

ST0257 – Sistemas operativos
Proyecto 1: Simulación de Trading

PhD. Edison Valencia Díaz

MBA, I.S. José Luis Montoya Pareja

Departamento de Informática y Sistemas
Universidad EAFIT
Medellín, Colombia, Suramérica

RESUMEN

El trading es una actividad que requiere de alto poder de cómputo y de altos tiempos de respuesta. Es primordial que los tiempos de respuesta y el procesamiento de tareas sea casi en tiempo real. Simular el funcionamiento del mercado de Trading permite tener una vista de los problemas donde pueden aplicarse los conceptos que se verán en el curso de sistemas operativos durante el semestre.

PALABRAS CLAVE

Trading, procesamiento en paralelo, paradigmas de programación en paralelismo, eficiencia, procesos, virtualización, memoria.

CONTEXTO

Trading

Traducir literalmente la palabra *trading* es solo un muy pequeño aspecto de todo el contexto que a hoy implica o se usa como sinónimo.

Trading literalmente es Comercio. La definición que se encuentra en el diccionario de Oxford dice que es “la acción o actividad de comprar y vender bienes y servicios”. [1]

El *e-trading* es un término que ha venido cogiendo mucha fuerza desde la década de los 90's. El diccionario de Cambridge lo define como “un método de comercializar acciones, monedas, etc., a través de mercados electrónicos de acciones como el NASDAQ.” [2].

Para que se hagan una idea de la magnitud y de la importancia del mercado de e-trading, el sitio web *The Desk* (<https://www.fi-desk.com/>) reportó que solo el operador Tradeweb había reportado un volumen total de US\$27.4 Trillones tranzados durante el mes de Febrero de 2023. [3] Eso se traduce en los siguientes valores:

Tabla 1. Cantidad de dólares procesador por el operador Tradeweb en Febrero de 2023.

Valor	Unidad
\$ 27.400.000.000.000,00	Mes
\$ 978.571.428.571,43	Día
\$ 40.773.809.523,81	Hora
\$ 679.563.492,06	Minuto
\$ 11.326.058,20	Segundo

Fuente: Elaboración propia.

Aunque hay instituciones financieras, personas o gobiernos que en una sola transacción podrían mover US\$11.2 millones de dólares en una sola transacción, es prácticamente imposible que por segundo una sola transacción mueva tal cantidad de dinero. Vamos a suponer que una transacción mueva en promedio US\$10,000 en el mercado. Eso nos da un total de 1,132.61 transacciones por segundo en un solo operador de los que operan en Wall Street.

Antes de describir cómo opera el mercado de valores, vamos a describir cómo funciona un mercado de acuerdo con las definiciones básicas de economía:

1. **Oferta** representa la cantidad de bienes o servicios que los productores están dispuestos a vender en el mercado a diferentes precios. Si el precio de un bien es alto, generalmente hay más productores que están dispuestos a ofrecer más unidades al mercado.
2. **Demanda** representa la cantidad de bienes o servicios que los consumidores están dispuestos a comprar en el mercado a diferentes precios. Si el precio de un bien es bajo, los consumidores están dispuestos a adquirir más unidades.

Los productores fijan el precio de acuerdo con un costo de producción y un margen de ganancia, que es la utilidad adicional (ganancia) que el productor quiere generar sobre el producto o servicio.

En el caso del Trading, existen unos actores que participan del mercado:

1. **Broker o Corredor de bolsa** es el intermediario entre el inversor y el mercado. Es una empresa que debe de estar debidamente regulada y autorizada por la autoridad competente de cada país si quiere ser legal y poder ofrecer sus servicios en dicho país. [4]
2. Un **trader** es una persona física que invierte dinero en los mercados, comprando y vendiendo activos, en busca de una ganancia o plusvalía económica. [4]
3. El precio con el que un trader le compra al bróker una moneda o acción se denomina **bid**. [5]
4. El precio con el que un trader le vende al bróker una moneda o acción se denomina **ask**. [5]

5. La diferencia entre el precio bid y el ask en un mercado se denomina **spread. [6]**

Durante cada minuto puede hacerse un número muy grande de transacciones por montos increíblemente grandes. Esta información se recopila en las plataformas de trading y se graba para su posterior descarga y análisis financiero en las diferentes herramientas de visualización de trading existentes en el mercado. Este proyecto desarrollará durante todo el semestre, los ambientes necesarios para realizar una **simulación de trading** basada en dichas bases de datos.

Simulación de Trading

Se les va a suministrar una base de datos de diferentes combinaciones de precios de monedas. Esta información deberá ser leída por un proceso que lea una moneda en particular y el período de tiempo donde se recopila la información.

