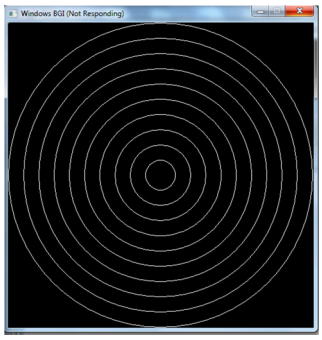
Universidade do Minho	**Escola de Engenharia da Universidade do Minho** **Mestrado Integrado em Eng. Electrónica Industrial e Computadores** **Programação de Computadores**	**2014/2015** **MIEEIC** **(1º Ano)** **1º Sem**
DOCENTE: Luís Paulo Reis		
FICHA DE EXERCÍCIOS 12: PROGRAMAÇÃO GRÁFICA		

Programação Gráfica



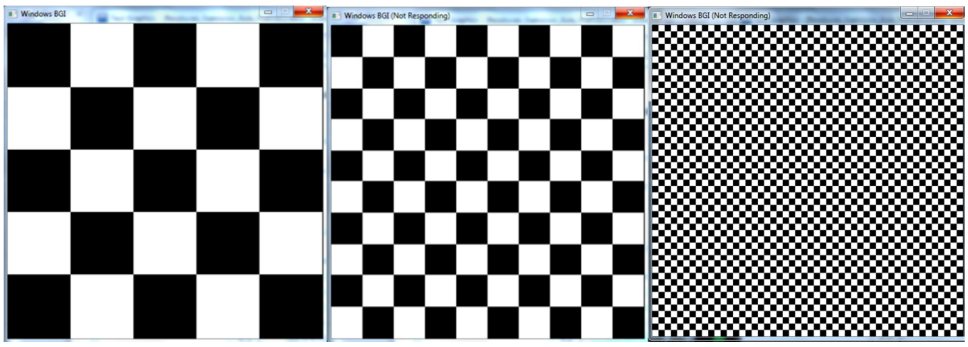
Suponha instalada a biblioteca gráfica BGI e o seguinte programa principal que abre uma janela gráfica com 500x500 pixeis de área útil:

```
#include "graphics.h"
int main(void)
{
    initwindow(509,511);
    …  //chamadas às funções das alíneas seguintes
    getchar();
}
```

10.1) Construa uma função `void desenha_circulos();` que desenhe 10 círculos concêntricos centrados na janela gráfica com diâmetros 50, 100, 150, 200, …, 500

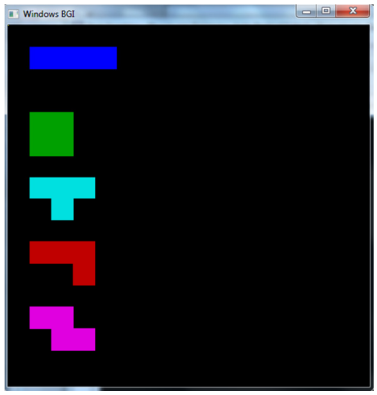
10.2) Construa uma função `void mouse_click();` que permita ao utilizador desenhar círculos preenchidos de cor aleatória em todos os pontos que clique no ecrã com o botão esquerdo do rato e quadrados preenchidos com o botão direito do rato. Os círculos deverão ter 20 pixéis de diâmetro e os quadrados 20 pixéis de lado.

10.3) Construa uma função `void desenha_grelha(int n)` que desenhe na janela gráfica uma grelha (preta e branca) com n*n células. Exemplos abaixo para valores de n=5, n=10 e n=50.



Suponha as seguintes estruturas:

```
#define amp 30    //ampliação do desenho
struct vert{
  int x, y;       //posicao x,y relativa a (xx,yy) do vertice
};
struct peca{
  int xx, yy;     //posicao absoluta
  int cor;        //cor da linha e enchimento (0-15)
  int nv;         //numero de vertices
  vert v[20];     //array de vertices da peca
    };
```



10.4) Construa uma função `void desenha_peca(peca p)` que desenhe peças (polígonos no ecrã com enchimento sólido, utilizando a cor respetiva). Os vértices do polígono são sucessivos, sendo o vértice 0, o vértice superior esquerdo, com coordenadas (0,0) e estando localizado no ponto `(xx,yy)*amp`. O desenho deve ser ampliado utilizando o valor do "`define`" correspondente.

10.5) Utilize a função `void desenha_tetris(void)` para desenhar conco peças do jogo Tetris (com cores 1-5) e enchimento sólido. Não precisa de desenhar as duas peças simétricas.

10.6) Construa um jogo simples em que um círculo azul com raio 20, aparece numa posição aleatória do ecrã até ao utilizador conseguir passar com o rato em cima dele, altura em que desaparece. Imediatamente aparece um novo círculo numa nova posição aleatória do ecrã e assim sucessivamente. O objetivo do utilizador é conseguir apanhar 10 círculos no menor tempo possível. No final do jogo aparece no ecrã o tempo gasto pelo utilizador. Nota: Para obter o tempo utilize as seguintes funções:

```
double diffclock(clock_t clock1,clock_t clock2) {
   return (double)(clock1-clock2)/CLOCKS_PER_SEC; }
…
clock_t begin, end;
begin=clock();
…
end=clock();
int time = diffclock(end,begin);
```

10.7) Construa um programa que permita jogar de modo muito simples o conhecido jogo "Snake". Inicialmente o jogo deve ser baseado num retângulo desenhado com a área de 200x200 pixéis centrado na janela gráfica. A cobra começa como uma linha com comprimento 50 desde (200,250) a (250,250). Em cada instante vai crescendo 2 pixéis sendo que a direção de crescimento pode ser rodada para a direita ou esquerda com os botões esquerdo e direito do rato. Se a cobra bater no seu próprio corpo ou no retângulo exterior, o jogo termina.

10.8) Altere o programa de modo a ter um auxílio do computador para jogar, i.e. fazendo com que o PC selecione, em cada instante (se necessário para não colidir) uma nova direção aleatória de modo a auxiliar o humano no seu jogo.

10.9) Acrescente ao jogo o aparecimento de alvos em posições aleatórias do ecrã. Sempre que o utilizador capturar um alvo, um novo alvo aparece e o score é incrementado em 100 pontos e a cobra acelera (i.e. a velocidade do jogo aumenta com o incremento do score).

10.10) Faça com que a cobra só cresça depois de capturar um alvo e durante um tempo curto, tal como acontece no jogo tradicional "Snake" completando assim um jogo completamente funcional de "Snake".