-2 Puntos a Gráfico-Etapa 1 en Documentación).
- No se encuentra el archivo de documentación (-23 Puntos en Documentación).
- En la etapa 3 el robot se sale del laberinto generando un **ArrayIndexOutOfBoundsException** (-8 Puntos a Ruta generada es lógica en Etapa 3).
- No se hace uso de la clase Pilot en la Etapa 4 (-7 a Correcto funcionamiento de Pilot en Etapa 4).
- No recibe argumentos opcionales para personalización (-5 Puntos en Etapa 4).

XIX-B. Recomendaciones

- Las tareas sin **tag Final** en su último commit se consideran **no entregadas**. Solo por esta tarea se descontará puntaje por el no cumplimiento de esta regla.
- A pesar de no haber logrado terminar con todas las etapas, debieron de documentar hasta donde llegaron para no perder todos los puntajes de la documentación.
- Cada día de atraso restan 5 puntos a la nota final de la tarea, pero consideren que un día más de trabajo puede hacerles ganar más puntaje del que perderán debido al descuento de atraso.
- Era muy importante que utilizaran la clase piloto para controlar los movimientos del robot.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se realizan al proyecto con este commit, intentar no poner "actualización en X", expliquen qué actualización hicieron!). En tareas siguientes esto podría generar descuentos.
- Cuando se trabaja con arreglos siempre es necesario manejar o restringir los casos en que se pueda desbordar al recorrerlo, para este caso, cuando el robot sale del laberinto genera una excepción debido a que se intenta acceder a índices indefinidos o inexistentes del arreglo.

XIX-C. Aspectos Destacados

- Interesante forma de ejecutar la estrategia en la etapa 4.

XX. GRUPO 19 : 201630012-0 201630015-5 201630004-K

XX-A. Descuentos

- Readme solo menciona como compilar y ejecutar una etapa de la tarea. (-1 PUNTOS a Readme)
- No se explica la relación entre las clases principales. (-5 PUNTOS a Explicación)
- No se mencionan ni indican como fueron abordadas y/o solucionadas las dificultades encontradas. (-4 PUNTOS a Dificultades y Soluciones)
- Código de la cuarta etapa no siempre llega a la salida, debido a que algunas veces queda *Out of Bounds* o en un loop. (-5 PUNTO a Etapa 4)
- No se entrega código fuente para la etapa 5. (-10 PUNTOS a Etapa 5)
- Cómo no se entrega etapa 5, no se puede compilar. (-2 PUNTOS a Cero Warning, Cero Errores)

XX-B. Recomendaciones

- Los mensajes entregados en los commits deben tener información sobre los cambios realizados, manteniendo un lenguaje formal y ser breves, pero concisos. En tareas siguientes esto podría generar descuentos.
- Rotular adecuadamente los ejes de los gráficos a utilizar.

XX-C. Aspectos Destacados

- Se demuestra uso colaborativo del repositorio.
- El formato utilizado para la redacción del readme es agradable a la vista.

XXI. GRUPO 20 : 201430040-9 201430052-2 201430035-2

XXI-A. Descuentos

- No se encuentra el gráfico pedido para la etapa 1 (-2 Puntos a Gráfico-Etapa 1 en Documentación).
- Se pedían a lo menos tres problemas y sus soluciones (-7 Puntos a Dificultades y Soluciones en Documentación).
- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen. (-5 PUNTOS a Etapa 3)
- A ejecuta el código sin entradas opcionales se genera automática mente un **ArrayIndexOutOfBoundsException**. Además al colocar los parámetros opcionales el robot hace caso omiso, ni si quiera cambia su dirección y la salida no corresponde a la ingresada, finalmente jamás sale del laberinto (-10 Puntos en Etapa 4).
- Exactamente el mismo problema que en la etapa 4, pero además no presenta ninguna propuesta diferente (-10 Puntos en Etapa 5).

