

JS 10 - MasterMind (12)



Mastermind to gra planszowa dla dwóch graczy, polegająca na odgadnięciu ukrytego kodu. Gra została wynaleziona w 1970 roku przez Mordechaja Nura. Zasady polegają na odszyfrowaniu czterech kolorowych kul w ciągu dziewięciu tur. Obecnie Mastermind dostępny jest w różnych wersjach, w tym mobilnej oraz magnetycznej. Gra stymuluje myślenie analityczne oraz rozwija umiejętności logicznego myślenia. Wspólne granie w Mastermind sprzyja współpracy i interakcji społecznej. Dzięki promocjom i różnorodnym ofertom, gra zyskuje na popularności także na platformach e-commerce.

Dwóch graczy z góry decyduje, ile rund rozegrają (liczba rund musi być parzysta). Jeden gracz układa kod z czterech większych, kolorowych pionków. Powtórzenia są dozwolone, więc gracz może również wybrać cztery pionki w tym samym kolorze. Kod zostaje umieszczony w czterech zakrytych zasłonką otworach, więc nie są widoczne dla drugiego gracza, który staje się odgadującym. Odgadujący stara się odgadnąć kod, zarówno kolory jak i ich kolejność, w dziesięciu (czasami osmiu lub dwunastu) próbach. Próba polega na umieszczeniu rzędu większych pionków na planszy, po czym zostaje oceniona przez gracza kodującego, poprzez ulokowanie od zera do czterech szpilek do udzielania informacji w tym samym rzędzie co domniemany kod. Każda czarna szpilka oznacza, że jeden z pionków o danym kolorze został umieszczony w domniemany kodzie na właściwym miejscu. Szpilka biała wskazuje na istnienie pionka o właściwym kolorze, lecz na niewłaściwym miejscu. Po ocenieniu próby odgadujący podejmuje kolejną. Gra kończy się po poprawnym wskazaniu kodu przez odgadującego lub po dziesięciu nieudanych próbach.

Pamiętaj o tym, by zrzut ekranu DOKUMENTOWAŁ Twoją pracę

Plansza (1)

- W swoim folderze utwórz 2 nowe dokumenty: **js10.html** i **js10.js**
- Otwórz oba dokumenty w notatniku, a dokument HTML w przeglądarce
- Do dokumentu **HTML** wklej tekst z ramki

```
<html>
<head>
  <meta charset=utf8>
  <title> MASTERMIND </title>
  <script src=js10.js></script>
</head>
<body>
<center>
  <font size=6>Libront Waclaw</font>
  <br>
  <canvas width=300 height=570 id=MASTERMIND></canvas>
</center>

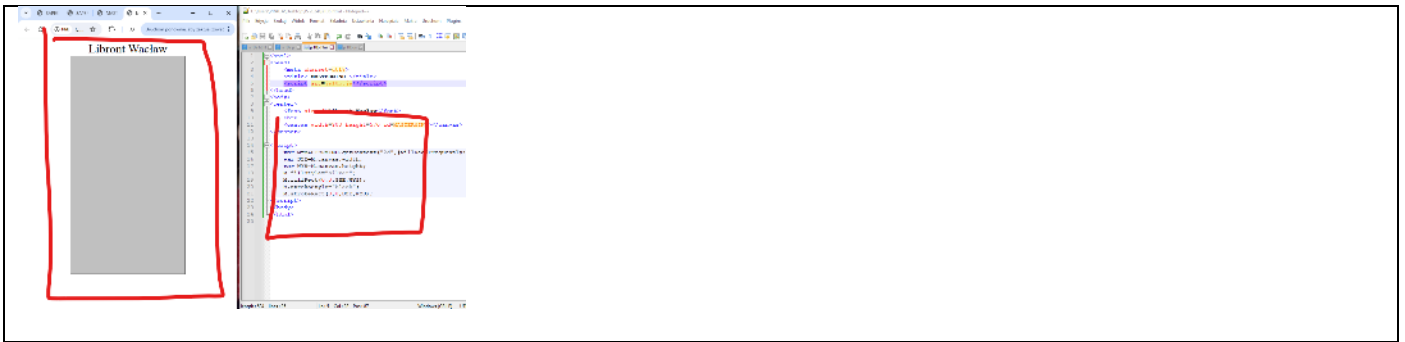
<script>
  var M=MASTERMIND.getContext("2d",{willReadFrequently: true,});
  var SZE=M.canvas.width;
  var WYS=M.canvas.height;
  M.fillStyle="silver";
  M.fillRect(0,0,SZE,WYS);
  M.strokeStyle="black";
  M.strokeRect(0,0,SZE,WYS);
</script>
</body>
</html>
```

obszar canvas: szare tło i czarny brzeg ramki, rozmiar 300x570 pikseli

dostęp do canvas za pomocą „uchwyty” o nazwie M

zadeklarowane dwie zmienne SZE i WYS, w których zapamiętujemy szerokość i wysokość obszaru canvas

- Zmień tytuł strony **MASTERMIND** na swoje **inicjały**
- Wpisz swoje **nazwisko i imię**
- Zapisz dokumenty i odśwież przeglądarkę
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



