

KSI 2012

Úloha 1-3: Návštěva

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

13. října 2012

1 Úvod

Dlouho jsem tápal, jak tuto úlohu relativně jednoduše vyřešit. Nakonek se ukázalo, že stačilo uvědomit si důležitou věc, a to, že téměř každý programátorský problém, který řešíme, již někdo před námi řešil. Dokonce i v případě robota Karla. Tudíž jsem přistoupil k hledání řešení přímo v ukázkových úlohách Karla ([1]) a ejhle, poslední, tedy 11. příklad, je přesně to, co jsem hledal. Teď už zbývá jen všechno pochopit a vysvětlit:

2 Zdrojový kód

Zdrojový kód k tomuto problému je přiložen v souboru *Reseni.txt* v kódování UTF-8 a byl vytvořen <http://ksi.fi.muni.cz/karel/> osobně.

3 Popis řešení

Program se spouští funkcí *ZBLUDISTE*.

Program je založen na rekurzivním volání funkce, která zjišťuje možnosti cesty z aktuálního pole. Celou cestu si Karel označuje značkami. Jakmile robot narazí na pole ze značkou, nebo na slepou uličku, vynořuje se z rekurze až do takového pole, z kterého vede cesta jinam, než kudy se původně vydal. Pokud cesta k cíli existuje, tento algoritmus zajistí její nalezení.

Následuje kompetní slovní popis funkce programu.

3.1 Pomocné funkce

1. Funkce *POLOZ2*

zajišťuje položení 2 značek.

2. Funkce *Z5*

provede jeden krok zpět a otočí robota do původního směru.

3. Funkce *CELEM – VZAD*

otočí robota o 180°.

4. Funkce *VPRAVO – VBOK*

otočí robota doprava.

3.2 Funkce *ZJISTIMOZNOSTI*



Jedná se o hlavní funkci, která zajišťuje hledání správné cesty až k cíli.

Tato funkce je založena na rekurzivním volání sama sebe.

Jak už bylo popsáno, pokud je na aktuálním poli značka, tak je provedeno vynoření z rekurze. To zajišťuje první podmínka "KDYŽ JE ZNAČKA", která v případě nalezení značky provede krok o jedno pole zpět zapomocí funkce "Z5".

Pokud se na aktuálním poli značka nenachází, dojde k označení tohoto pole 2-mi značkami a to z toho důvodu, že pokud by v budoucnu byla jedna značka sebrána rekurzivním vynořováním, aby na poli zůstala ještě jedna značka, která zajistí, že je cesta označena jako nevedoucí k cíli.

Následuje podmínka pro "DOMOV", která zajistí ukončení programu v případě nalezení domova.

Pokud nejsme v domově, provedeme kontrolu všech 4 směrů na přítomnost zdí. A to takovým způsobem, abychom na konci tohoto cyklu v případě přítomnosti zdí ve všech směrech kromě toho, ze kterého jsme přišli, skočili otočení právě na tento směr - tj. případ "slepá ulička". Pokud nejsme ve slepé uličce a v cyklu je vyhodnocena podmínka "KDYŽ NENÍ ZEĎ" kladně, dojde k rekurzivnímu volání sama sebe a celý výše i níže zmíněný algoritmus se opakuje.

Pokud jsme ve slepé uličce, dojde ke zvednutí jedné značky (tím je označena špatná cesta), ke kroku o jedno pole zpět a k otočení čelem-vzad. To z toho důvodu, aby Karel nabyl správného směru, jakmile se vynoří z rekurze zpět do této funkce.

Je důležité si uvědomit, že při nalezení cíle je algoritmus ukončen funkcí *STOP* a to zajistí, že nedojde k vynořování z rekurze, které by sebralo značky.

V ideálním případě tato funkce úspěšně nalezne cíl.

3.3 Funkce *ZBLUDISTE*



```
ZBLUDISTE
POLOZ2
KDYŽ JE DOMOV
STOP
KONEC, JINAK
OPAKUJ 4-KRÁT
KDYŽ NENÍ ZEĎ
KROK
ZJISTIMOZNOSTI
KONEC
VLEVO-VBOK
KONEC
ZVEDNI
KONEC
KONEC
```

Ale protože ideální případ nenastává vždy, je potřeba funkci *ZJISTIMOZNOSTI* zapouzdřit. To řeší tato funkce.

Funkce zajišťuje prvotní úkony algoritmu, tj. položení dvou prvních značek, ověření domova a nalezení prvního průchozího směru. Také zajišťuje následující funkcionalitu: pokud by někdo Karla ohraničil hned na začátku ze všech 4 stran, tak algoritmus ihned skončí - bez volání funkce *ZJISTIMOZNOSTI*.

4 Závěr

Cesta pro robota Alfonze je vyznačena konkrétním počtem značek: 2 značkami. Cesta na pole s jakýmkoliv jiným počtem značek nevede do cíle.

Pochopení celého algoritmu mi nepřišlo příliš složité, nýbrž kdybych ho měl vymýšlet sám, trvalo by mi to asi hodně dlouho.

Toto řešení je téměř přesnou kopií zdrojového kódu ze stránky příkladů k robotu Karlovi [1].

Reference

- [1] KSI *Příklady k vývojovému prostředí robota Karla*
<http://ksi.fi.muni.cz/karel/priklady.html>