

Informe Tarea 1 - Métodos Numéricos

Bruno Quezada

27 de septiebre

Asignatura: Métodos Numéricos - FI 3104-1
Profesor Cátedra: Valentino González
Auxiliares: José Vines, Jou-Hui Ho

1 Pregunta 1

1.1 Describa la idea de escribir el main driver primero y llenar los huecos después ¿Por qué es buena idea?

Porque desde el punto de vista humano, es más fácil comprender y testear el código de esta manera. Al poner primero el main driver e ir rellenando después, permite visualizar todos los pasos que son necesarios en forma simple. En el caso contrario que se escriba el programa como una sola gran función, es muy engorroso para evaluar y testear ya que al estar todo ligado no se puede evaluar el funcionamiento de cada parte por separado. El tiempo extra que toma programar en forma más inteligible para el programador será compensado al simplificar las tareas de mejorar la eficiencia de cada una de sus partes y encontrar errores de programación.

1.2 ¿Cuál es la idea detrás de la función `mark_filled`? ¿Por qué es buena idea crearla en vez del código original al que reemplaza?

La idea es que a esta función se le agrega un manejo de errores que puede evaluar si los parámetros ingresados están bien, además de entregar un mensaje de error más preciso para el usuario. Otra ventaja es que al programarla como una función separada, se le pueden realizar test en forma independiente para poder evaluar si la función está cumpliendo lo deseado.

1.3 ¿Qué es refactoring?

El refactoring es realizar cambios a la arquitectura de un código con el fin de mejorar su calidad, sin cambiar el comportamiento de este.

1.4 ¿Por qué es importante implementar test que sean sencillos de escribir? ¿Cuál es la estrategia usada en el tutorial?

Es importante porque, por lo general, es necesario implementar muchos test a nuestros programas y si estos son engorrosos de escribir será muy costoso testear el programa. La estrategia utilizada en el tutorial consiste en utilizar datos de tipo string para realizar el mapeo de la grilla, en vez de utilizar únicamente ciclos *for* o *while*.

1.5 El tutorial habla de dos grandes ideas para optimizar programas ¿Cuáles son esas ideas? Descríbalas

Intercambio memoria y tiempo de cómputo: almacenar información redundante para prevenir recalcular valores implica un mayor uso de memoria pero un menor tiempo de cómputo.

Intercambio de tiempo de programación con eficiencia del procesador:
trade human time for machine performance

1.6 ¿Que es lazy evaluation?

La "lazy evaluation" consiste en no calcular valores de variables del programa si no han de ser usados. En el caso del tutorial, la función no calcula valores de las celdas que no sean adyacentes a la zona rellena, ya que no clasifican como candidatos.

1.7 Describa la "other moral" del tutorial

Consiste en partir escribiendo una versión simple de implementar y fácil de entender primero para luego luego implementar de a poco las mejoras, atacando cada parte por separado.

2 Pregunta 2

2.1 Introducción

2.2 Procedimiento

2.3 Resultados

2.4 Análisis y Conclusiones