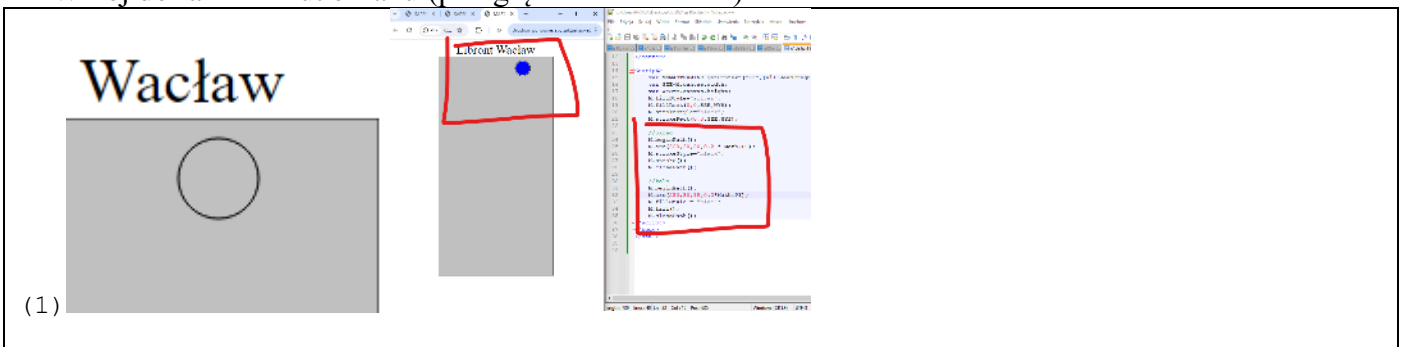
Okręgi i koła (1)

- HTML `</script>` `M.strokeRect(0,0,SZF)`
- (1)

```
M.beginPath();
  M.arc(220,30,20,0,2 * Math.PI);
  M.strokeStyle="black";
  M.stroke();
  M.closePath();
```

- HTML `</script>` `M.closePath();`
- (2) `M.closePath();`

- Zapisz dokumenty i odśwież przeglądarkę
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



Funkcja Pion (1)

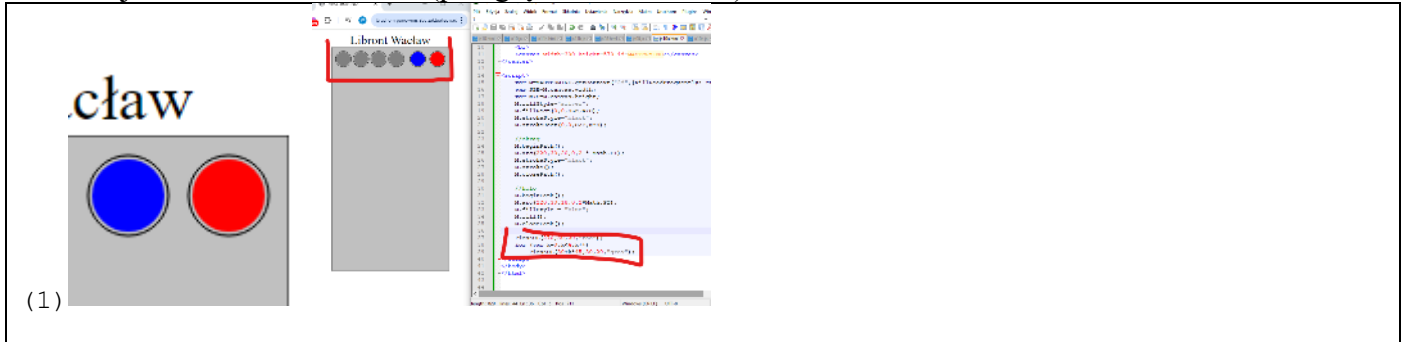
Pion to połączenie okręgu i koła

- JS
- ```
function PionKOL(x,y,pro,kol){
 //brzeg
 M.beginPath();
 M.arc(x,y,pro,0,2 * Math.PI);
 M.strokeStyle="black";
 M.stroke();
 M.closePath();
 //srodek
 M.beginPath();
 M.arc(x,y,pro-2,0,2 * Math.PI);
 M.fillStyle=kol;
 M.fill();
 M.closePath();
}
```

```
M.closePath() ;
```

```
</script>
```

- (1) `PionKOL(270,30,20,"red");`
- Narysuj 4 szare piony
- `for (var k=0;k<4;k++)`  
`PionKOL(30+k*45,30,20,"grey");`
- Zapisz dokumenty i odśwież przeglądarkę
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Plansza (1)

Na planszy znajduje się miejsce na:

- piony, które wylosował komputer
- pola na pionki wstawiane przez użytkownika
- pola na pionki informacyjne wstawiane przez komputer
- pionki służące do wybierania kolorów

