

I. 375.- Fermatpont,Pólya-mode

- Kovács Balázs Marcell
- Budapest, ELTE Radnóti Miklós Gyakorló Iskola,12. évf.
- kbalazsm@gmail.com

A megoldáshoz használt programozási eszköz: Wolfram *Mathematica* 10.0.0.0

A program ingyenes próbaverziója letölthető, de a megoldás futtatásához elegendő a Wolfram CDF Player letöltése.

Linkek: <https://www.wolfram.com/mathematica/trial/> <https://www.wolfram.com/cdf-player/>

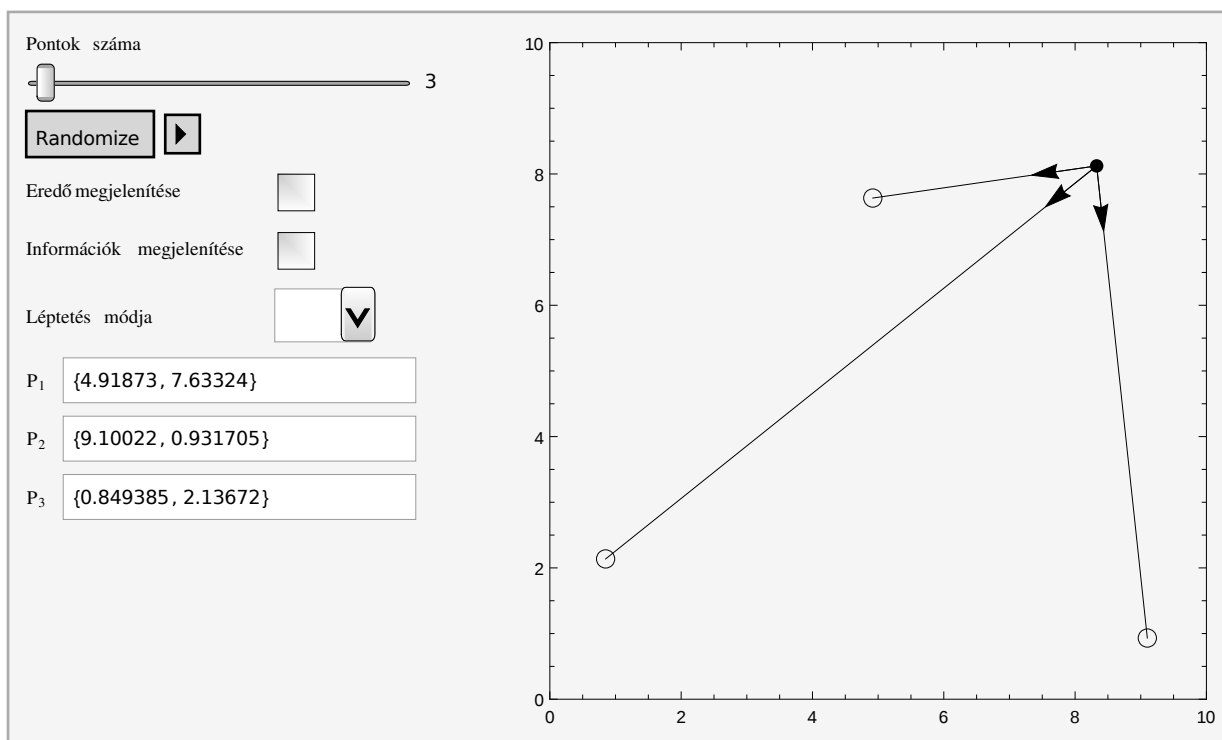
A lyukak és a közös pont egy 10x10-es négyzeten belül helyezkednek el. Ezen belül a lyukak pozíciója tetszőlegesen változtatható az egér használatával, de pontos koordináták is megadhatók a baloldali oszlopban. A pontok (lyukak) számát egy csúszkán 3 és 8 között tetszőlegesen változtathatjuk. Az *Eredő megjelenítése* opció bekapcsolása esetén az ábrán a közös pontból kiinduló egységvektorok mellett a program piros színnel ezek eredőjét is ábrázolja. A *Randomize* gomb a lyukakat véletlenszerűen szétszórja a 10x10-es négyzeten belül. A ▶ gombra kattintva indítható el az animáció, melynek során a közös pont a keresett pontba áll be.

A megoldás során használt legfontosabb nyelvi eszköz a Dynamic függvény, ez teszi lehetővé az interakciót, illetve az animálást is. A léptetés kétféle módja közül lehet választani. Az egyik esetben a az eredő erővel egyenesen arányos a közös pont elmozdulása, a másik esetben mindig ugyanakkora. Az első esetben a közeledés magát finomítja, míg a másodikban a közös pont mindig túlhalad a kívánt ponton és onnan visszafordul. A végtelen hosszú ugrálás elkerülése érdekében minden egyes visszafordulásnál a felére csökkentjük a lépésenként megtett távolságot. Ha közös pont mozgása már elhanyagolható, leállítjuk az animációt.

A megoldás során nagymértékben kihasználjuk a *Mathematica* nyelv (Wolfram nyelv) adta lehetőségeket, például vektorok összegzését vagy vektorok egységvektorra való alakítását lehetővé tevő beépített függvényeket. A rendszer automatikus optimalizációkat végez, így a beépített függvények használata gyorsabb, mint az azonos műveletek "kézzel" való elvégzése.

A forráskód a dokumentum végén megtalálható.

