

USP – ICMC – SSC

SSC0902 – Organização e Arquitetura de Computadores (OrgArq)

Professor responsável: *Fernando Santos Osório*

Semestre: 2021/2

Monitor: Marcos Vinícius Pietrucci

Horário: Seg. 10h10 e Quinta 08h10

Wiki: SSC-902 e **Facebook:** SSC0902-FOsorio

Web: <http://www.icmc.usp.br/~fosorio/>

NRO. USP: < Colocar o NUSP nos programas fonte>

DATA: 06 / 12 / 2021

NOME: < Colocar o NOME nos programas fonte>

Prova Final 06.12

>> COLOCAR SEU NOME E NRO. USP COMO COMENTÁRIO DOS PROGRAMAS ENTREGUES!

***PROVA P2 – FINAL – SSC0902 OrgArq (Programação)
Programação MIPS com o MARS ou SPIM***

Implemente o programa em Linguagem de Montagem do Processador RISC **MIPS 32 bits** (.S ou .ASM ou .TXT – Arquivo texto com programa), usando o montador e simulador apresentado nas aulas, o **MARS ou SPIM**, usados em nossas aulas e para fazer os exercícios durante o semestre, seguindo exatamente a especificação e instruções dadas abaixo => Entregar 1 programa completo quem implementa o “adivinha o número”.

PROGRAMA MIPS) Faça um programa para o RISC MIPS 32bits no MARS ou SPIM (montadores e simuladores) que desafie uma pessoa a adivinhar um número de até 3 dígitos antes e 3 dígitos após a vírgula, ou seja, um valor entre 000.000 e 999.999. O usuário do programa poderá tentar adivinhar o número com múltiplas tentativas, onde o programa a cada tentativa vai indicar se o número informado é maior, menor, ou, se o usuário acertou o “número surpresa”. Vamos nos referir ao número que o usuário deve adivinhar como número oculto ou número surpresa, e ao número informado pelo usuário como número tentativa. Quando o usuário informar um “número tentativa” que for igual ao número oculto, e portanto, “adivinhar o número”, o programa encerra, mostrando uma mensagem de SUCESSO e uma contagem da quantidade de vezes que o usuário informou um número tentativa (quantidade de tentativas até acertar). Atenção o programa deve ter uma SUB-ROTINA, conforme descrito mais abaixo.

Exemplo de Interação: Supondo que o número surpresa seria 100.000 - Versão FLOAT e Versão INT

Adivinhe um numero entre 0.0 e 999.999 (com 3 casas antes e depois da virgula)

Digite um numero: 10.000

O numero oculto eh MAIOR!

Digite um numero: 50.000

O numero oculto eh MAIOR!

Digite um numero: 200.000

O numero oculto eh MENOR!

Digite um numero: 100.000

Voce ACERTOU! Parabens!

Numero Oculto: 100.000

Numero Tentativa: 100.000

Total de Tentativas: 4

Adivinhe um numero entre 0.0 e 999.999 (com 3 casas antes e depois da virgula)

Digite a parte inteira do numero: 10

A parte inteira do numero oculto eh MAIOR!

Digite a parte decimal do numero: 10

A parte decimal do numero oculto eh MENOR!

Digite a parte inteira do numero: 50

A parte inteira do numero oculto eh MAIOR!

Digite a parte decimal do numero: 5

A parte decimal do numero oculto eh MENOR!

Digite a parte inteira do numero: 200

A parte inteira do numero oculto eh MAIOR!

Digite a parte decimal do numero: 0

A parte decimal do numero oculto esta CERTA!

Digite a parte inteira do numero: 100

A parte inteira do numero oculto eh CERTA!

Digite a parte decimal do numero: 0

A parte decimal do numero oculto esta CERTA!

Voce ACERTOU! Parabens!

