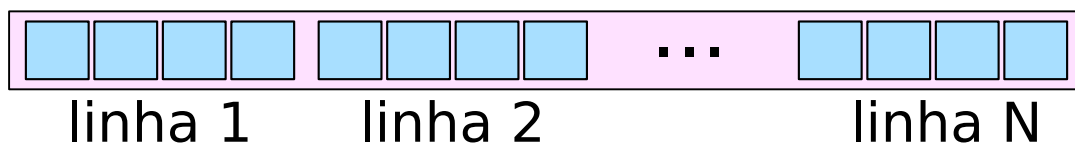


DCA0201 – Paradigmas de programação (1ª avaliação - 2016.1)

Aluno: _____ Matrícula: _____

1. Uma determinada aplicação demanda o uso de matrizes de inteiros de 4 bytes. Tais matrizes possuem uma quantidade de linhas variável, mas com uma quantidade de colunas fixa e igual a 4. O programador decidiu armazenar a matriz em um único array, com as linhas representadas em sequência, como ilustra a figura seguinte.



Nessa aplicação, o processo de cópia de matrizes é intensivo, de sorte que resolveu criar uma pequena aplicação para decidir pela melhor forma de realizar essa cópia. Essa aplicação é mostrada na listagem que segue.

```
#include <stdio.h>
#include <time.h>
#include <math.h>
#include <stdlib.h>
#include <string.h>

void copyA(int *dst, int *src, int size){
    int i, j;
    for(i=0; i<size; i++){
        for(j=i*4; j<i*4+4; j++){
            dst[j]=src[j];
        }
    }
}

void copy4(int *dst, int *src){
    memcpy(dst,src,4);
}

void copyB(int *dst, int *src, int size){
    int i;
    for(i=0; i<size; i++){
        copy4(dst+4*i,src+4*i);
    }
}

void copyC(int *dst, int *src, int size){
```

```

    int i;
    for(i=0; i<size; i++){
        memcpy(dst+4*i,src+4*i,4);
    }
}

void copyD(int *dst, int *src, int size){
    int i, j;
    memcpy(dst,src,4*size);
}

int main (){
    clock_t t;
    int i;
    int *src, *dst, size;
    // size - numero de linhas da matriz
    // src - matriz de origem
    // dst - matriz de destino
    size = 400;

    /* funcao copyA */
    printf ("Calculating...\n");
    t = clock();
    for(i=0; i<500000; i++){
        copyA(dst,src,size);
    }
    t = clock() - t;
    printf ("copyA: %d clicks (%f seconds).\n",
        (int)t,((float)t)/CLOCKS_PER_SEC);

    /* funcao copyB */
    printf ("Calculating...\n");
    t = clock();
    for(i=0; i<500000; i++){
        copyB(dst,src,size);
    }
    t = clock() - t;
    printf ("copyB: %d clicks (%f seconds).\n",
        (int)t,((float)t)/CLOCKS_PER_SEC);

    /* funcao copyC */
    printf ("Calculating...\n");
    t = clock();
    for(i=0; i<500000; i++){
        copyC(dst,src,size);
    }
    t = clock() - t;
    printf ("copyC: %d clicks (%f seconds).\n",

```

```

        (int)t, ((float)t)/CLOCKS_PER_SEC);

    /* funcao copyD */
    printf ("Calculating...\n");
    t = clock();
    for(i=0; i<500000; i++){
        copyD(dst,src,size);
    }
    t = clock() - t;
    printf ("copyD: %d clicks (%f seconds).\n",
        (int)t, ((float)t)/CLOCKS_PER_SEC);

    return 0;
}

```

Compilada e executada em um determinado computador, o programa produziu o seguinte resultado:

```

Calculating...
copyA: 2676402 clicks (2.676402 seconds).
Calculating...
copyB: 1539304 clicks (1.539304 seconds).
Calculating...
copyC: 540464 clicks (0.540464 seconds).
Calculating...
copyD: 20508 clicks (0.020508 seconds).

```

Nestas condições, realize o que se pede:

- (0,8 ponto) Insira no código, nos locais onde julgar convenientes, as linhas necessárias para alocar e liberar as matrizes de inteiros usadas nos testes.
- (0,4 ponto) Descreva o funcionamento das função `copyA()`;
- (0,4 ponto) Descreva o funcionamento das função `copyB()`;
- (0,4 ponto) Descreva o funcionamento das função `copyC()`;
- (0,4 ponto) Descreva o funcionamento das função `copyD()`;
- (0,4 ponto) Descreva o processo adotado pelo programador para comparar o desempenho das funções de cópia que criou.
- (0,8 ponto) Realize uma análise comparativa do desempenho das quatro funções de cópia testadas.
- (0,4 ponto) Repare que as funções `copyB()` e `copyC()` possuem código bastante semelhante. Porque então produziram resultados tão diferentes nos testes de desempenho?

Consulte a documentação que achar necessária **apenas no computador**. Prepare um relatório em texto simples (usando emacs ou gedit, por exemplo) conforme modelo apresentado no final desse documento. Apresente os comentários que sejam necessários para o bom entendimento da sua resposta, bem como as cópias dos códigos fontes no dentro do proprio relatório. Submeta sua solução no SIGAA. **A cópia é proibida, sob pena de nulidade do exame!**

=== MODELO DE RELATORIO DE PROVA EM TXT ===

Aluno:

Q1: resposta

Q2: resposta

Q3: resposta

...
