

## **TRABAJO PRÁCTICO N.º 1 – Visualización de mensajes GPS en formato NMEA**

### **1) Objetivo**

El objetivo de este T.P. consiste en desarrollar un aplicativo de consola con comandos en línea de órdenes y escrito en lenguaje ANSI C, que permita interpretar mensajes GPS en formato NMEA a partir de un archivo y presentado en formato CSV y KML<sup>1</sup>.

### **2) Alcance**

Mediante el presente T.P. se busca que el alumno adquiera y aplique conocimientos sobre los siguientes temas:

- programas en modo consola;
- argumentos en línea de órdenes (*CLA*);
- modularización;
- *makefile*;
- archivos de texto y binarios;
- memoria dinámica;
- punteros a función;
- tipo de dato abstracto (T.D.A. Vector, T.D.A. *ad hoc*);
- estructura básica de un archivo CSV;
- estructura básica de un documento KML;
- estructura básica de los mensajes GPS en formato NMEA.

### **3) Desarrollo**

En este T.P. se pide escribir un programa ejecutable que interprete mensajes GPS en formato NMEA a partir de un archivo de texto y genere archivos de salida en formato CSV y KML con la información de localización geográfica de la trayectoria provista.

El programa ejecutable, denominado “gpsviewer.exe” (WinXX) o “gpsviewer” (Unix), debe ser invocado de la siguiente forma:

WinXX:

```
gpsviewer -fmt <formato> -out <salida> <entrada>
```

UNIX:

```
./gpsviewer -fmt <formato> -out <salida> <entrada>
```

Los comandos en línea de órdenes utilizados por la aplicación son los indicados a continuación.

#### **Formato del índice a generar**

Comando	Descripción	Valor	Tipo de dato
-fmt	Formato del índice a generar	“csv”	Cadena de caracteres
		“kml”	Cadena de caracteres

---

<sup>1</sup> <https://developers.google.com/kml/documentation/kmlreference?hl=es>

### Archivo de salida <salida>

Es la ruta del archivo índice que se desea construir.

### Archivos de entrada <entrada>

Es la ruta del archivo de texto que contiene los mensajes GPS.

Nota: Se puede asumir que los comandos en línea de órdenes estarán en cualquier orden (en pares), si esta estrategia simplifica el desarrollo de la presente aplicación, pero se debe consignar la decisión en el informe.

### Formato de entrada

Los archivos de entrada contienen una serie de mensajes GPS en formato NMEA. Con el enunciado del trabajo práctico se adjunta el manual de referencia de un aparato GPS marca SIRF, en donde se describen los distintos mensajes GPS basados en el estándar NMEA-0183<sup>2</sup>. Los datos de localización geográfica se deberán obtener de los mensajes de tipo GGA que proveen la información esencial de ubicación sobre el geoide terrestre y la información de la precisión de la solución del receptor GPS.

Ejemplo de datagrama GGA

\$GPGGA,002153.000,3342.6618,N,11751.3858,W,1,10,1.2,27.0,M,-34.2,M,,0000\*5E

Nombre	Ejemplo	Unidad	Descripción
ID Mensaje	\$GPGGA		Encabezado del protocolo
Tiempo UTC	2153		hhmmss.sss
Latitud	3342.6618		ddmm.mmmm
Indicador N/S	N		N=norte o S=sur
Longitud	11751.3858		dddmm.mmmm
Indicador E/W	W		E=este o W=oeste
Indicador de posición fija	1		Ver documentación.
Satélites utilizados	10		Rango: 0 a 12
HDOP	1.2		Dilución de precisión horizontal
Altura sobre el nivel del mar	27	metros	
Unidad	M	metros	
Separación del geoide	-34.2	metros	
Unidad	M	metros	
Age of Diff. Corr.		sec	Campo nulo para receptores no diferenciales
Diff. Ref. Station ID	0		
Suma de verificación	*5E		
<CR> <LF>			Fin del mensaje

---

<sup>2</sup> [https://www.nmea.org/content/nmea\\_standards/nmea\\_0183\\_v\\_410.asp](https://www.nmea.org/content/nmea_standards/nmea_0183_v_410.asp)

La suma de verificación es una verificación compuesta de un carácter en formato hexadecimal definido según la especificación NMEA. Consiste de una operación XOR de los 7 bits menos significativos de cada carácter después del “\$” y antes del “\*”. El resultado binario de la operación es representado en el equivalente ASCII de dos caracteres hexadecimales. Cada mensaje interpretado deberá ser verificado.

Todos los mensajes de de tipos distintos al GGA deberán ser ignorados.

Además, por un error en la manipulación de los archivos las líneas se han mezclado aleatoriamente, por lo tanto será necesario ordenarlos por el campo de Tiempo UTC para poder reconstruir las trayectorias correctamente.

### Formatos de salida

#### a) Formato CSV

El formato del documento CSV de salida a generar es:

```
<Fecha>|<latitud>|<longitud>|<altura sobre el nivel del mar>
...
<Fecha>|<latitud>|<longitud>|<altura sobre el nivel del mar>
```

Se trata de un archivo CSV sin encabezado, campos sin calificador y carácter separador “|”.

El formato para la fecha deberá ser el siguiente:

Formato	Significado
AAAAMMDDHHmmSS	Año (4 dígitos), mes (2 dígitos), día (2 dígitos), hora (2 dígitos), minuto (2 dígitos), segundo (2 dígitos)

Será necesario interpretar la hora, los minutos y los segundos contenidos dentro de los mensajes de GPS y luego transformarlos a hora local del sistema utilizando el día, mes y año del sistema, dado que estos datos no se encuentran dentro de dicho mensaje.

