



## Problema #2 – 2023.2

### Linguagem Assembly

#### 1. Tema

Desenvolvimento de programas usando linguagem Assembly e aplicação de conceitos básicos de arquitetura de computadores.

#### 2. Objetivos de Aprendizagem

Ao final da realização deste problema, o/a discente deverá ser capaz de:

- Programar em Assembly para um processador com arquitetura ARM;
- Entender o conjunto de instruções da arquitetura ARM e saber como utilizá-las de acordo com a necessidade do sistema;
- Avaliar o desempenho de um código assembly através de medidas sobre o comportamento de sua execução no sistema.

#### 3. Problema

Desenvolver uma IHM (Interface Homem-Máquina) que apresente em um display LCD as informações do sensor desenvolvido. A interface deve substituir a que foi desenvolvida em linguagem C, atendendo aos mesmos requisitos. O protótipo dessa interface será embutido em um computador de placa única (SBC).

#### 4. Requisitos

O problema a ser desenvolvido no SBC Orange Pi deve atender às seguintes restrições:

- 4.1. O código deve ser escrito em Assembly;
- 4.2. O sistema só poderá utilizar os componentes disponíveis no protótipo.

#### 5. Produto

Todo o projeto deverá ser disponibilizado na plataforma GitHub. No prazo indicado no cronograma a seguir, cada equipe deverá apresentar:

- 5.1. Levantamento de requisitos;
- 5.2. Código
  - 5.2.1. Código em linguagem Assembly;
  - 5.2.2. Todos os códigos deverão estar detalhadamente comentados;
- 5.3. Script de compilação tipo Makefile para geração do código executável;

- 5.4. Documentação técnica escrita no arquivo README do projeto no GitHub, contendo, no mínimo:
  - 5.4.1. Detalhamento dos software usados no trabalho, incluindo softwares básicos;
  - 5.4.2. Arquitetura do computador usado nos testes;
  - 5.4.3. Descrição dos tipos de instruções utilizadas;
  - 5.4.4. Descrição de instalação, configuração de ambiente e execução;
- 5.5. Descrição dos testes de funcionamento do sistema, bem como, análise dos resultados alcançados.

## 6. Cronograma

Semana	Data	Descrição
8	ter. - 19/set.	<b>Paralisação - Sintest</b>
	qui. - 21/set.	Problema 2 – Apresentação
9	ter. - 26/set.	<b>Problema 1 – Entrega/Avaliação</b>
	qui. - 28/set.	Problema 2 – Seção Tutorial #2
10	ter. - 03/out.	Problema 2 – Seção Tutorial #3
	qui. - 05/out.	Problema 2 – Seção Desenvolvimento #1
11	ter. - 10/out.	Problema 2 – Seção Tutorial #4
	qui. - 12/out.	<b>Feriado – Nossa Senhora Aparecida</b>
12	ter. - 17/out.	<b>Feira de Graduação</b>
	qui. - 19/out.	Problema 2 – Seção Desenvolvimento #2
13	ter. - 24/out.	Problema 2 – Seção Tutorial #5
	qui. - 26/out.	<b>Problema 2 – Entrega/Avaliação</b>

## 7. Avaliação

Para avaliar o envolvimento do grupo nas discussões e na apresentação, o tutor poderá fazer perguntas variadas a qualquer membro, tanto nas sessões tutoriais quanto na apresentação. O estudante que não comparecer, ou se atrasar, no dia da sessão de apresentação, terá automaticamente nota 0,0 (zero) no problema, excetuando-se as condições que permitem 2ª chamada de avaliações, conforme regulamento do curso.

A nota final será a composição de 3 (três) notas parciais:

Critério	Critérios para a nota	Peso
Desempenho Individual	Participação individual nas sessões tutoriais, de acordo com o interesse e entendimento demonstrados pelo aluno, assim como sua assiduidade, pontualidade e contribuição nas discussões. Essa nota inclui o desempenho do estudante na apresentação do problema no laboratório.	3

Documentação	Documentação técnica de cada grupo, considerando qualidade da redação (ortografia e gramática), organização dos tópicos, definição do problema, descrição da solução, explicação dos experimentos, análise dos resultados, detalhando os itens não atendidos, se for o caso.	3
Códigos	Qualidade do código fonte (organização e comentários), e execução correta dos códigos binários de acordo com testes de validação que explorem as situações de uso.	4