### JS

```
function Plansza(){
 var x=30;
 var y=30;
 var s=45;
 var r=20;
 var kol="black";
 //wiersz 0 - tutaj komputer wstawia losowe pionki
 for (var k=0;k<4;k++)
 PionKOL(x+k*s,y,r,kol);
 //wiersze 1-10 - tutaj gracz wstawia swoje pionki
 y=40;
 for (var w=1;w<=10;w++)
 for (var k=0;k<4;k++)
 PionKOL(x+k*s,y+w*s,r,kol);
 //wiersze 1-10 - tutaj komputer wstawia pionki informacyjne
 x=205;
 y=40;
 s=25;
 r=10;
 for (var w=0;w<=10;w++)
 for (var k=0;k<4;k++)
 PionKOL(x+k*s,y+w*(s+20),r,kol);
 //pola do wybierania kolorów pionków
 y=540;
 x=30;
 s=48;
 r=20;
 PionKOL(x+0*s,540,r,"blue");
 PionKOL(x+1*s,540,r,"lime");
 PionKOL(x+2*s,540,r,"aqua");
 PionKOL(x+3*s,540,r,"red");
 PionKOL(x+4*s,540,r,"magenta");
 PionKOL(x+5*s,540,r,"yellow");
}
```

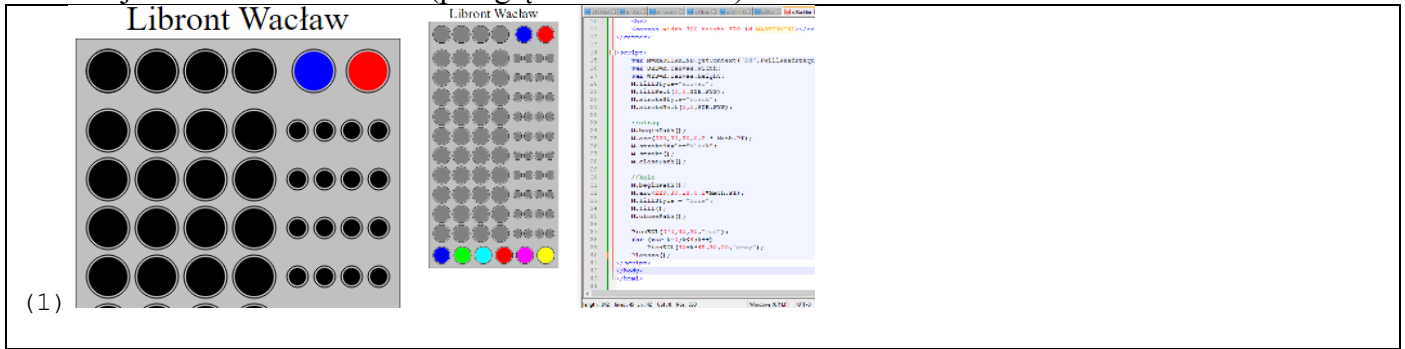
```
PionKOL(30
```

```
</script>
```

```
Plansza();
```

- (1) Zapisz dokumenty i odśwież przeglądarkę

- Zmień kolor „dziur” na szary
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Zdarzenia (1)

Zdarzenia to rzeczy, które wykonują się na elementach HTML. Przykładowo kliknięcie przez użytkownika przycisku wywołuje zdarzenie `click`, najechanie myszą na obrazek – `mouseover`, wybranie elementu – `focus`, wysłanie formularza – `submit`, naciśnięcie przycisku na klawiaturze – `keydown`.

```

<center width=300
</center>

<label id=idMYSZ></label>
<label id=idKLAW></label>

```

```

Plansza();
</script>
var MYSZKA={x:0,y:0};
Zdarzenia();
JS

```

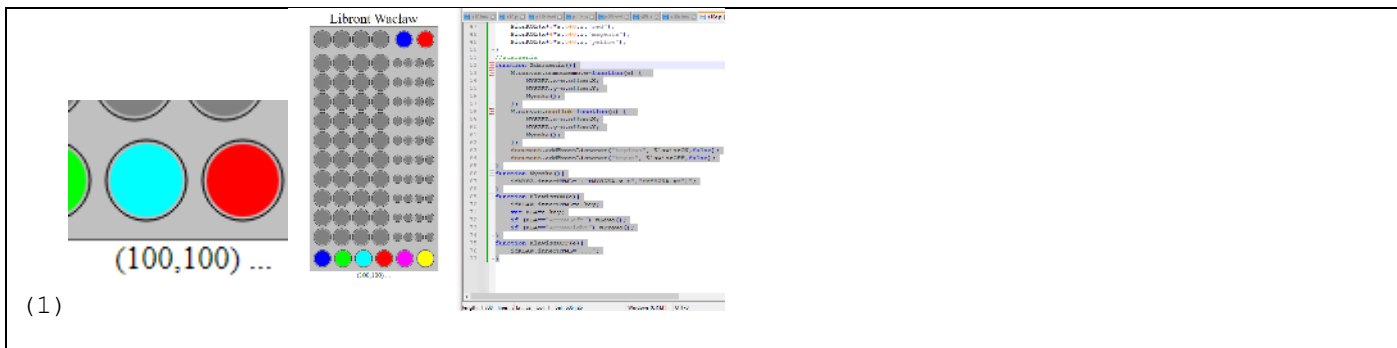
```

function Zdarzenia(){
 M.canvas.onmousemove=function(e) {
 MYSZKA.x=e.offsetX;
 MYSZKA.y=e.offsetY;
 Myszka();
 };
 M.canvas.onclick=function(e) {
 MYSZKA.x=e.offsetX;
 MYSZKA.y=e.offsetY;
 Myszka();
 };
 document.addEventListener("keydown", KlawiszON,false);
 document.addEventListener("keyup", KlawiszOFF,false);
}
function Myszka(){
 idMYSZ.innerHTML="MYSZKA: (" +MYSZKA.x +"," +MYSZKA.y+")";
}
function KlawiszON(e){
 idKLAW.innerHTML=e.key;
 var KLA=e.key;
 if (KLA=="ArrowLeft") Wlewo();
 if (KLA=="ArrowRight") Wprawo();
 if (KLA=="Enter") WDol();
}
function KlawiszOFF(e){
 idKLAW.innerHTML="...";
}