XXI-B. Recomendaciones

- El nombre del proyecto debe tener el formato indicado en la plataforma informativa de tareas. **apellido1.nombre1-apellido2.nombre2-apellido3.nombre3-TareaX**. Para la siguiente tarea este ítem tendrá un descuento.
- Deben de ver y seguir cada uno de los pasos de la entrega de tareas que se encuentra en la plataforma informativa de tareas.
- Los mensajes entregados en los commits deben tener información sobre los cambios realizados, manteniendo un lenguaje formal y ser breves, pero concisos. En tareas siguientes esto podría generar descuentos.
- Cuando se trabaja con arreglos siempre es necesario manejar o restringir los casos en que se pueda desbordar al recorrerlo, para este caso, cuando el robot sale del laberinto genera una excepción debido a que se intenta acceder a índices indefinidos o inexistentes del arreglo.

XXI-C. Aspectos Destacados

- Excelente README.
- Uso colaborativo de commit, muy bien!.

XXII. GRUPO 21 : 201030013-7 201030003-K

XXII-A. Descuentos

- Se hace uso entrega atrasada del proyecto. (5 PUNTOS a Descuentos)

XXII-B. *Recomendaciones*

- Recordar el uso de GIT para las próximas tareas, ya que se evalúa y tiene su respectiva ponderación en la nota final de éstas.

XXII-C. *Aspectos Destacados*

- Excelente archivo de documentación, un formato muy ordenado y agradable a la vista. Breve y conciso y cumple con todos los criterios de evaluación.
- Buen desarrollo de la tarea, cumple con todos los requisitos solicitados además de ser bastante robusta ante diversos tipos de pruebas.

XXIII. GRUPO 22 : 201121020-4 201121032-8

XXIII-A. *Descuentos*

- Un día de atraso (**-5 Puntos en Descuentos**).
- No existe un archivo de documentación, pero se cuenta la explicación de su solución dentro del README. (**-9 Puntos en Documentación**).
- No se hace uso de Makefiles para la compilación de sus etapas (**-3 Puntos a Makefile en Documentación**).
- No es necesaria la generación de paquetería, crea demasiados archivos basura para esta ocasión. (**-2 Puntos en Descuentos**).
- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen. (**-5 Puntos a Etapa 3**)

XXIII-B. *Recomendaciones*

- Repositorio duplicado puede generar una nota 0 para futuras tareas, tener cuidado con eso.
- La explicación de sus soluciones deben de estar en un documento formal pdf correspondiente a la documentación, para el README solo es necesario colocar cómo se compila y se ejecuta su código, además de aclaraciones pertinentes.
- Deben de ver y seguir cada uno de los pasos de la entrega de tareas que se encuentra en la plataforma informativa de tareas.
- La manera de compilar es `javac -g Stage(n).java`.
- Hay que limitarse a colocar el código en la carpeta de la etapa y utilizar los **Makefiles** para la compilación de cada etapa.
- El diagrama UML de su última etapa debe presentarse en una documentación general y no en un archivo de imagen suelto dentro de la última etapa.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se realizan al proyecto con este commit, intentar no poner "actualización en X", expliquen qué actualización hicieron!, y no subir toda una etapa en un commit!). En tareas siguientes esto podría generar descuentos.

XXIII-C. *Aspectos Destacados*

- Se logra una buena robustez en la etapa 4.

XXIV. GRUPO 23 : 201273065-1 201790232-9 201430022-0

XXIV-A. *Descuentos*

- No se entrega gráfico obtenido de la etapa 1. **-2 PUNTOS a Gráfico-Etapa 1**
- No se mencionan ni indican como fueron abordadas y/o solucionadas las dificultades encontradas. (**-4 PUNTOS a Dificultades y Soluciones**)
- El manejo del tiempo de simulación es incorrecto y esto lleva a que no se represente adecuadamente la simulación. (**-2 PUNTOS a Etapa 1**)
- Formato de archivos .pbm de salida no se respeta para laberintos con tamaño distinto a 70. (**-4 PUNTOS a Etapa 2**)
- Código fuente no compila, por lo que no se puede evaluar el funcionamiento de esta etapa. (**-15 PUNTOS a Etapa 4**)
- No se entrega código fuente para la etapa 5. (**-10 PUNTOS a Etapa 5**)
- Cómo no se entrega etapa 5, no se puede compilar. Además, tampoco compila la etapa 4. (**-4 PUNTOS a Cero Warning, Cero Errores**)

