**Navazování spojení**

**Řešení problému**

Potřebujeme přidat co nejméně hran tak, abychom se dostali všude.

**Zpracování inputu**

Abychom neprocházeli zbytečně moc možností vezmeme si z inputu jen to, co potřebujeme. Chceme získat páry vrcholů, Hlavu (H) a Ocas (T), pro které platí, že z hlavy se dostaneme do Ocasu, do hlavy nesměřuje žádný jiný vrchol a z Ocasu nejde žádná hrana v případě, že se nejedná o samostatný vrchol či cyklus.

V mém popisu využívám číslované vrcholy, nemusí to být čísla, může to být absolutně cokoliv, čísla využívám pro jednodušší vysvětlení.

Tyto páry jsou ku příkladu v ukázkovém grafu: 1H-4T, 6H-4T, 6H-7T. V případě, že jsme narazili na cyklus tak se nic nemění a bude to fungovat stejně.

U rozšíření: 1H-4T, 8H-T, 7H-5T, v tomto příkladě máme cyklus.

**Co budeme využívat?**

Vytvoříme si list na Hlavy (HL), Ocasy (TL), list Unikátních Párů hlav a ocasů (SL) a slovník Vrcholů (VL). HL a TL budou držet čísla vrcholů, SVL drží vždy 2 čísla, indexy k listu hlav a ocasů.

**Jak zjistíme Hlavy a Ocasy?**

Začneme na jakémkoliv vrcholu, vždy se podíváme, kam můžeme jít z vrcholu a z jakých vrcholů můžeme jít do vrcholu. Vždy, když začínáme na prvním vrcholu bez toho abychom se do něj dostali pokračováním, vytvoříme nové 2 čísla „indexy“, HI „Head index“ a TI „Tail Index“, které budou současné délky HL a TL. Vždy na každém nově objeveném vrcholu provedeme „**Objevení Vrcholu**“ s drženými indexy. Po Objevení vrcholu se vždy prvně podíváme, kam všude můžeme jít, přednost ale vždy má především cesta zpátky (hrana směrující do našeho současného vrcholu). Pokaždé provedeme „**Uložení vrcholu**“. Nezačínáme na již prozkoumaných vrcholech, pokud jsme prozkoumali všechny vrcholy uděláme z SL list reálných párů, a ne jenom referencí, ty vrátíme a tím končíme se „**Zpracováním inputu**“.