```

funkcja `Zdarzenia()` zawiera obsługę przesuwania myszki `onmousemove` i klikania `onclick` w zmiennej `MYSZKA` zapisujemy współrzędne wskaźnika myszki i wyświetla na stronie funkcja obsługuje również klawiaturę - do strzałek lewa i prawa przypisano funkcje (na razie nie obsługiwane)

- Zapisz dokumenty i odśwież przeglądarkę
- (1) Ustaw wskaźnik myszki w punkcie **100,100**
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Pobieranie kolorów (1)

Myszka wskazuje kolorowe piksele na planszy - ich rozpoznawanie będzie niezbędne

- `<label id=idKOLOR></label>`
- `</center>`
- `<br>`
- `<label id=idKOLOR></label>`
- JS

```
function JakieNRGB(x,y){
 var kol = M.getImageData(x,y,1,1);
 var R=kol.data[0];
 var G=kol.data[1];
 var B=kol.data[2];
 var n=RGBNR({R,G,B});
 return {n,R,G,B};
}
function RGBNR(rgb){
 var nr=255;
 var R=rgb.R;
 var G=rgb.G;
 var B=rgb.B;
 if (R==0 && G==0 && B==0) nr=0;//black
 if (R==0 && G==0 && B==255) nr=1;//blue
 if (R==0 && G==255 && B==0) nr=2;//lime
 if (R==0 && G==255 && B==255) nr=3;//aqua
 if (R==255 && G==0 && B==0) nr=4;//red
 if (R==255 && G==0 && B==255) nr=5;//magenta
 if (R==255 && G==255 && B==0) nr=6;//yellow
 if (R==255 && G==255 && B==255) nr=7;//white
 if (R==128 && G==128 && B==128) nr=8;//grey
 if (R==192 && G==192 && B==192) nr=9;//silver
 return nr;
}
function Kolory(){
 var nrgb=JakieNRGB(MYSZKA.x,MYSZKA.y);
 idKOLOR.innerHTML="RGB: (" +nrgb.R+" ,"+nrgb.G+" ,"+nrgb.B+") "+ nrgb.n;
}
function UstawPiona(x,y,r){
 var nrgb=JakieNRGB(MYSZKA.x,MYSZKA.y);
 var kol="rgba (" +nrgb.R+" ,"+nrgb.G+" ,"+nrgb.B+" ,1)"
 PionKOL(x,y,r,kol);
}
}
```

funkcja *JakieNRGB* podane numer koloru i składowe RGB na podstawie współrzędnych myszki

funkcja *RGBNR* zamienia parametry RGB koloru na numer z tabeli kolorów, którymi się posługujemy w programie  
jeżeli nie znalazło koloru, to nr=255

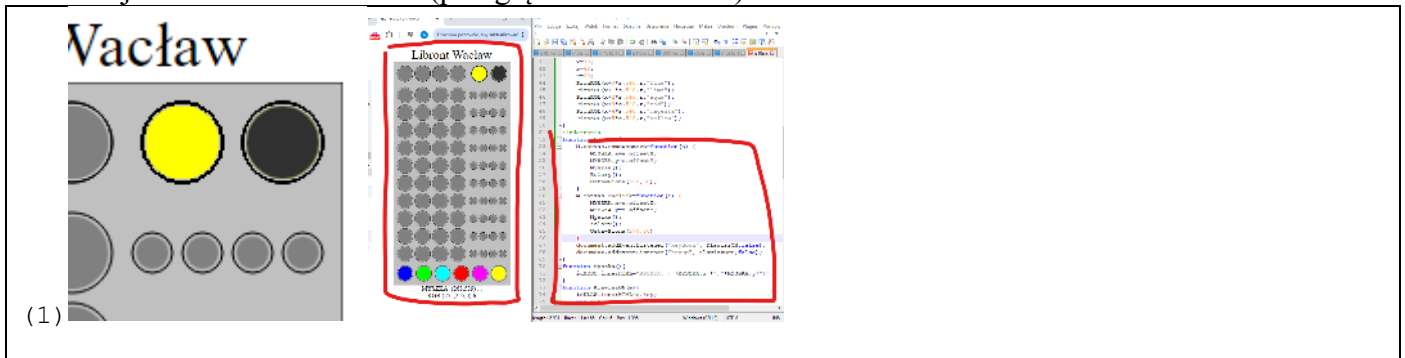
funkcja *Kolory* zapisuje parametry wskazywanego przez myszkę piksela na ekranie

funkcja *UstawPiona* rysuje pion w punkcie (x,y) na podstawie koloru wskazywanego przez myszkę  
kolor RGB ustawiamy na podstawie funkcji *JakieNRGB*

`Myszka();`

- `Zdarzenia onmousemove };`
- `Kolory();`
- `UstawPiona(220,30,20);`
- w zdarzeniu onclick podobnie
- `Kolory();`
- `UstawPiona(270,30,20);`

- Zapisz dokumenty i odśwież przeglądarkę
- myszka wskazuje kolor, który wyświetlany jest w pierwszym polu z prawej
- klikamy w kolor - ustawiamy drugie kółko z prawej
- (1) Ustaw następujący schemat kolorów
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Piony na planszę (1)

W czasie gry musimy ustawiać piony w określonych polach planszy.

