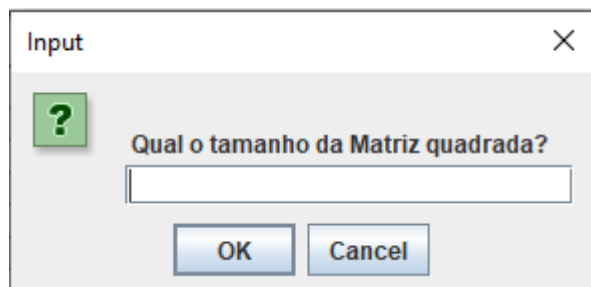
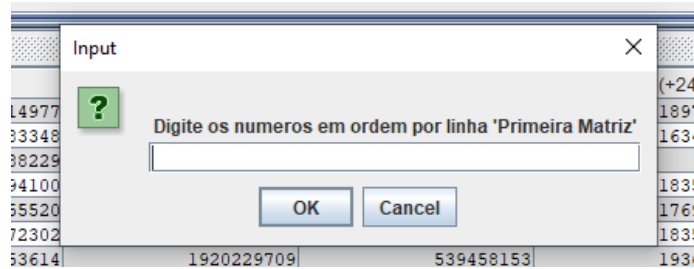


Lab 7

Nessa atividade eu realizei um multiplicador de matriz quadrada genérica de tamanho N, a alocação das matrizes foi feita de maneira dinâmica dentro da memória, usando o cálculo do número de elementos dentro de cada matriz para alocar posteriormente cada uma delas

```
16
17 partePrincipalAdquireInput:
18     li $v0, 51
19     la $a0, promptQuestaoTamanho
20     syscall
21     bnez $a1, inputInvalido
22     bgt $a0, 32, inputInvalido
23     move $t0, $a0
24     move $s0, $t0 #Tamanho da matriz
25     mul $s1, $t0, 4 #Tamanho da linha
26     mul $s2, $t0, $t0
27     mul $s2, $s2, 4 #Matriz na memoria
28     sw $t0, tamanhoMatriz
29     la $s3, matrizA #matrizA
30     add $s4, $s3, $s2 #MatrizB
31     add $s5, $s4, $s2 #MatrizResultado
32     mul $t0, $t0, $t0 #DimensÃo da matriz
33     li $t1, 0
34     move $t3, $s3
35     jal obtenMatrizUm
36     jal Cancel
37 inputInvalido:
```





Após isso os inputs da matriz passam pelo algoritmo de multiplicação e escrita na memória, após isso eles são traduzidos para ascii e escritos no resultado.

```
calculaMain:
    beq $t3, $s0, Cancel
    beq $t2, $s0, ResetaReg2 #Loop2
    beq $t1, $s0, ResetaReg #Loop1 - i < numero de columnas
    lw $t4, 0($s6) #Carrega o endereço da variavel da matriz A
    lw $t5, 0($s7) #Carrega o endereço da variavel da matriz B
    mul $t9, $t4, $t5 #Multiplica as variaveis
    addu $t8, $t8, $t9 #Soma as variaveis
    addi $s6, $s6, 4 #Add Linha
    add $s7, $s7, $s1 #Add coluna numero de elementos * 4
    addi $t1, $t1, 1
    j calculaMain

ResetaReg:
    sw $t8, 0($s5) # Guarda Resultado
    li $t8, 0 #Reseta somatorio
    addi $s5, $s5, 4 #Move Ponteiro, #Fim da matriz
    li $t1, 0 #Reseta contador
    addi $t2, $t2, 1 #Soma Contador de iteração do loop secundario
    addi $t7, $t7, 4 #Pecorre para a proxima linha
    move $s7, $s4
    move $s6, $s3
    add $s7, $s7, $t7
    add $s6, $s6, $t6
    j calculaMain

ResetaReg2:
    beq $t3, $s0, Cancel
    li $t7, 0
    li $t2, 0
    li $t1, 0
    add $t6, $t6, $s1
    move $s7, $s4
    move $s6, $s3
```

```
124      move $t1, $v0
125      LoopDeEscritaDeLinha:
126          beq $t9, $s0, EscreveNewLine
127          addi $t9, $t9, 1
128          move $t5, $zero
129          move $t6, $zero
130          move $t7, $zero
131          lw $s7, space
132          lw $t2, ($t3)
133          addi $t3, $t3, 4
134      Milhar:
135          blt $t2, 1000, Centena
136          addi $t5, $t5, 1
137          subi $t2, $t2, 1000
138          j Milhar
139      Centena:
140          blt $t2, 100, Dezena
141          addi $t6, $t6, 1
142          subi $t2, $t2, 100
143          j Centena
144      Dezena:
145          blt $t2, 10, EscreveNaMemoria
146          addi $t7, $t7, 1
147          subi $t2, $t2, 10
148          j Dezena
149      EscreveNaMemoria:
150          addi $t5, $t5, 40
151          sw $t5, 0($s5)
152          addi $s5, $s5, 4
153          addi $t6, $t6, 40
154          sw $t6, 0($s5)
155          addi $s5, $s5, 4
156          addi $t7, $t7, 40
157          sw $t7, 0($s5)
158          addi $s5, $s5, 4
```

Por último as matriz na memória, pode se observar circulado em vermelho a matriz resultado!

268501216	1949200228	29816	10	11	12	13	14	15
268501248	16	17	18	19	20	21	22	23
268501280	24	25	26	27	732	765	798	830
268501312	972	1014	1128	1179	1230	0	48	55