#### b) Formato KML

A continuación se presenta un ejemplo de la trayectoria representada en formato KML, para más información referirse a la documentación oficial<sup>3</sup>.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Document>
    <name>Rutas</name>
    <description>Ejemplos de rutas</description>
    <Style id="yellowLineGreenPoly">
      <LineStyle>
        <color>7f00ffff</color>
        <width>4</width>
      </LineStyle>
      <PolyStyle>
        <color>7f00ff00</color>
      </PolyStyle>
    </Style>
```

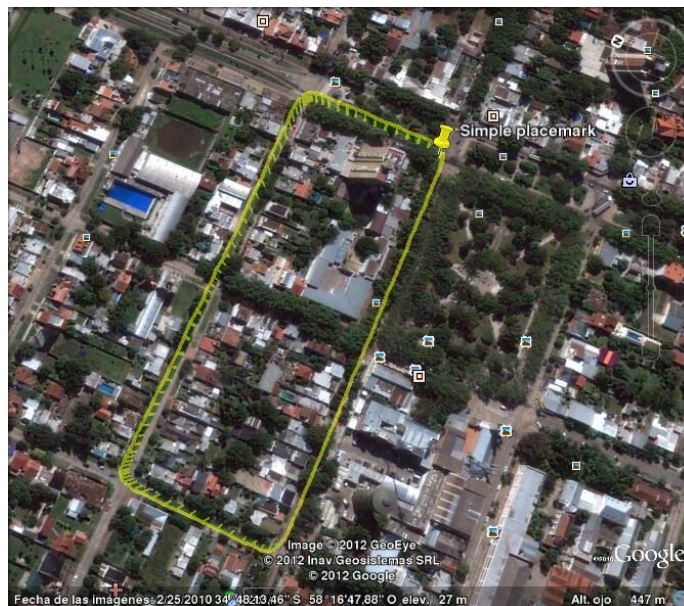
<sup>3</sup> [https://developers.google.com/kml/documentation/kml\\_tut](https://developers.google.com/kml/documentation/kml_tut)

```

    <Placemark>
      <name>Relieve absoluto</name>
      <description>Pared verde transparente con contornos
amarillos</description>
      <styleUrl>#yellowLineGreenPoly</styleUrl>
      <LineString>
        <extrude>1</extrude>
        <tessellate>1</tessellate>
        <altitudeMode>absolute</altitudeMode>
        <coordinates> -112.2550785337791,36.07954952145647,2357
-112.2549277039738,36.08117083492122,2357
-112.2552505069063,36.08260761307279,2357
-112.2564540158376,36.08395660588506,2357
-112.2580238976449,36.08511401044813,2357
-112.2595218489022,36.08584355239394,2357
-112.2608216347552,36.08612634548589,2357
-112.262073428656,36.08626019085147,2357
-112.2633204928495,36.08621519860091,2357
-112.2644963846444,36.08627897945274,2357
-112.2656969554589,36.08649599090644,2357 </coordinates>
      </LineString>
    </Placemark>
  </Document>
</kml>

```

Una vez generados, los archivos pueden ser abiertos por el programa Google Earth, lo que permite visualizar la trayectoria. Para abrir el archivo, ir a *File*→*Open* y elegir el archivo con formato KML que se desea visualizar. A continuación se brinda un ejemplo de visualización.



## Operación

El programa debe analizar los archivos que contienen los mensajes GPS, detectar los mensajes que contienen la información de localización geográfica y generar los archivos en los formatos de salida correspondientes.

Es importante tener en cuenta que dado que los datos de entrada se encuentran desordenados se deberán cargar en memoria en forma de arreglo para poder ordenarlos y reconstruir las trayectorias correctamente.

#### **4) Restricciones**

La realización de este programa está sujeta a las siguientes restricciones:

- Se debe recurrir al uso de T.D.A. para almacenar los elementos de información correspondientes al mensaje de GPS en formato GGA.
- Se debe recurrir al uso de punteros a función a fin de parametrizar la impresión de los archivos de salida.
- Se deben utilizar funciones y una adecuada modularización.
- Se debe construir un proyecto mediante la utilización de *makefile*.
- Hay otras cuestiones que no han sido especificadas intencionalmente en este requerimiento, para darle al desarrollador la libertad de elegir implementaciones que, según su criterio, resulten más favorables en determinadas situaciones. Por lo tanto, se debe explicitar cada una de las decisiones adoptadas y el o los fundamentos considerados para ellas.

#### **5) Entrega del Trabajo Práctico**

Se debe presentar la correspondiente documentación de desarrollo. Ésta se debe entregar en forma impresa y encapetada, de acuerdo con la numeración siguiente:

- 1) Carátula. Incluir una dirección de correo electrónico.
- 2) Enunciado (el presente documento).
- 3) Diagrama de flujo básico y simplificado del programa (1 carilla A4).
- 4) Estructura funcional del programa desarrollado (diagrama de funciones, 1 carilla A4).
- 5) Explicación de las alternativas consideradas y las estrategias adoptadas.
- 6) Resultados de la ejecución (corridas) del programa, en condiciones normales e inesperadas de entrada.
- 7) Reseña de los problemas encontrados en el desarrollo del programa y las soluciones implementadas para subsanarlos.
- 8) Conclusiones.
- 9) *Script* de compilación.
- 10) Código fuente.

#### **6) Bibliografía**

Se debe incluir la referencia a toda bibliografía consultada para la realización del trabajo: libros, artículos, etc.

#### **7) Fecha de entrega: 22 de junio de 2019.**