Jedno pole jest zawsze wyróżnione - do tego wstawiamy piony

Wyróżnione pole możemy zmieniać w kolumnach strzałkami: ← - →

ENTER ↵ przenosi wyróżnienie do nowego wiersza

kliknięcie w kolor ustawia pion w wyróżnionym polu

Zdarzenia ();

- `</script>`
- `var plaWIE=1;`
- `var plaKOL=1;`
- kolumny planszy 1-4 wiersze planszy 1-10
- JS

```
function Wprawo() {
 if (plaKOL<4) plaKOL++;
 else plaKOL=1;
}
function Wlewo() {
 if (plaKOL>1) plaKOL--;
 else plaKOL=4;
}
function WDol() {
 if (plaWIE<=10) {
 plaWIE++;
 plaKOL=1;
 }
}
function PLAXY(k,w) {
 var x0=30;
 var y0=85;
 var r=20;
 var s=45;
 var x=x0+(k-1)*s;
 var y=y0+(w-1)*s;
 return {x,y};
}
```

funkcja Wprawo zwiększa wskaźnik kolumn o 1, gdy osiągnie czwartą kolumną to wracamy do 1

funkcja Wlewo działa podobnie, z pierwszej przechodzimy do czwartej

funkcja WDol wciskamy enter i przechodzimy do pierwszej kolumny nowego wiersza

maksymalnie 10 wierszy o potem koniec gry

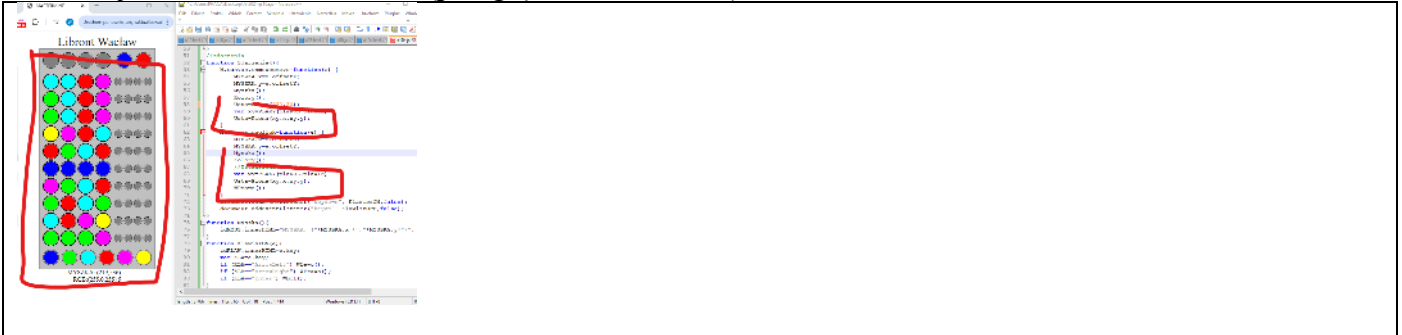
`UstawPiona(220,30,20);`

- `onmousemove`
- `var xy=PLAXY(plaKOL,plaWIE)`
- `UstawPiona(xy.x,xy.y,10);`

na podstawie współrzędnych wyróżnionego pola planszy plaKOL i plaWIE wyliczamy współrzędne pion i ustawiamy mniejszego pion (żeby nie zasłaniał) w kolorze wskazywanym przez myszkę

```
UstawPiona(270,30,20);
```

- onclick };
- var xy=PLAXY(plaKOL,plaWIE)  
UstawPiona(xy.x,xy.y,20);
- WPravo();  
po kliknięciu ustawiamy kolor i dodatkowo przesuwamy się na kolejne pole w wierszu
- Wypełnij wszystkie pola planszy dowolnymi kolorami
- Zapisz dokumenty i odśwież przeglądarkę
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Tablica (1)

Wszystkie wybrane kolory przez użytkownika, a także wylosowane przez komputer najłatwiej zapamiętać w tablicy, która będzie zawierać numery kolorów

```
var plaKOL=1;
```

