Úkol 1: Řadící algoritmy

**Šimon Bučka**

Třídící algoritmy jsou souborem algoritmických technik, které mají za cíl uspořádat množinu dat do specifického pořadí, obvykle vzestupně nebo sestupně. Tyto algoritmy se vyskytují v mnoha variantách, přičemž každá z nich má své vlastní výhody a nevýhody v závislosti na charakteristikách vstupních dat a požadavcích na výkonnost. Příklady populárních třídících algoritmů zahrnují Bubble Sort, Merge Sort, Quick Sort a Heap Sort

# BogoSort:

Je to velmi neefektivní a nepoužitelný algoritmus řazení. Pracuje tak, že náhodně zamíchá prvky, dokud seznam náhodou nevyjde seřazený.

## Časová složitost:

### **Nejlepší případ**:

O(n)

### **Průměrný případ**:

O((n+1)!)

### **Nejhorší případ**:

O(∞)

# Insertion Sort:

Insertion Sort je jednoduchý a efektivní třídící algoritmus, který porovnává prvky v poli a postupně vytváří seřazené pole. Algoritmus pracuje tak, že opakovaně vezme nezařazený prvek a umístí ho na správné místo v seřazené části pole.

## Časová složitost:

### **Nejlepší případ:**

O(n)

### **Průměrný případ**:

O(n^2)

### **Nejhorší případ:**

O(n^2)

# Selection Sort:

Selection Sort (Výběrové třídění) je jednoduchý třídící algoritmus, který postupně vybírá nejmenší (nebo největší) prvek ze seznamu a umisťuje ho na správné místo v seřazené části seznam

## Časová složitost:

### **Nejlepší případ:**

O(n^2)

### **Průměrný případ:**

O(n^2)

### **Nejhorší případ:**

O(n^2)

# Pancake Sort:

Pancake Sort (Palačinkové třídění) je netradiční algoritmus pro třídění prvků v seznamu. Tento algoritmus je inspirovaný fyzickým otáčením palačinek, jak se otáčejí na pánvi. Hlavním cílem algoritmu je převrátit prvky v seznamu tak, aby byly v požadovaném pořadí.

## Časová složitost:

### **Nejlepší případ:**

O(n)

### **Průměrný případ:**

O(n^2)

### **Nejhorší případ:**

O(n^2)

# Shell Sort:

Shell Sort je algoritmus řazení, který rozděluje seznam na menší podseznamy a poté tyto podseznamy řadí pomocí vložení. Postupně zmenšuje mezery mezi prvky a opakuje tento proces, dokud není seznam seřazen.

## Časová složitost:

### **Nejlepší případ:**

O (n log(n))

### **Průměrný případ:**

O (n log(n))

### **Nejhorší případ:**

O (n^2)