

Universidade Presbiteriana Mackenzie
Ciência da Computação
Sistemas Operacionais – Turma 04P11
Alan Meniuk Gleizer – 10416804
Caio Vinicius Corsini Filho – 10342005

Relatório Lab 06 – Uso do Malloc

código desenvolvido

```
#include <stdio.h>
#include <stdlib.h>

// 0 struct da celula (node)
struct reg {
    int conteudo;
    struct reg *prox;
};
typedef struct reg celula;

//----- 2 -----//
// Funcao que imprime toda a linked list do head ate o tail
void imprimirValores(celula *cabeca){
    celula *atual = cabeca;
    while(atual != NULL){
        printf("%d ", atual->conteudo);
        atual = atual->prox;
    }
    printf("\n");
}

//----- 4 -----//
// Funcao que remove todos os nos da linked list do head ate o tail
void removerCelulas(celula *cabeca){
    celula *atual = cabeca;
    while(atual != NULL){
        celula *aux = atual->prox;
        free(atual);
        atual = aux;
    }
}

void exercicio6() {
    // Ler a informacao de memoria do sistema (em /proc/meminfo)
    FILE *meminfo = fopen("/proc/meminfo", "r");
```

```

if (meminfo == NULL) {
    perror("Erro ao abrir /proc/meminfo");
    return;
}

long memTotalKB = 0;
char buffer[256];

// ler o arquivo ate achar a linha MemTotal, onde temos a informacao
while (fgets(buffer, sizeof(buffer), meminfo)) {
    if (sscanf(buffer, "MemTotal: %ld kB", &memTotalKB) == 1) {
        break;
    }
}

// Fecha o arquivo
fclose(meminfo);

// Converte kilobytes para bytes
long memTotalBytes = memTotalKB * 1024;

// Imprime o total de RAM em bytes
printf("Total de RAM: %ld bytes\n", memTotalBytes);

// Calculo da qtd de celulas que caberiam
long qtdCeculasMem = memTotalBytes / sizeof(celula);
printf("Considerando uma célula de %ld bytes e RAM de %ld bytes, caberiam %ld células na memória.\n", sizeof(celula), memTotalBytes, qtdCeculasMem);

return;
}

void main(){

    //----- 1 -----//
    // Criando os tres nodes para simular manualmente
    celula *cabeca = (celula*) malloc(sizeof(celula));
    celula *meio = (celula*) malloc(sizeof(celula));
    celula *rabo = (celula*) malloc(sizeof(celula));

    cabeca->conteudo = 10;
    cabeca->prox = meio;

    meio->conteudo = 14;
    meio->prox = rabo;

```

```

    rabo->conteudo = 21;
    rabo->prox = NULL;

    //----- 2 -----//
    imprimirValores(cabeca);

    //----- 3 -----//
    int quantMemoGasta = sizeof(celula);
    printf("A quantidade de memoria gasta por intancia da celula eh: %d.\n", quantMemo-
Gasta);

    //----- 4 -----//
    removerCelulas(cabeca);

    //----- 6 -----//
    exercicio6();
}

```

print da execução

```

agleizer@linuxzinho: ~/Documents/OSlabs_github_repo/OSlabs/lab06
agleizer@linuxzinho:~/Documents/OSlabs_github_repo/OSlabs/lab06$ gcc link*.c -o linkedList
agleizer@linuxzinho:~/Documents/OSlabs_github_repo/OSlabs/lab06$ ./linkedList
10 14 21
A quantidade de memoria gasta por intancia da celula eh: 16.
Total de RAM: 66520408064 bytes
Considerando uma célula de 16 bytes e RAM de 66520408064 bytes, caberiam 4157525504 células na memória.
agleizer@linuxzinho:~/Documents/OSlabs_github_repo/OSlabs/lab06$ |

```