

Instruções:

1. Esta atividade pode ser realizada individualmente ou em grupo de **até TRÊS** alunos (no máximo).
2. A atividade consiste na implementação de um procedimento utilizando a linguagem de montagem do RISC-V (conforme as instruções a seguir), o qual é apresentado logo a seguir.
3. Deve ser postado, no material didático, um relatório, com uma capa identificando a Instituição, o curso, a disciplina, o professor, o nome da atividade, os autores do trabalho e data em que o mesmo for entregue. O corpo do relatório deverá conter a resolução dos exercícios, incluindo: **código-fonte** em linguagem de montagem e **capturas de tela** demonstrando a execução correta das entradas e saídas realizadas via **console do simulador**.
4. O código-fonte deve ser escrito em arquivo com extensão .asm e, **obrigatoriamente**, com um nome que identifique os membros do grupo (ex. ***Astarion_ShadowHeart.asm***). Obs: Todos os arquivos de todos os grupos serão reunidos em uma mesma pasta e, por isso, não use nomes como Programa_01.asm.
5. Cada código fonte deve conter um cabeçalho comentado que identifique a disciplina, a atividade, o programa e os nomes dos membros do grupo. Ex:

```
# Disciplina: Arquitetura e Organização de Computadores  
# Atividade: Avaliação 03 - Programação de Procedimentos  
# Grupo: - Astarion  
#         - Shadow Heart
```
6. O arquivo ASM e o relatório em formato PDF deverão ser postados no ambiente Material Didático compactados em um único arquivo em formato ZIP, conforme instruções fornecidas em aula.
7. O prazo para entrega do relatório e postagem dos códigos fonte é 18:59 do dia **13/11/2023**.
8. Trabalhos em atraso terão a nota reduzida proporcionalmente a quantidade de dias de atraso.
9. O professor poderá solicitar a qualquer momento que **qualquer aluno** do grupo faça uma **demonstração explicativa da execução dos códigos no RARS**.
10. **A implementação deverá apresentar resultados corretos para qualquer conjunto de dados**. Uma solução que **não execute corretamente** terá, automaticamente, um **desconto de 50% na nota**, sendo que o professor também avaliará a correção de segmentos específicos do código (controle de execução, acesso a memória,...).
11. Se forem identificados **trabalhos** com grau de **similaridade** que caracterize cópia (autorizada ou não) ou adaptação, a nota dos grupos será a **nota de um trabalho dividida** pelo número de grupos que entregou esses trabalhos similares.

ESPECIFICAÇÃO DO PROBLEMA 1

Enunciado:

Utilizando a linguagem de montagem do RISC-V, implemente um procedimento que determine a soma dos elementos de um vetor de zero até a posição passada por parâmetro.

Requisitos:

1. Crie um vetor de 100 posições no `.DATA` e preencha ele com valores distintos. Exemplo: vetor: `.word 0,1,2,3,4,5,6,...`
2. A função principal (MAIN) deve carregar em registradores de argumento o endereço base do vetor e o número de posições a ser somada.
 - a. Verifique se o número de posições deverá estar entre 2 e 100;
3. No retorno do procedimento, a função principal deve imprimir uma mensagem informando o valor da soma de todas as posições do vetor (de zero até a posição passada por argumento) calculado pelo procedimento.
4. O procedimento deve utilizar pelo menos um registrador `s`. (para estimular o uso da pilha)
5. A função principal e o procedimento devem respeitar as regras relativas ao uso dos registradores do RISC-V, preservando na pilha os registradores de acordo com o especificado nessas regras (ex. a função principal deve preservar os registradores `T` que necessita que sejam preservados e o procedimento deve preservar os registradores `s` que utilizar). Verifique as regras descritas no livro texto.
6. O código deve ser escrito respeitando o estilo de programação ASM, usando tabulação para organizar o código em colunas (rótulos, mnemônicos, operandos e comentários).
7. Procure comentar ao máximo o seu código. Isto é um hábito da programação *assembly*.
8. No Apêndice A do livro estão listadas todas as instruções e pseudo-instruções do RISC-V, incluindo diferentes instruções úteis para comparação de valores e desvio condicional.
9. No seu relatório, apresente uma análise indicando quais instruções do conjunto de instruções do RISC-V foram utilizadas.

NOTA:

Abaixo, segue uma referência de código C de um procedimento que implementa o algoritmo soma vetores:

```
int vet_soma(int vet[TAM], int pos) {
    int soma=0;
    for (int i=0; i<pos; i++) {
        soma = soma + vet[i];
    }
    return soma;
}
```

ESPECIFICAÇÃO DO PROBLEMA 2

Enunciado:

Utilizando a linguagem de montagem do RISC-V, implemente um procedimento **recursivo** que determine a soma dos elementos de um vetor de zero até a posição passada por parâmetro.

Requisitos:

1. Siga os mesmos passos do problema 1,

NOTA:

Abaixo, segue uma referência de código C de um procedimento que implementa o algoritmo soma vetores:

```
int rec_vet_soma (int vet[TAM], int pos) {  
    if (pos < 0) {  
        return 0;  
    }  
    return vet[pos] + soma (vet, pos-1);  
}
```

ANÁLISE DOS RESULTADOS

Enunciado:

A parte mais importante desse trabalho é a análise comparativa dos resultados obtidos nos 2 problemas. Para isso procure coletar estatísticas (*instruction statistics*) das duas soluções com diferentes tamanhos de vetor.

Recomenda-se o uso de gráficos para auxiliar na demonstração dos resultados.

Tente responder perguntas como qual solução obtém o melhor desempenho, como os dados pioram ou melhoram as soluções e porque isso acontece.

CÁLCULO DA NOTA

Nota = (25*relatório + 25*problema1 + 25*problema2 + 25*análise)/100