

Úloha 1-2: Karlík na minách

Jan Horáček

Gymnázium, Brno, Vídeňská 47; jan.horacek@seznam.cz

2. listopadu 2014

1 Úvod

Úlohu "Karlík na minách" jsem z velké části převedl na grafový problém - konkrétně prohledávání do šířky. A to proto, že způsob, jakým probíhá odkrývání polí, pokud Karlík klikne na pole, které nesousedí s žádnou minou, se nápadně podobá právě průchodu grafu do šířky. Ukončovací podmínkou procházení do šířky je v tomto případě nenulový počet sousedních min daného pole. Podrobný popis algoritmu viz níže.

2 Popis algoritmu

Celý algoritmus se skládá ze 4 základních kroků:

1. Ze vstupního pole *plane* vypočteme počty sousedních min všech polí. Pokud se na daném poli přímo nachází mina, uložíme do příslušného prvku 2-rozměrného pole číslo -1 . Toto pole (označme ho *neighbors*) budeme nadále používat jako stavové pole pro zbytek algoritmu.
2. Dokud existují, hledáme nulové hodnoty v poli *neighbors*. To znamená, že hledáme taková políčka, která nesousedí s žádnou minou. Tato políčka expandujeme prohledáváním do šířky, přičemž pole *neighbors* nám poslouží jako místo pro ukládání flagu o tom, že algoritmus dané políčko už navštívil (políčkám, která navštíví, přiřadí $neighbors[x][y] = -1$).
3. Nyní jsou všechna nulová pole expandovaná - vše, co algoritmus mohl expandovat "rychle" (od nulového políčka se expanduje dále, viz zadání), bylo expandováno. Nezbyvá tedy, než "hrubou silou" otevřít všechna pole, která splňují podmínku $neighbors[x][y] > 0$ - tj. políčka sousedící s nějakou minou, mina na nich ale není a zároveň ještě nebyly navštíveny.

3 Závěr

Karlík tedy potřebuje celkem tolik kliků, kolik je uzavřených oblastí s nulami + tolik kliků, kolik je nenulových polí, která nesouvisí alespoň s jedním nulovým polem. Na základě neúspěšného hledání záludnosti v zadání a toho, že jsem v simulátoru nikdy "neklikl" na již odkryté pole, si troufám říci, že lepší už Karlík snad ani nemůže být.