XXIV-B. *Recomendaciones*

- Respetar formato del nombre del repositorio para siguientes tareas. En tareas siguientes esto podría generar descuentos.
- Los mensajes entregados en los commits deben tener información sobre los cambios realizados y ser breves, pero concisos. En tareas siguientes esto podría generar descuentos.
- Se debe realizar un mayor número de commits para se refleje adecuadamente el progreso de la tarea, además de tener respaldos más actuales en caso de realizar un cambio que dañe su código.
- Todos los integrantes del grupo deberían realizar un número similar de commits.
- El formato de la documentación no es muy agradable de ver, se puede ordenar para llevarlo a una mejor forma justificando los párrafos, añadiendo sangrías y cuidando más la redacción.

XXV. GRUPO 24 : 201103007-9 201604147-8

XXV-A. *Descuentos*

- Un día de atraso (**-5 Puntos en Descuentos**).
- No se entrega gráfico obtenido de la etapa 1. (**-2 Puntos a Gráfico-Etapa 1**)
- No se incluye un diagrama de alto nivel de su última etapa desarrollada (**-5 a Diagramas UML en Descuentos**).
- No explican sus Dificultades y soluciones (**-9 a Dificultades y soluciones en Documentación**).
- Código no es robusto para tercera etapa, no se propone solución para cuando el robot puede salir fuera de la imagen. (**-5 Puntos a Etapa 3**)
- Existe un conflicto en la Etapa 4 con el recibimiento de los parámetros de entrada, si coloco 2 parámetros no está recibiendo. Además el robot va demasiado cerca de la pared y cuando sale del laberinto el programa entra en loop y la batería no se agota. (**-5 Puntos en Etapa 4**).
- No se realiza la etapa 5 (**-10 Puntos en Etapa 5**).

XXV-B. *Recomendaciones*

- No es necesario la generación de paquetería ni las extinciones de las clases, aunque se entiende sus intenciones. Se recomienda violentamente utilizar el formato requerido para las siguientes tareas y seguir las instrucciones, pues permite generalizar la revisión de mejor manera. A la siguiente tarea se hará descuento.
- Para la etapa 3 en adelante, pueden copiar y pegar sus soluciones de etapas anteriores y modificar lo que sea necesario.
- Cuando se trabaja con arreglos siempre es necesario manejar o restringir los casos en que se pueda desbordar al recorrerlo, para este caso, cuando el robot sale del laberinto, en la etapa 3, genera una excepción debido a que se intenta acceder a índices indefinidos o inexistentes del arreglo.
- El recorrido del robot debe ser apreciable en el archivo resultante.
- **Deben agregar a los 2 ayudantes al repositorio** con permisos de **Master**, pues la revisión de tareas es aleatoria.
- Para la siguiente tarea tratar de que todos los miembros realicen el mismo número de commits (los miembros menos participativos tendrán menor puntaje final), y darle contexto e información a los mensajes de éstos (qué cambios se realizan al proyecto con este commit, intentar no poner `^actualización en X`, expliquen qué actualización hicieron!). En tareas siguientes esto podría generar descuentos.

XXV-C. *Aspectos Destacados*

- Un buen manejo de los archivos makefile y uso de paquetería, a pesar que no se requería. (**+2 en Extra**).
- Muy bien por hacer la documentación en formato web.
- Muy bien pensado lo de la batería para abordar temas de looping.