La base de datos está en un formato archivo tipo CSV o JSON (se entrega a los estudiantes ambos) y de cada moneda se tienen diferentes períodos de tiempo: **Diaria, cada 4 horas, cada hora, cada 30, 15, 5 y 1 minuto.**

Para el siguiente ejemplo, usaremos el caso del archivo CSV de una hora. Dentro de los archivos, aparece la siguiente información de ejemplo:

```
2013-05-23 07:00,101.844,101.929,101.435,101.566,60  
2013-05-23 08:00,101.569,101.599,101.035,101.365,113
```

La primera columna es la fecha y la hora de la información (23 de mayo del 2013 a las 7 a.m.) La segunda columna es el precio de apertura del mercado a las 7:00 a.m (101.844). La tercera columna es el valor más alto que tuvo el mercado en ese intervalo de tiempo (de 7:00:00 a 7:59:59 a.m., 101.929), mientras que la cuarta columna es el valor más bajo que tuvo el mercado en esa hora (101.435); la siguiente es el último precio que tuvo el mercado a las 7:59:59 (101.566), es decir, el precio de cierre. Y la última columna es la cantidad de operaciones que se realizaron durante ese lapso (60 operaciones).

La simulación se realizará de la siguiente forma:

1. Un proceso debe leer el archivo con la información de los precios antes descrito. Se lee la información de una sola moneda en un período de tiempo dado (por ejemplo, precios Euro vs Dólar en lapsos de 30 minutos).
2. A medida que la información es colocada en la memoria, otro proceso la debe graficar en pantalla. Un ejemplo de cómo graficar la información es el siguiente:

Figura 1. Ejemplos de velas japonesas. Fuente: Construcción propia.



Las velas de color rojo indican que el precio está descendiendo, mientras que las de color verde indican que el precio está ascendiendo.

3. El ejercicio de simulación viene dado por la cantidad de operaciones que se están realizando en el lapso dado. En el ejemplo anterior, de 7 a 8 a.m. se hicieron 60 operaciones donde la primera operación se hizo a 101.844 y la última a 101.566.

La simulación dibuja la vela partiendo del precio de apertura y se deben generar 58 valores aleatorios entre el mínimo y el máximo de cada unidad de tiempo. El precio de cierre va variando hasta que se generan los 58 valores aleatorios y el último precio que nos coloca en la gráfica (precio de cierre) es donde debe quedar cada vela.

4. Las velas se van pintando de izquierda a derecha.
5. Con la misma información se puede tener una estimación de la tendencia mediante el uso de **promedios móviles**. Estos se explican en la referencia [7]. Se debe graficar también (y en un proceso aparte) un promedio móvil simple (SMA) de 5 períodos y de otro color, un promedio móvil de 13 períodos. Los promedios móviles solo se deben empezar a graficar hasta tanto se cumplan los períodos mínimos para generar un valor promedio.
6. El proceso se repite hasta que se leen todas las líneas del archivo y se grafican todas las velas y promedios móviles.

ACTIVIDADES

Construir un programa que realizar los pasos antes descritos y muestre en pantalla el comportamiento de la moneda especificada.

La descripción del comando es la siguiente:

NOMBRE

simtrading – Simulación de operación de trading

SINOPSIS

simtrading [OPCIONES] -m=PAR FILE

DESCRIPCION

Lee la información desde el archivo en formato CSV o JSON y muestra en pantalla las velas japonesas correspondientes a la información almacenada y las líneas de tendencia mediante promedios móviles simples (SMA).

-p=PERIODO

Indica el período de tiempo desde donde se recolectó la información. Los valores de PERIODO permitidos son (*case sensitive*):

M1

M5

M15

M30

H1

H4

D1

Si el parámetro -p no se incluye, se asume que por defecto el período es H1.

-f=FORMATO

Indica el formato usado en el archivo de entrada. Los valores de FORMATO permitidos (*case sensitive*) son:

CSV

JSON

-m=PAR

Se le informa al programa la moneda o acciones o materiales **commodities** [8] que se utilizará para el análisis. Los valores de PAR permitidos son:

PAR	Descripción
BRENTCMDUSD	Petróleo Crudo Brent
BTCUSD	Bitcoin
EURUSD	Euro
GBPUSD	Libra esterlina
USA30IDXUSD	Dow Jones
USA500IDXUSD	Standard and Poor's (S&P)
USATECHIDXUSD	Nasdaq

XAGUSD	Plata
XAUUSD	Oro

Este parámetro es *obligatorio*.

Estado de salida del proceso:

- 0 Si el proceso termina OK
- 1 Si el proceso termina con errores

FILE es el archivo donde está guardada la información de la moneda.

