

Software Básico - Respostas

Nico I. G. Ramos

GRR20210574

Novembro 2023

- A. Se você pudesse voltar no tempo, o que você (de hoje) recomendaria a você do primeiro dia de aula (de Software Básico) para minimizar o sofrimento do desenvolvimento deste trabalho?

Não encontrei muitas dificuldades para realizar o trabalho, eu gosto de trabalhar com *Assembly* e debugar no *gdb*. Saber usar o *gdb* e pensar no código como uma máquina de estados facilitou bastante o desenvolvimento do código. Uma recomendação que faria para o meu eu do começo do semestre seria para não começar o trabalho exausto e sem tomar o remédio para o TDAH. O que contribuiu para as duas principais dificuldades que tive: dificuldade com os símbolos e com a ligação.

A primeira ocorreu pois, por ter declarado o *original_brk* no *.h* logo que criei o arquivo, os valores não eram alterados corretamente. Eventualmente apaguei essa informação da minha memória, mas após examinar a execução no *gdb* e perceber que havia dois labels com o mesmo nome foi fácil intuir que provavelmente estava usando o mesmo label em duas partes distintas do trabalho. Após perceber isso, foi só uma questão de olhar os arquivos e procurar pela declaração errada.

Após remover o *original_brk* do *.h*, tive problema ao efetuar a ligação pois o ligador não encontrava o símbolo. Embora na hora não tenha entendido porque o símbolo na *BSS* não estava sendo exportado, foi intuitivo adicionar o *_global* e verificar se isso fazia com que o símbolo fosse exportado.

A dificuldade com a ligação se misturou com a dupla declaração do *original_brk* e a não exportação do símbolo na *BSS*. Durante o desenvolvimento fiquei confuso se os erros eram porque as flags (ou a falta delas) *-FPIE* e *-FPIC* estavam fazendo com que a ligação falhasse e se estavam contribuindo para que o valor do *original_brk* não fosse alterado corretamente. Após ler a documentação e pesquisar em fóruns, consegui fazer com que a ligação ocorresse e pude examinar o erro que estava dando com o *original_brk*.

- B. O que você recomendaria ao professor da disciplina quando ele se preparar para o próximo semestre remoto a fim de aumentar o grau de absorção do conteúdo da disciplina por parte dos alunos?

Acho que alguns estudos dirigidos ajudariam bastante na segunda parte da disciplina.

- C. Explique quais os trechos de código e quais as principais alterações que você fez para que a segunda parte funcionasse, ou indique o motivo de você não ter conseguido terminar a alteração. Indique, por exemplo, quais as linhas de código que você mudou e com qual objetivo.

Para alterar a estratégia de alocação, foram feitas duas principais mudanças na lógica:

1. **First fit:** percorre a *heap* em busca de um bloco que possa utilizar, caso o encontre, tenta fazer a divisão imediatamente e retorna. Caso não encontre, irá chegar no final da *heap* e um novo bloco será criado.

2. **Worst fit:** percorre toda a heap, a cada bloco, verifica se pode ser utilizado e se é maior do que o maior bloco encontrado, caso seja, atualiza os registradores do maior tamanho e do endereço do maior bloco. Quando alcançar o final da *heap*, caso um bloco tenha sido encontrado tenta fazer a divisão e retorna, caso contrário, aloca um novo bloco e retorna.

O que cada registrador representa também foi alterado entre as duas implementações:

Registrador	First Fit	Worst Fit
%rax	————	<i>Brk</i> atual
%r12	Tamanho requisitado	Maior tamanho encontrado
%r13	<i>Brk</i> atual	Endereço de início do maior bloco
%r14	Endereço atual	Endereço atual
%r15	Tamanho do bloco atual	Tamanho requisitado

No *first fit* o valor do *brk* não precisaria ser salvo, mas como o *%rax* não é preservado, por garantia e para não ter que se preocupar em ficar verificando se o valor não seria utilizado após alguma *syscall*, logo que comecei a implementar o código escolhi salvar o valor dele. No *worst fit*, não pude salvar o valor dele pois não queria utilizar a *stack* para salvar as variáveis locais e, para manter a consistência com o restante do código, precisei liberar um registrador salvo para armazenar o endereço do maior bloco encontrado. Como quando a única *syscall* é feita o valor do *brk* atual já foi utilizado, não é um problema não ter o valor do *%rax* salvo.