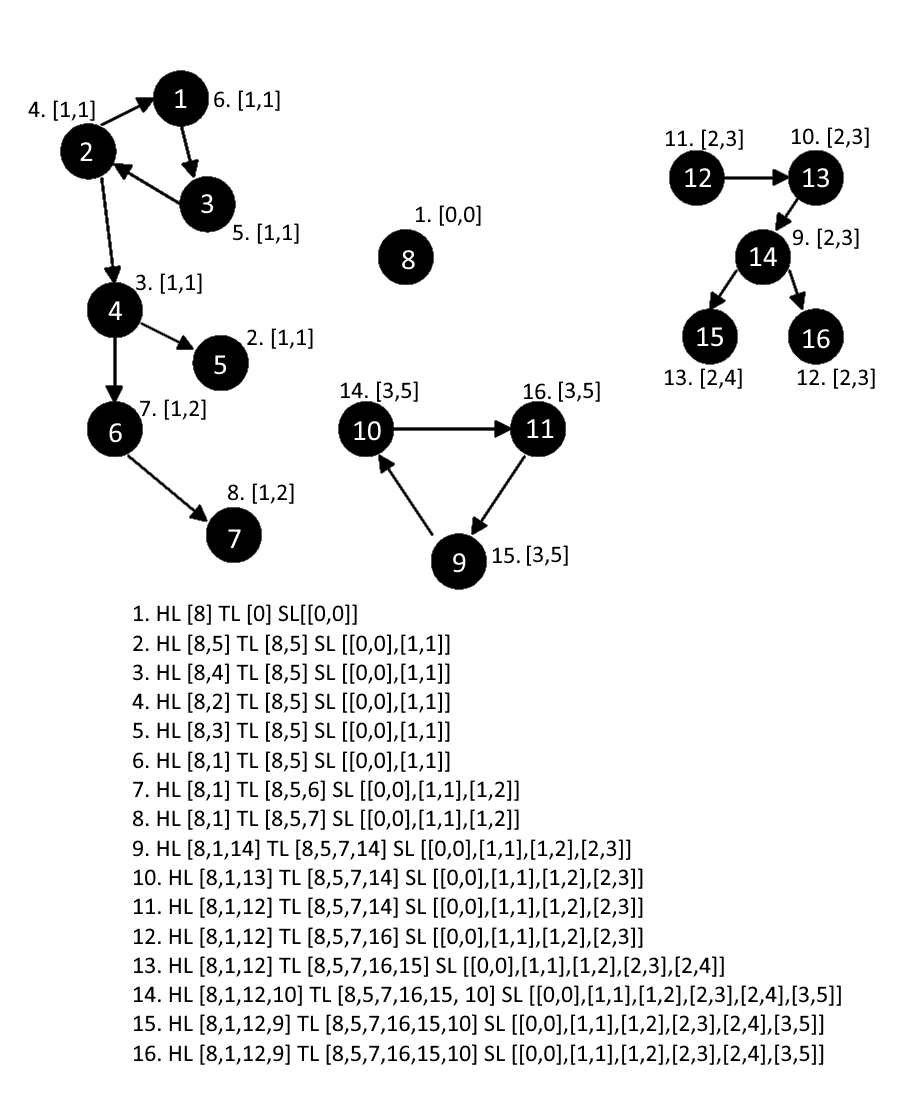
**Objevení vrcholu**

V případě, že nemáme ještě vytvořený čísla v HL a TL vytvoří jak Hlavu, tak Ocas jako současný vrchol, v případě že už jsou vytvořená čísla na indexech, přepíše se číslo v TL na pozici TI na číslo současně objeveného vrcholu. Dále, jestli objevíme vrchol, který již byl objevený, tak si vezmeme a) (jdeme opačným směrem hran) jeho hlavu, b) (jdeme po směru hran) ocas a pokud se to stane dál nehledáme, pokud není jiná hrana ze současného vrcholu, která směřuje jinam.

**Uložení vrcholu**

Přidá do SL, „listu Unikátních Párů“, pokud tam ještě není.

Tady je ukázka, jak najít hlavy a ocasy, tato ukázka pokrývá i rozšíření. U každého vrcholu je napsané číslo, které reprezentuje kolikátý krok jsme ho prozkoumali a dole je napsané, jak to vypadalo po provedení daného kroku.



Z této ukázky dostaneme 6 párů: 8H-8T, 1H-5T, 1H-7T, 12H-16T, 12H-15T, 9H-10T.

**Úprava zpracovaného inputu**

V moment, co máme tyto páry si odstraníme duplikáty, tím je myšleno že pokud Hlava i Ocas již existuje jako Hlava/Ocas u jiného páru tak ten pár vymažeme.

Například u ukázkového grafu odstraníme 6H-4T, protože 6H se objevila a 4T se také objevil v jiném páru, takže tento pár odstraníme a dostaneme 1H-4T, 6H-7T. Jak to ale uděláme? abychom je vymazali tak si nejdříve projdeme všechny páry a uděláme si dva slovníky, kde uložíme kolik je jakých vrcholů v Hlavách a v Ocasech. Pokud délka slovníků odpovídá počtu vrcholů, nic neodstraňujeme, ale pokud ano zkontrolujeme, zda jsou duplikáty a odstraníme duplikáty, pokud ano.

U rozšíření žádné duplikáty nemáme, takže nám zůstalo: 1H-4T, 8H-8T, 7H-5T.

**Vytvoření hran**

Teď už jenom finální krok, propojíme Ocas směrem do hlavy, která je o jeden pár dále, pokud je jen jeden pár tak to bude samo se sebou a pokud je to poslední pár, propojíme s první Hlavou, u ukázkového grafu jsme měli 1H-4T a 6H-7T a z toho se nám stanou hrany 1H <- 7T a 6H <- 4T. Pokud nastane například 1H <- 1T, tak se nepřidá žádná hrana, protože nechceme dát hranu samu k sobě.

U rozšíření získáme hrany 1H <- 5T, 8H <- 4T, 7H <- 8T.

Ukázkový graf Řešení ukázkového grafu

A diagram of a path

Description automatically generated A diagram of a diagram

Description automatically generated

(Rozšíření) Ukázková graf (Rozšíření) Řešení ukázkového grafu

A diagram of a triangle

Description automatically generated A diagram of a triangle with arrows and numbers

Description automatically generated

**Zdůvodnění řešení**

Toto řešení bude vždy funkční, protože jediné vrcholy, které potřebujeme propojit jsou ty co nemají jak vstupní, tak odchozí hrany (Nepočítaje cyklus), proto nám stačí znát Hlavy a Ocasy a pokud propojíme každý Ocas s následující Hlavou, a to nám udělá cyklus.

Díky tomu, že odstraňujeme duplikáty to bude i řešení s nejméně možnými hranami, protože nechceme propojit tyto hrany více než jednou, tudíž řešení s méně hranami nejde dosáhnout.

**Časová komplexita**

Získání Hlav a Ocasů z inputu je časová komplexita O(n). Vyfiltrování duplikátů je O(k) kde k (0 <= k <= 2n) představuje počet Hlav a Ocasů, takže časová komplexita této funkce je O(n). Poslední část, vytvoření hran je O(p) kde p (0 <= p <= n/2), takže časová komplexita funkce je O(n).

Tyto funkce jdou sekvenčně, takže celková časová komplexita tohoto algoritmu je O(n).