CONSIDERACIONES GENERALES

1. El desarrollo de la práctica puede ser individual o en equipos de máximo tres personas.
2. La entrega de la práctica se realizará entregando los fuentes y el informe por el buzón recepción de trabajos de Eafit Interactiva (cualquier otro medio no será admitido).
3. Se debe informar al profesor a más tardar el 28 de julio a las 6:00 p.m. los integrantes del equipo.
4. El informe final deberá contener una breve descripción de cómo funciona el programa, que dificultades debieron superar para el desarrollo de la práctica y posibles mejoras que consideran, se puede hacer a la misma.
5. La práctica se puede realizar en cualquier lenguaje de programación. En el informe deben informar cual es la versión del compilador o del runtime que están usando.
6. Cada semana los jueves, se sacará un espacio de 10 a 15 minutos al inicio de la clase para hablar de la práctica y resolver dudas.
7. Criterios de evaluación (ver Anexo 1)

FECHA DE ENTREGA

Jueves 17 de agosto en clase a través de Eafit Interactiva.

SUSTENTACIÓN

Jueves 17 de agosto en clase. El mecanismo de sustentación es el siguiente:

1. Cada equipo muestra su desarrollo ejecutando en vivo.
2. Debe mostrar cómo solucionó cada uno de los retos y cómo lo relacionaron con conceptos vistos en clase.
3. Se realizarán preguntas por parte del docente para validar el entendimiento individual de los conceptos aplicados. Esto significa que, aunque el trabajo es en equipo, la nota del trabajo puede ser diferente para cada uno de acuerdo con la calidad de las respuestas.

REFERENCIAS

- [1] <https://www.google.com/search?client=firefox-b-d&q=what+is+trading>
- [2] <https://dictionary.cambridge.org/us/dictionary/english/e-trading>
- [3] <https://www.fi-desk.com/e-trading-platforms-see-booming-credit-volumes/>
- [4] <https://es.investing.com/academy/trading/que-es-un-broker-que-es-un-trader-y-diferencias/>
- [5] <https://www.investing.com/education/terms/bid-and-ask-prices-200357973>
- [6] <https://www.investopedia.com/terms/s/spread.asp>
- [7] <https://www.investopedia.com/terms/m/movingaverage.asp>
- [8] <https://www.investopedia.com/terms/c/commodity.asp>

Nombre de la asignatura: Sistemas Operativos

Competencia a la que aporta la asignatura: Conocer el sistema operativo del computador para un mejor desarrollo, diseño y ejecución de las aplicaciones y aplicar nuevas soluciones

Resultado de asignatura evaluado: Construcción de una aplicación de simulación de trading.

Evento evaluativo: Proyecto 1

Porcentaje del evento evaluativo: 30%

Criterios (que tributen al RA de asignatura)	Cumple con altos estándares (4.5-5)	Cumple a satisfacción (4-4.4)	Cumple parcialmente (3.5-3.9)	Incumple parcialmente (2.5-3.4)	Incumple totalmente (0-2.4)	Peso asignado al criterio sobre la calificación.
Análisis de fundamentación para la solución: Virtualización de procesos y memoria Claridad en el concepto	Entiende completamente la necesidad y plantea varias opciones de solución. Utiliza conceptos adecuadamente para la solución.	Entiende completamente la necesidad y plantea una opción de solución.	Omitió un elemento clave para el entendimiento de la necesidad	Omitió varios elementos para el entendimiento de la necesidad.	Demuestra poco o nulo entendimiento del problema.	40%
Diseño de la solución Solución óptima	El diseño tiene en cuenta los conceptos vistos en clase y argumenta la elección de su solución. La solución elegida es la óptima para el problema.	El diseño tiene en cuenta los conceptos vistos en clase y argumenta la elección de su solución.	Aunque se tuvieron en cuenta los conceptos vistos en clase, no hubo argumentación correcta en la elección de la solución.	No se tuvieron en cuenta los conceptos vistos en clase.	Demuestra poco o nulo entendimiento de patrones de paralelismo al momento de explicar la solución.	40%
Funcionalidad Calidad de la solución frente a necesidad planteada	El programa funciona correctamente.	La solución elegida no es la óptima pero el programa funciona correctamente	El programa con los conceptos vistos en clase no funciona correctamente.	El programa no tiene en cuenta ningún concepto visto en clase y la solución funciona	Se entrega la solución parcialmente o no se entrega ninguna solución.	20%

Criterios (que tributen al RA de asignatura)	Cumple con altos estándares (4.5-5)	Cumple a satisfacción (4-4.4)	Cumple parcialmente (3.5-3.9)	Incumple parcialmente (2.5-3.4)	Incumple totalmente (0-2.4)	Peso asignado al criterio sobre la calificación.
				correcta o parcialmente.		