

Nº Mec.: _____ Nome: _____

NOTE BEM: Leia atentamente todas as questões, comente o código usando a linguagem C e respeite a convenção de passagem de parâmetros e salvaguarda de registos que estudou. Na tradução para o *Assembly* do MIPS respeite rigorosamente os aspetos estruturais e a sequência de instruções indicadas no código original fornecido.

O código em C apresentado pode não estar funcionalmente correcto, pelo que **não deve ser interpretado**.

Este teste é constituído por 4 folhas.

1) Analise o programa *Assembly* seguinte e responda às questões que se seguem:

```

      .data                                # 0x10010000
X1:   .asciiz "TSTEX2-2021" #
X2:   .space 26                        #
X3:   .byte  0xFF                      #

      .text                                # 0x00400000
      .globl main                        #
main:  ori    $t0, $0, 0x39              #
      la     $t4, X1                     #
      la     $t5, X2                     #
      move   $t6, $t5                   #
L1:    bge    $t4, $t5, L3                #
      lb     $a0, 0($t4)                  #
      bgt    $a0, $t0, L2                #
      sb     $a0, 0($t6)                  #
      addiu  $t6, $t6, 1                  #
L2:    addiu  $t4, $t4, 1                  #
      j      L1                          #
L3:    lw     $v0, 0($t5)                  #
      jr     $ra                          #
    
```

Carater	Cod. ASCII
'!'	0x21
'#'	0x23
'\$'	0x24
'%'	0x25
'*'	0x2A
'+'	0x2B
'-'	0x2D
'0'	0x30
'@'	0x40
'A'	0x41
'Z'	0x5A
'a'	0x61
'z'	0x7A

- Qual o número total de posições de memória ocupado pela *string* "X1"?
- Qual o endereço de memória a que corresponde o *label* "X2"?
- Se "X2" for o endereço inicial de um *array* de "floats", qual a dimensão máxima desse *array*?
- Se "X2" for o endereço inicial de um *array* de "floats", qual o endereço de memória da posição X2[4] desse *array*?
- Qual o número total de bytes de memória usado pelo segmento de dados do programa?
- Considerando que a primeira instrução do trecho de código fornecido está armazenada a partir do endereço 0x00400000, quais os endereços a que correspondem os *labels* "L1" e "L2"? (tenha em atenção as instruções virtuais do programa).
L1: _____ L2: _____
- Quantas vezes é realizado, de forma completa, o ciclo de programa?
- Qual o valor do registo \$t5 no fim do programa?
\$t5: _____
- Qual o valor do registo \$v0 no fim do programa?
\$v0: _____

```
int fun1(double *a, int N, double *b)
{
    int k = 0;
    double *p;

    for( p = a; p < (a + N); p++)
    {
        if( (*p / 2.0) != 0.0 )
        {
            *b++ = *p;
        }
        else
            k++;
    }
    return (N - k);
}
```

Variável	Registo
a	
N	
b	
k	
p	

[illegible]

```
int funk( int, int );

int fun2(int *p, int k)
{
    int n=0;
    while ( *p != k )
    {
        n = n + funk(*p, k);
        p++;
    }
    return n;
}
```

Variável	Registo
p	
k	
n	

[illegible]

a) Preencha a tabela seguinte:

Name	Size	Align	Offset
name	50	1	0
num	4	4	50 -> 52
grade	4	4	56
type	1	1	60
student	64	4	

```
typedef struct
{
    char name[50];
    int num;
    float grade;
    char type;
} student;
```

```
float fun3(student *std, int n)
```

```
{
    int i;
    float sum=0.0;

    for(i=0; i < n; i++)
    {
        print_string(std[i].name);
        print_float(std[i].grade);
        sum += std[i].grade;
    }

    return sum / 2.0;
}
```

Variável	Registro
std	\$a0
n	\$a1
i	\$t0
sum	\$f2

```
int main(void)
```

```
{
    static student std[2] = {{ "Rei Eusebio", 12345, 17.2, 'F' },
                              { "Rainha Amalia", 23450, 12.5, 'C' } };

    print_float(fun3(std, 2));
    return -1;
}
```

[illegible]