- </script>
- var POLA=[];
- ZerowanieTablicy();
- PrzerysujPlansze();
- JS

```
function ZerowanieTablicy(){
 var p=[];
 for (var w=0;w<=10;w++){
 p[w]=[];
 for (var k=0;k<=4;k++){
 p[w][k]=8;
 }
 POLA=p;
 }
}
function PrzerysujPlansze(){
 for (var w=1;w<=10;w++){
 for (var k=1;k<=4;k++){
 var nr=POLA[w][k];
 var xy=PLAXY(k,w);
 var kol=NRKOL(nr);
 PionKOL(xy.x,xy.y,20,kol);
 }
 }
}
function NRKOL(nr){
 var kol="black";
 if (nr==0) kol="black";
 if (nr==1) kol="blue";
 if (nr==2) kol="lime";
 if (nr==3) kol="aqua";
 if (nr==4) kol="red";
 if (nr==5) kol="magenta";
 if (nr==6) kol="yellow";
 if (nr==7) kol="white";
 if (nr==8) kol="grey";
 if (nr==9) kol="silver";
 return kol;
}
```

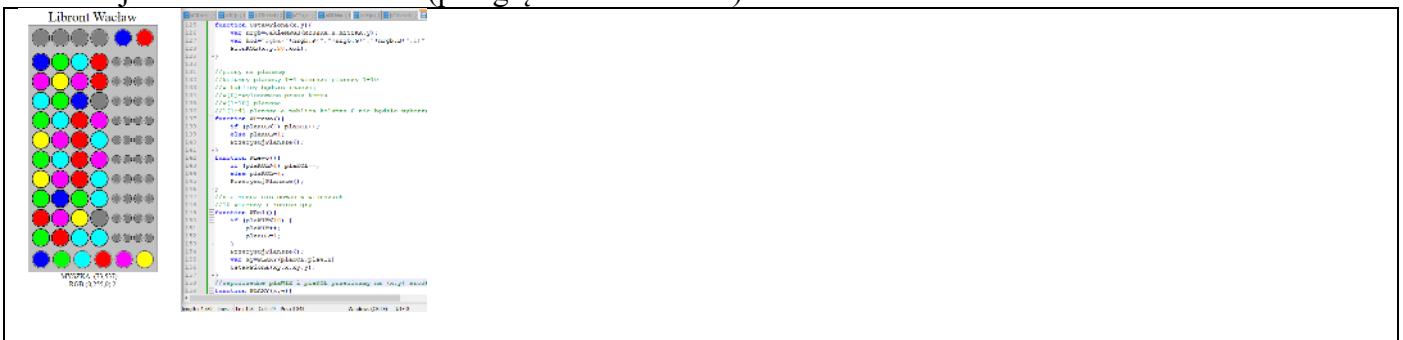
funkcja ZerowanieTablicy wstawia do wszystkich komórek tablicy POLA liczbę 8 - kolor szary

funkcja PrzerysujPlansze sprawdza komórki tablicy POLA i ustawia pionki na planszy zgodnie z kolorami

funkcja NRKO zamienia numer koloru z tablicy POLA na właściwą nazwę koloru

```
else plaKOL=1;
```

- Wprawo }
- PrzerysujPlansze ();
- Tak samo wstaw funkcję do WLewo i Wdol  
po każdym kliknięciu myszki i wciśnięciu ENTERa rysujemy całą planszę - na wszelki wypadek  
UstawPiona (xy.x,xy.y,20);
- onclick Wprawo ();  
var nrgb=JakieNRGB (MYSZKA.x,MYSZKA.y);  
POLA[plaWIE] [plaKOL]=nrgb.n;  
po kliknięciu ustawiamy odpowiednią komórkę tablicy POLA  
Wprawo ();
- onclick }  
var xy=PLAXY (plaKOL,plaWIE)  
UstawPiona (xy.x,xy.y,10);  
nowe pole z prawej i od razu kolorowy pion - nie trzeba ruszać myszką  
PrzerysujPlansze ();
- Wdol }  
var xy=PLAXY (plaKOL,plaWIE)  
UstawPiona (xy.x,xy.y,10);  
nowe pole z prawej i od razu kolorowy pion - nie trzeba ruszać myszką
- Zapisz dokumenty i odśwież przeglądarkę
- Pokoloruj wszystkie pola planszy
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Komputer losuje kolory (1)

Komputer losuje 4 kolory spośród sześciu i wstawia ich numery do wiersza 0 w tablicy POLA

Wersje trudniejsze Masterminda pozwalają wybierać puste pola lub kolory podwójne

```
<label id=idKOLOR></label>
```

- </center>
- <br>
- <input type=button id=idMIESZAJ value="START" onclick=Mieszaj (>>
- JS

```
function losowa (p, k) {
 return Math.floor(Math.random() * (k-p+1)+p);
}
function Losowanie4Kolorow () {
 var tk=[0,0,0,0,0,0];
 var tx=[0,0,0,0];
 var x=0
 do {
 var nr=losowa (0,5);
 if (tk[nr]==0){
 tk[nr]=1;
 tx[x]=nr+1;
 x++;
 }
 }
 } while (x < 4);
```



```

 return tx;
}
function WstawLosoweKolory(){
 var t4=Losowanie4Kolorow();
 for (var i=0;i<4;i++){
 var x=30+45*i;
 var y=30;
 var nr=t4[i];
 var kol=NRKOL(nr);
 PionKOL(x,y,20,kol);
 POLA[0][i+1]=nr;
 }
}
function Mieszaj(){
 if (idMIESZAJ.value=="START"){
 idMIESZAJ.value="STOP";
 StartMieszania();
 }
 else{
 idMIESZAJ.value="START";
 StopMieszania();
 }
}
function StartMieszania(){
 takt=setInterval(WstawLosoweKolory, 500);
}
function StopMieszania(){
 clearInterval(takt);
 UkryjKolory()
}
function UkryjKolory(){
 for (var i=0;i<4;i++){
 var x=30+45*i;
 var y=30;
 PionKOL(x,y,20,"black");
 }
}
}

