



## 1 - Trabalho

### Trabalho individual

Pontuação: 1,0 EXTRA na I Unidade

#### Descrição:

- Implementar na linguagem Assembly do MIPS uma pequena biblioteca *string* com funções de manipulação de caracteres. Por se tratar de função do assembly MIPS, é obrigatório o uso da convenção, ou seja, registradores do tipo *save* (\$s0, \$s1, ...) usados pela função precisam ser salvos na pilha assim como o \$ra caso a função chame outra. Também é obrigatório que as funções recebam seus parâmetros pelos registradores de argumento (\$a0, \$a1, ...) e caso retornem algum valor fazer isso usando os registradores de valor (\$v0, \$v1).
- Implemente todas as funções em um mesmo arquivo que deverá ser salvo como "string.asm". Para testar essas funções implemente um arquivo chamado "main.asm" que deverá incluir a biblioteca de funções usando a diretiva **.include "string.asm"**. O Main deverá imprimir um menu com as opções que o usuário poderá testar (numerar cada opção de 1 até o máximo de funções implementadas) e esperar que seja digitado uma das opções (não é necessário fazer teste de validade das opções). A opção 0 (zero) do menu deverá encerrar o programa. Cada opção escolhida pelo usuário deverá fazer a chamada à função correspondente e retornar ao menu principal em seguida.
- Códigos sem comentários não serão aceitos. Todos os arquivos deverão ter inicialmente um comentário com o nome do aluno e o que está implementado ali.
- Os arquivos devem ser "zipados" e "postados" no SIGAA até a data definida, não sendo aceito após esta data. O trabalho é opcional, não havendo dependência dele para obtenção do máximo da nota da respectiva unidade.
- As funções, com suas respectivas definições são listadas a seguir:

FUNÇÃO	DEFINIÇÃO
unsigned int strlen( char *s) ✓	Recebe um ponteiro que indica uma string e retorna quantos caracteres essa string possui.
int strcmp( char *s, char *t) ✓	Compara duas strings (do caracter mais a esquerda até o mais a direita) e retorna um inteiro. Se esse inteiro for negativo, é porque a primeira string é menor que a segunda. Se retornar um inteiro positivo, é porque a segunda string é maior que a segunda. Se retornar 0, é porque as strings são idênticas
char *strcat( char *dest, char *src) ✓	Copia (concatena) a string 'src' ao final da string 'dest'.
char *strncat( char *dest, char *src, int count) ✓	Copia (concatena) 'count' elementos da string 'src' no final da string 'dest'
char *strncpy( char *dest, char *src, int count); ✓	Faz a mesma coisa da strcpy, mas em vez de copiar todos os elementos, copia somente os 'count' primeiros elementos da 'src' para a 'dest'.