Náhodné zamíchání pole – Fisher-Yates Shuffle

# Úvodní informace

Úloha má za cíl demonstrovat jednu z nejznámějších a nejpoužívanějších metod sloužící k náhodnému zamíchání pole známou jako Fisher-Yates Shuffle. Tento způsob je velice jednoduchý k pochopení i naprogramování. Proto se hodí k procvičení cyklů.

# Cíle úlohy

* Procvičení:
  + Cykly
* Získání (pseudo)náhodného čísla ze zadaného intervalu
* Hledání relevantních informací v internetových zdrojích
* Získání poznatků o (pseudo)náhodných číslech

# Náročnost

* 1 vyučovací hodina
* Obtížnost:

# Prerekvizity

* Cykly

# Metodika výuky

Síla úlohy spočívá v její triviálnosti. Kromě cyklů nejsou potřeba žádné jiné znalosti.

Zadání:

Napište program, který náhodně přehází prvky v poli pomocí Fisher-Yates Shuffle. Jako vstup programu uvažujte velikost pole. Hodnota bude rovna desetinásobku klíče prvku (tzn. na klíči 3 bude hodnota 30). Výsledek vypište ve tvaru klíč – hodnota, a to před i po zamíchání.

Vstup: 6

Výstup (příklad):

0 - 0

1 - 10

2 - 20

3 - 30

4 - 40

5 – 50

-----

0 - 30

1 - 40

2 - 10

3 - 0

4 - 20

5 - 50

Pro vysvětlení použité metody je možné použít přiložený pomocný soubor fisher-yates-shuffle.png, kde je pomocí obrázku popsáno, jak algoritmus funguje. Je možné takový rozbor provést na tabuli. Vygenerujeme náhodné číslo ze stále zkracujícího se intervalu (maximum se rovná indexu nejvyššího prvku, minimum 0) a nejvyšší prvek prohodíme s náhodně vybraným prvkem (prohodíme jejich hodnoty), zároveň nejvyšší prvek zneaktivníme (proběhne výše zmíněné snížení intervalu). Takto pokračujeme, dokud nám nezůstane jeden samotný prvek. Specifický případ k řešení je v případě, že vylosujeme nejvyšší prvek. Existují dva způsoby řešení (které mají stejný výsledek). Můžeme prvek prohodit sám se sebou, nebo prohození vynechat. Tato speciální situace je naznačena v typovém příkladu v pomocném slidu.

Obsah obrázku stůl

Popis byl vytvořen automaticky

Obrázek - Vysvětlené zamíchání pole s typovým příkladem

Samotný program je koncipován velice jednoduše. Nachází se pod názvem fisher-yates-shuffle v příslušném programovacím jazyce. Budeme potřebovat tři cykly. V prvním naplníme pole hodnotami a zároveň vypíšeme v požadovaném tvaru, v druhém provedeme zamíchání pole a ve třetím vypíšeme zamíchané pole. Během programování by každý měl pracovat samostatně.

Jediné, co může působit problém, je neznalost vytvoření (pseudo)náhodného čísla z nějakého intervalu. Jelikož je řešení v jednotlivých programovacích jazycích odlišné, budeme hned na začátku úlohy studenty buď instruovat (řekneme/napíšeme řešení tvorby náhodného čísla), nebo je necháme, aby si tyto informace zjistili z otevřených zdrojů (nikoliv od souseda). Dle dovedností skupiny by se autor spíše přikláněl k druhé možnosti řešení.

Za zmínku stojí, že v jazyce C++ je nutné přidat náhodný seed. V ukázce je použit příkaz srand(time(NULL)), který za seed považuje čas v sekundách od Epoch time (unixový čas). Pokud pracujeme v tomto jazyce, doporučuje se demonstrovat zadání náhodného čísla a poté několikrát spustit kompilovaný exe program, aby bylo vidět, že výsledné pole je stále randomizováno.

Po úspěšném vyřešení je možné pokračovat v diskusi nad tímto příkladem. Vhodné je v ní vysvětlit náhodná a pseudonáhodná čísla v souvislosti s informatikou.

### Otázky do diskuse

1. Jedná se doopravdy o zcela náhodné zamíchání? Pokud ne, jak se těmto náhodným číslům říká (najděte na internetu)?
2. Co je to seed a k čemu slouží (najděte na internetu)?
3. Dokážete vymyslet, kde se taková pseudonáhodná čísla dají uplatnit?
4. Proč nemůže počítač generovat zcela náhodná čísla?

### Možné problémy

* Neočekávají se problémy, které by studenti nebyli schopni sami vyřešit.