

Universidad Nacional de Asunción - Facultad Politécnica

Examen Parcial de Programación

Fecha: 30 de octubre de 2021

Directivas y Recomendaciones

- Los problemas deben de ser resueltos usando el Lenguaje C y las herramientas vistas hasta la fecha del examen.
- Los comentarios aclaratorios pueden ayudar en la corrección del examen, influyendo favorablemente en su calificación.
- Utilice en su programa los nombres de las estructuras indicadas en el enunciado.
- La duración del examen es de 120 minutos.
- El código fuente debe subirse al aula virtual de la materia (en EDUCA).
- La interpretación del problema es parte de la evaluación del examen
- El ejercicio será evaluado por casos de pruebas establecidos por el Profesor.
- El profesor estará evaluando los ejercicios con alguna herramienta de comparación de códigos. Códigos iguales o muy semejantes no serán considerados para la corrección. Se tendrán en cuenta códigos encontrados en internet.

Tema 1: Formato de números (100%)

Se proporciona un código inicial para este problema. Escriba una función llamada `fmtMiles` que reciba como parámetros:

1. Un número entero `N` (`int N`).
2. Un número entero `A` (`int A`).
3. Un carácter `C` (`char C`).
4. Una cadena `sal` (`char sal[]`)

La función `fmtMiles` debe cargar en la cadena `sal` el mismo número `N`, pero formateado con los separadores de miles. Por ejemplo: si `N` es `1234567`, entonces la cadena que corresponde a este número formateado con los separadores de miles es `"1.234.567"`. Se define como *"ancho"* de `N` a la cantidad de caracteres necesarios para representarlo en este formato (sin contar el carácter de finalización de una cadena en C). Para nuestro ejemplo, el ancho correspondiente a `N` es de `9` (7 dígitos + 2 puntos).

Además, la función `fmtMiles` debe considerar los parámetros `A` y `C`. El valor de `A` indica la longitud de la cadena que contendrá `sal` (sin contar el carácter de finalización de una cadena en C). Si el valor de `A`, en comparación al *ancho* correspondiente a `N`, resulta:

- **MAYOR**, entonces se debe completar los espacios vacíos a la derecha (de la representación de `N`) con tantos caracteres `C` que sean necesarios para llenar esos espacios vacíos.
- **MENOR**, entonces se debe cargar en `sal` una cadena de longitud `A` que consista completamente de caracteres `C`.
- **IGUAL**, entonces se debe cargar en `sal` únicamente la representación de `N` en el formato indicado (ignorando `C`).

Estas indicaciones pueden ser visualizadas en los ejemplos presentados más abajo.

Observaciones importantes:

- Como se indica en el enunciado y en el código proporcionado, deberá definir una función `fmtMiles` que realice lo solicitado (no se aceptará que la funcionalidad se desarrolle en el `main`).
- Verificar (dentro de la función solicitada) que `N` sea un número positivo. Si no lo es, entonces `sal` deberá contener una cadena vacía.
- Verificar (dentro de la función solicitada) que `A` sea un número positivo. Si no lo es, entonces `sal` deberá contener una cadena vacía.
- Verificar (dentro de la función solicitada) que `C` no sea un carácter correspondiente a un dígito, o el carácter `-` o `+`. Si es alguno de ellos, entonces `sal` deberá contener una cadena vacía.
- El valor máximo de `N` es `2147483647`

Además, el programa deberá considerar la lectura de un archivo de entrada `numeros.txt` (que estará en la misma carpeta que el programa), que contendrá una lista de casos de prueba para la función `fmtMiles`. Cada línea del archivo tiene el siguiente formato:

`N(int) \t A(int) \t C(char)`

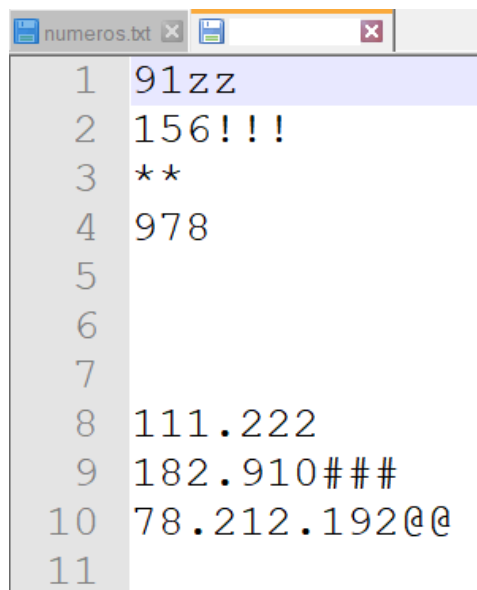
Se proporciona un ejemplo para el archivo `numeros.txt`.

Los resultados para estos casos de entrada (los números formateados) deben guardarse en otro archivo de salida (el nombre queda a cargo del estudiante).

Ejemplos:

N	A	C	En <code>sal</code> debe cargarse	Observación
91	4	z	91zz	Longitud mayor a lo necesario para N
156	6	!	156!!!	Longitud mayor a lo necesario para N
768	2	*	**	Longitud menor a lo necesario para N
978	3	?	978	Longitud igual a lo necesario para N
-5	10	#		N es negativo, por lo que la cadena <code>sal</code> debe estar vacía
18	0	=		A es cero, por lo que la cadena <code>sal</code> debe estar vacía
20	2	+		C es +, por lo que la cadena <code>sal</code> debe estar vacía
111222	7	*	111.222	Longitud igual a lo necesario para N
182910	10	#	182.910###	Longitud mayor a lo necesario para N
78212192	12	@	78.212.192@@	Longitud mayor a lo necesario para N

Archivo de salida para el ejemplo:



```

1 91zz
2 156!!!
3 **
4 978
5
6
7
8 111.222
9 182.910###
10 78.212.192@@
11

```

Valoración del Programa (Puntaje):

- Respetar el formato de las entradas y las salidas esperadas del programa (25 % del ejercicio)
- La función implementada es correcta (40% del ejercicio)
- La salida del programa es correcta (35% del ejercicio)