Numero Oculto: 100.000

Numero Tentativa: 100.000

Total de Tentativas: 4

>> **VOCÊ pode optar em implementar o programa usando FLOATs ou INTEIROS**

INFORMAÇÕES IMPORTANTES sobre a IMPLEMENTAÇÃO DA PROVA:

- **SUB-ROTINA que deve ser IMPLEMENTADA OBRIGATORIAMENTE**

Rotina: TestaNumero

Entradas (Parâmetros de Entrada):

1. Valor digitado, o “número tentativa” (se for inteiro são 2 números)
2. Valor esperado, o “número oculto” ou “número surpresa” (se for inteiro são 2 números)

Saída:

1. Exibir na tela a mensagem (Maior, Menor ou Acertou)
Conforme exemplo acima: “O numero oculto eh MAIOR!”, “Voce ACERTOU! Parabens!”,
ou as mensagens de acordo com os demais exemplos dados. Só exibir as mensagens do tipo:
“maior”, “menor”, “acertou”
2. Retorno da subrotina: valor 1 se acertou, ou, valor 0 se não acertou.

Na “linguagem C” seria algo assim:

int TestaNumero (nrotentativa, nrooculto); // Sendo o retorno 0 ou 1.

>> Esta rotina deve ser usada cada vez que o usuário tentar adivinhar um novo número.

>> Esta rotina deve ser implementada de acordo com as convenções de uso dos registradores do MIPS e de chamadas de sub-rotinas.

- Não esqueça de implementar um contador de tentativas!

- Definição do Número Surpresa (Número Oculto)

Você deve definir no código uma variável (endereço contendo um valor) e indicar o valor Surpresa/Oculto

A variável deve ser chamada de NOCULTO => Nome da Variável em memória com nro. oculto.

Inicialize esta variável com o número que você deseja que o usuário deve acertar.

Exemplo:

NOCULTO 100.000 (float) ou

NOCULTOI 100 (parte inteira) e NOCULTOD 0 (parte decimal, após a vírgula)

- **OPCIONAL:** Para quem for fazer usando o MARS, pode usar a syscall que gera números aleatório, mas atenção, é importante gerar um número que tenha apenas 3 casas antes e 3 casas após a vírgula, sendo consideradas no programa de adivinhação. Pode gerar 2 inteiros limitando os valores (3 casas), ou, pode gerar 2 inteiros (3 casas) e depois compor eles em um float, ou, pode gerar um float e depois na comparação “desconsiderar” as casas abaixo da 3 casa decimal.

Detalhes: 2 inteiros => 123 e 456

2 inteiros que viram um float => 123 => 123.0 ; 456 => 456.0 => 0.456 → 123.456

1 float digitado => 123.456

1 float aleatório gerado => 0.123456789 multiplica por 1000 → 123.456789

- **CUIDADO COM VALORES FLOAT:** Sempre considere que pode haver uma imprecisão...
=> Digitado 100.0 MAS pode ser que fique armazenado 100.00001
=> Valor Oculto: 100.000
Como resolver? Digitado MENOS Oculto = 0.00001 que é menor que 0.001 (menor erro aceito)
Se você subtrair os valores, o valor restante deve ser menor que 0.001 ! (esse é o truque!)
- USE SYSCALLs para ler valores do teclado e exibir valores na tela (Console).
Use o devido syscall conforme seu programa precisa de um FLOAT ou de um INTEIRO.

1. Para quem for implementar usando FLOAT
Atenção com problemas de precisão!
2. Para quem for implementar usando INTEGER
Atenção, somente se acertar as 2 partes (inteira e decimal) é que o usuário acertou!

FIM!

INFORMAÇÕES IMPORTANTES :

REGRAS EM RELAÇÃO REALIZAÇÃO DESTA PROVA

- A PROVA É INDIVIDUAL. Não pode consultar PESSOAS.
- PODE CONSULTAR O MATERIAL DE AULAS e MATERIAIS DA INTERNET (não pessoas).
- PROVA DEVE SER REALIZADA CONSIDERANDO AS REGRAS DO “Contrato” QUE VOCÊ ACEITOU => <https://forms.gle/a1jNJ8Rt4tFn74116>
(É necessário aceitar os termos do contrato para que a prova seja corrigida!)
- ENVIO DA PROVA POR E-MAIL PARA: **fosorio@icmc.usp.br** com cópia (Cc:) para **fosorio@gmail.com**
SUBJECT/ASSUNTO: **SSC0902 P2 Final - <colocar aqui o seu nome>**
Enviar anexado apenas o FONTE do PROGRAMA da resposta da prova (.asm .s ou .txt) não envie binários!

BOA PROVA!