Out[169]=



```

In[169]:= DynamicModule[
{
n = 3, (*Pontok száma*)
pc = {5, 5}, (*közös pont*)
pont = Table[RandomReal[10, 2], {n}], (*a lyukak helyei*)
controls, (*szöveges bevitel a pontos koordinátához*)
locators, (*a lyukak elhelyezése kattintással*)
eredo, (*eredő vektorst meghatározás*)
v, (*A közös pontból egy adott pont felmutató normálvektor*)
t = 0, (*az animáció pörgetése*)
vorosnyil = True, (*eredő megjelenítése*)
lockor, (*a locator ne az alapértelmezés szerint jelenjen meg*)
animator, (*animáció vezérlése*)
animatorProto,
minx = ∞, miny = ∞, maxx = -∞, maxy = -∞, (*a közös pont szélső helyzetei*)
x, y,
elozopc = {5, 5}, (*követjük, hogy mennyit mozdult a közös pont*)
d = 0, (*elmozdulás*)
factor = 1, (*animáció finomítása*)
running = True, (*megy-e az animáció*)
runnable = True, (*animator letiltása*)
erofuggo = True, (*a lépés arányos-e az eredő erővel*)
info = False (*információ megjelenítése*)
},

controls = {

```

```

{Text@"P1", InputField[Dynamic@pont[[1]]]},
{Text@"P2", InputField[Dynamic@pont[[2]]]},
{Text@"P3", InputField[Dynamic@pont[[3]]]},
{Text@"P4", InputField[Dynamic@pont[[4]]]},
{Text@"P5", InputField[Dynamic@pont[[5]]]},
{Text@"P6", InputField[Dynamic@pont[[6]]]},
{Text@"P7", InputField[Dynamic@pont[[7]]]},
{Text@"P8", InputField[Dynamic@pont[[8]]]}
};

lockor = Graphics[{Circle[{0, 0}, 1]}, ImageSize → 10];

locators = {
  Locator[Dynamic@pont[[1]], lockor],
  Locator[Dynamic@pont[[2]], lockor],
  Locator[Dynamic@pont[[3]], lockor],
  Locator[Dynamic@pont[[4]], lockor],
  Locator[Dynamic@pont[[5]], lockor],
  Locator[Dynamic@pont[[6]], lockor],
  Locator[Dynamic@pont[[7]], lockor],
  Locator[Dynamic@pont[[8]], lockor]
};

eredo[a_] := Sum[v@pont[[i]], {i, 1, a}];

v[a_] := Normalize[a - pc];

animator = Dynamic[Animator[Dynamic[t],
  AnimationRepetitions → Infinity, AppearanceElements → {"PlayPauseButton"},
  ImageSize → Large, AnimationRunning → running, Enabled → runnable],
  TrackedSymbols → {running, runnable}];

Dynamic[
  pc += If[erofuggo, eredo[n] / 10, Normalize[eredo[n]] * 0.5] * factor + 0 * t;
  {x, y} = pc;
  minx = Min[minx, x];
  miny = Min[miny, y];
  maxx = Max[maxx, x];
  maxy = Max[maxy, y];
  If[(minx < x < maxx) || (miny < y < maxy),
    factor /= 2;
    minx = miny = ∞;
    maxx = maxy = -∞;
  ];
  d = EuclideanDistance[pc, elozopc];
  elozopc = pc;

```

```

If[d < If[erofuggo, 10-4, 10-5],
  running = False;
  runnable = False;
,
  running = True;
  runnable = True;];

"",
TrackedSymbols => {t}]

(*Ha b rmilyen be ll t s megv ltozik, reseteljük a min/max-ot s a factort*)
Dynamic[
  minx = miny = ∞;
  maxx = maxy = -∞;
  factor = 1;
  elozopc = pc;
  runnable = True;
  pont;
  n;
  "",
  TrackedSymbols => {pont, n}]

Panel@Grid[{
  {Column[{
    Text["Pontok sz ma"],
    Slider[Dynamic@n, {3, 8, 1}, Appearance -> "Labeled" ],
    Grid[{{Tooltip[Button["Randomize", pont = Table[RandomReal[10, 2], {8}],
      ImageSize -> Large], "V letlenszerű pontok"],
      Tooltip[Dynamic@animator, "L ptet s / meg ll t s"]},
    }]],
    Grid[{{Text["Eredő megjelen t se"], Checkbox[Dynamic@vorosnyil]},
      {Text["Inform ci k megjelen t se"], Checkbox[Dynamic@info]},
      {Text["L ptet s m dja"],
        PopupMenu[Dynamic[erofuggo], {True -> "Eredő erővel ar nyos", False ->
          "Eredő erő nagys g t l független"}, True, PopupMenu[Null, {}]]}
      }, Alignment -> Left],
    Dynamic@Grid[controls[[1 ;; n]]],
    Spacer[{0, 10}],
    Dynamic@
      If[info, Row[{Text@"factor=", Text@factor, Text[" d="], Text@d}], ""]
    }],
  Dynamic@Graphics[{{PointSize[Large], Point[pc],
    Table[{Line[{pc, pont[[i]]}], Arrow[{pc, pc + v@pont[[i]]}], {i, 1, n}},
    locators[[1 ;; n]],
    Red, If[vorosnyil, Arrow[{pc, pc + eredo[n]}], Null]
  },
  PlotRangeClipping -> False,
  Frame -> True, PlotRange -> {{0, 10}, {0, 10}}, ImageSize -> Medium]

```

```
    }  
    }, Selectable → False, Alignment → {Left, Top}]  
]
```