```

*funkcja losowa(p,k) daje losowe liczby całkowite z przedziału p..k*

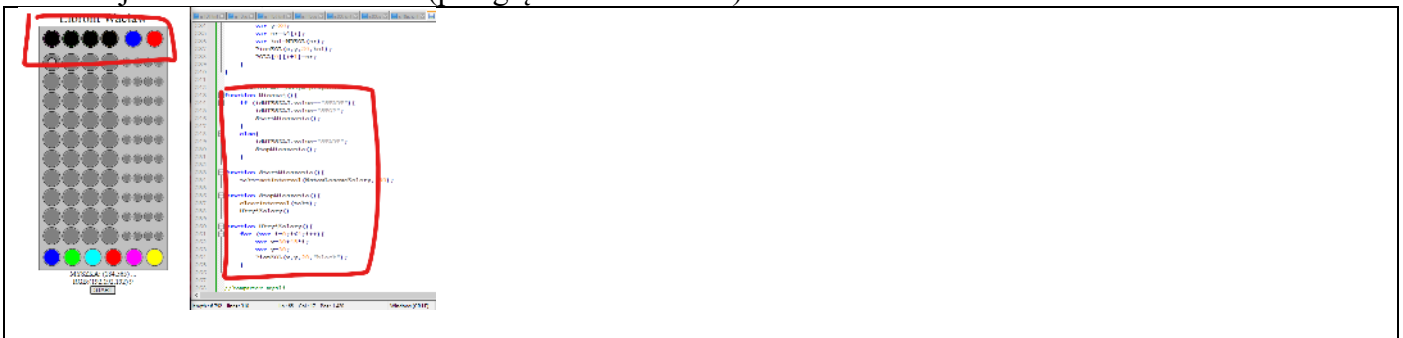
*funkcja Losowanie4Kolorow z sześciu kolorów wybiera 4 - po jednym bez duplikatów najpierw ustawiana jest tablica tk pętla do-while kreci się tyle razy aż będzie 4 jedynki gdy jest 4 jedynki to pozycje jedynek (numery kolorów) przepisywane do tablicy tx*

*funkcja WstawLosoweKolory pobiera numery kolorów z tablicy tx wylicza ich kolor, rysuje pion wstawia numer koloru do tablicy POLA*

*funkcja Mieszaj przypisane jest do przycisku START-STOP gdy wciskamy START to uruchamiamy funkcję StartMieszania co 500 milisekund wywołane wstawianie kolorów gdy wciskamy STOP, to uruchamiamy funkcję StopMieszania - zatrzymanie wstawiania kolorów*

*funkcja UkryjKolory - „chowa” za czarnymi pionami wylosowane kolory*

- Zapisz dokumenty i odśwież przeglądarkę
- Mieszanie kolorów następuje co pół sekundy (500 milisekund)  
Przyspiesz losowanie - ustaw **50 milisekund**
- Wylosuj kolory
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Komputer myśli (1)

*Komputer wylosował kolory, wybrano w pierwszym wierszu schemat kolorów.*

Komputer udziela informacji zwrotnej: w małych polach pojawiają się:

- czarne pionki - gdy jakiś kolor **jest** na swoim miejscu
- białe pionki - gdy jakiś kolor **nie jest** na swoim miejscu

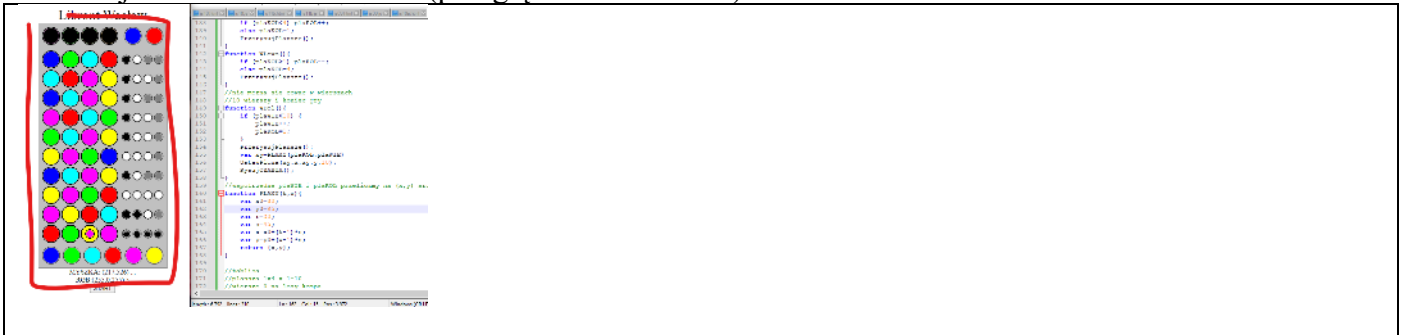
## • JS

```
function KomputerMysli() {
 var cza=0;
 for (var k=1;k<=4;k++){
 kom=POLA[0][k];
 gra=POLA[plawIE-1][k];
 if (kom==gra) cza++;
 }
 var bia=0;
 for (var k=1;k<=4;k++){
 kom=POLA[0][k];
 for (var g=1;g<=4;g++){
 gra=POLA[plawIE-1][g];
 if (kom==gra && g!=k) bia++;
 }
 }
 return {cza,bia};
}

function RysujCZABIA() {
 czabia=KomputerMysli();
 x=205;
 y=40;
 s=25;
 r=10;
 var w=plawIE-1;
 var ile=czabia.cza+czabia.bia;
 for (var p=0;p<ile;p++){
 if (p<czabia.cza)
 kol="black";
 else
 kol="white";
 PionKOL(x+p*s,y+w*(s+20),r,kol);
 }
}
```

