

# Trabajo Práctico de Laboratorio N° 2: Archivos HTML

**Docentes**

Mendez, Pablo Damián

**Alumnos**
**Legajo**

Mereles, Blanca Catalina

166958-8

Fecha			Aprobación
-------	--	--	------------

**Manual del usuario:** sistema operativo UNIX (*wget instalado*) ó Windows (*wget instalado*)

## Estrategia de Resolución

### Reporte A

La mayor parte del problema está resuelto en el punto b ,solo resta cambiar ciertas partes del código para conseguir este reporte..

En este caso debemos insertar una estructura condicional para solo escribir en nuestro archivo de salida las variaciones con valor negativo. De manera que solo cuando el substring del dato comenzará con un carácter '-' el dato sería escrito en el archivo de salida.

```

if (*substring == '-')
{
    fprintf(pl, "%s;", especie);
    while (*substring != '<')
    {
        fprintf(pl, "%c", *substring);
        substring++;
    }
    fprintf(pl, "\n");
}
}

```

### Reporte B

Para este caso se utiliza el código base provisto por el profesor de la cátedra. En el mismo se abre el archivo HTML, se lo guarda en una variable de tipo file, comenzará a leer el contenido por línea. Si la misma cumple con las características requeridas comenzará con la lectura de carácter por carácter.

```

if (html = fopen("SSL-TP2.html", "rt"))
{
    p2 = fopen("p2.csv", "w");
    fprintf(p2, "Especie;Precio de compra;Precio de venta;Apertura;Precio Mínimo;Precio Máximo\n");

    while (fgets(buffer, 2048, html)) // Lee una línea y guarda en buffer.
    {
        if ((pchar = strstr(buffer, "data-order")) && strstr(buffer, "Cdo.</td>")) |

```

Para este caso consistió en definir qué anclaje utilizar para splitear la línea, y obtener así los datos

requeridos, y las herramientas que se utilizara para splitear dicha línea. Utilizare el **strtok** y splitear la línea anclando en el '/', este carácter define el fin de una celda.

A partir de ahí debía identificar cuándo comenzar a leer el dato requerido con el substring obtenido con **strok**. Cada dos '<' el dato de las tablas estaba al alcance. De esta manera, simplemente con una estructura cíclica y contando los caracteres '<', se logra guardar los datos que cada reporte requería.

Una vez conseguido esto se recorre el string utilizando aritmética de punteros y se iba escribiendo en el archivo de salida los datos requeridos.

```
/*Escribamos el Vencimiento */  
  
substring = strtok(NULL, "/");  
while (i != 2){  
    if (*substring == '>')  
    {  
        i++;  
    }  
    substring++;  
}  
while (*substring != '<'){  
    substring++;  
}
```

## Reporte C

La mayor dificultad aquí aparece a la hora de tener que guardar los datos de compra y venta para poder ser luego comparados con el dato de la variación. Se utiliza la función **atof()**, la cual convierte un string en un dato de tipo double. La dificultad extra que suponía esta función es que necesitaba que el string presente como símbolo decimal el punto. Entonces, debí agregar al código un paso extra en el que se reemplaza en el buffer de tipo string la coma leída por un punto:

```
c = 0;  
while (*substring != '<'){  
    bventa[c] = *substring;  
    if (bventa[c] == ',') // Reemplazo coma por punto para poder convertir el string en float.  
    {  
        bventa[c] = '.';  
    }  
    c++;  
    substring++;  
}
```

Una vez guardadas las tres variables a comparar, creé una estructura condicional que marcaba lo siguiente:

- Solo escribía en el archivo de salida si la variación era negativa.
- Si la venta y la compra eran menores que la apertura la fila se escribía en verde.
- Si la venta y la compra no eran menores que la apertura se escribía sin colorear.

```

if (variacion < 0)
{
    if (venta < apertura && compra < apertura)
    {
        fprintf(P3, "<tr>\n");
        fprintf(P3, "<td> <span style=\"color:#10da10;\">%s </span></td>", especie);
        fprintf(P3, "<td> <span style=\"color:#10da10;\">%f </span></td>\n", variacion);
        fprintf(P3, "</tr>\n");
    }
    else
    {
        fprintf(P3, "<tr>\n");
        fprintf(P3, "<td> <span>%s </span></td>", especie);
        fprintf(P3, "<td> <span>%f </span></td>\n", variacion);
        fprintf(P3, "</tr>\n");
    }
}

```

```

}

```