UstawPiona(xy, x, xy.y, 10);

- Wdół
- RysujCZABIA();  
wciskamy ENTER i komputer udziela wskazówek na podstawie wstawionych kolorów
- Zapisz dokumenty i odśwież przeglądarkę
- Rozegraj partię (10 układów kolorów)
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Koniec Gry (1)

Gra kończy się, gdy zgadniesz układ kolorów albo wyczerpiesz limit 10 wierszy

## • JS

```
function OdkryjKolory() {
 for (var i=0;i<4;i++){
 var x=30+45*i;
 var y=30;
 var nr=POLA[0][i+1];
 var kol=NRKOL(nr);
 PionKOL(x,y,20,kol);
 }
}
```

```

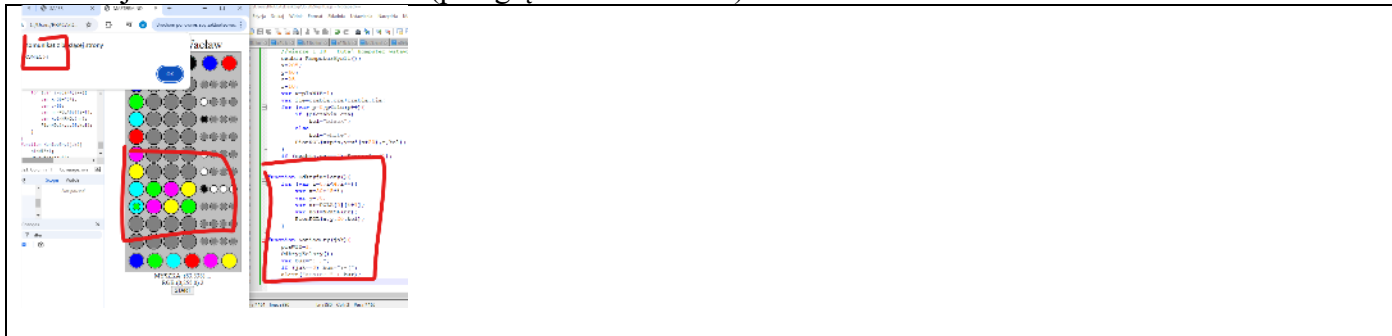
}
}
function KoniecGry(jak) {
 plaWIE=1;
 OdkryjKolory();
 var buz=":-)";
 if (jak==0) buz=":-(";
 alert("KONIEC " + buz);
}

```

*funkcja OdkryjKolory zastępuje czarne pionki kolorami ukrytymi w tablicy POLA[0]  
funkcja KoniecGry ustawia na nowo pierwszy wiersz i okrywa kolory*

```
PionKOL(x+p*s,y+w*(s+20),r,kol);
```

- RysujCZABIA
- `if (czabia.cza==4) KoniecGry(1);`  
gdy cztery czarne to koniec
- `RysujCZABIA();`
- Wdol
- `if (plaWIE==11) KoniecGry(0);`  
gdy wypełnione 10 wierszy to koniec
- Zapisz dokumenty i odśwież przeglądarkę
- Rozwiąż układ lub wypełnij 10 wierszy aby otrzymać komunikat o końcu gry
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)



## Od Nowa (1)

- HTML `</center>`
- `<input type=button value="OD NOWA" onclick=OdNowa()>`
- JS

```

function UstawCanvas(){
 M.fillStyle="silver";
 M.fillRect(0,0,SZE,WYS);
 M.strokeStyle="black";
 M.strokeRect(0,0,SZE,WYS);
}
function OdNowa(){
 UstawCanvas();
 Plansza();
 var MYSZKA={x:0,y:0};
 Zdarzenia();
 var plaWIE=1;
 var plaKOL=1;
 var POLA=[];
 ZerowanieTablicy();
 PrzerysujPlansze();
 Mieszaj();
}

```

*czyścimy obszar canvas  
rysujemy planszę  
Odnawiamy zdarzenia myszki i klawiatury  
ustawiamy wskaźnik pionów  
tworzymy i zerujemy tablicę na piony*

*przerysowujemy plansze  
uruchamiamy losowanie kolorów*

- Zapisz dokumenty i odśwież przeglądarkę
- Zagraj
- Wklej do ramki zrzut ekranu (przeglądarka i notatnik)

