

# AKADEMIA NAUK STOSOWANYCH W NOWYM SĄCZU

Wydział Nauk Inżynierskich  
Katedra Informatyki

## DOKUMENTACJA PROJEKTOWA ZAAWANSOWANE PROGRAMOWANIE

### **Drzewo BTS**

Autor:  
Mateusz Smaga  
Kamil Trzópek

Prowadzący:  
mgr inż. Dawid Kotlarski

Nowy Sącz 2024

## **Spis treści**

<b>1. Ogólne określenie wymagań</b>	<b>3</b>
<b>2. Analiza problemu</b>	<b>4</b>
<b>3. Projektowanie</b>	<b>6</b>
<b>4. Implementacja</b>	<b>7</b>
<b>5. Wnioski</b>	<b>8</b>
<b>Literatura</b>	<b>9</b>
<b>Spis rysunków</b>	<b>10</b>
<b>Spis listingów</b>	<b>11</b>

## 1. Ogólne określenie wymagań

Zadanie w projekcie jest napisanie programu przedstawiające strukturę „drzewa BST” działającego na sterce w języku C++. Drzewo winno być zaimplementowana w klasie. Funkcjonalność (metod) drzewa:

- - Dodaj element,
- - Usuń element,
- - Usuń całe drzewo,
- - Szukaj drogi do podanego elementu,
- - Wyświetl drzewo graficznie na ekranie, użytkownik wybiera metodę podczas wyświetlania (metody preorder, inorder, postorder) [oprogramuj wszystkie trzy],
- - Zapis do pliku tekstowego wygenerowanego drzewa,

W drugiej klasie należy zaimplementować (metody) zapis do pliku i odczyt z pliku utworzonego drzewa BTS (plik musi być zapisany binarnie). Funkcja main powinna wyświetlać menu z opcjami drzewa oraz odczytu i zapisu pliku. Program czeka na wybranie opcji. Wszystkie utworzone klasy mają być zaimplementowane w oddzielnych plikach. Funkcja main także powinna być w osobnym pliku.

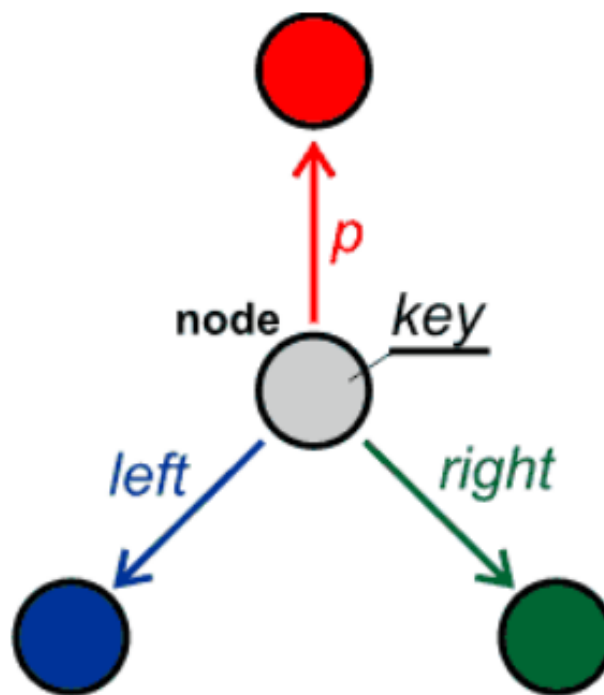
Przy oddawaniu projektu należy zaprezentować:

- - Co najmniej 5 commit’ów (każdej osoby),
- - Najpierw jedna osoba tworzy gałąź i po kilku comitach ją scala. Po scaleniu druga osoba
- - tworzy gałąź i po kilku comitach ją scala do głównej gałęzi ,
- - Obie osoby w grupie mają utworzyć nowe gałęzie (w jednym punkcie obie osoby zaczynają pracę równoległą), a po kilku comitach wykonują scalenie do głównej gałęzi,
- - Co najmniej 6 konfliktów, które należy rozwiązać (3 jedna osoba, 3 druga osoba) przy scalaniu gałęzi (w wcześniejszych punktach),

## 2. Analiza problemu

Drzewa poszukiwań binarnych - BST

Drzewo poszukiwań binarnych BST (ang. Binary Search Tree) jest dynamiczną strukturą danych zbudowaną z węzłów (ang. node). Każdy węzeł może posiadać dwóch potomków (left - lewy i right - prawy) oraz jednego przodka (p - parent)(2.1). Z każdym węzłem dodatkowo związany jest klucz (key) (link do strony z której pochodzą grafiki, oraz informacje [1])

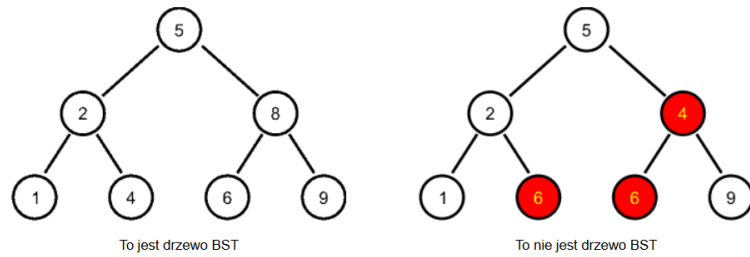


Rys. 2.1. node schemat

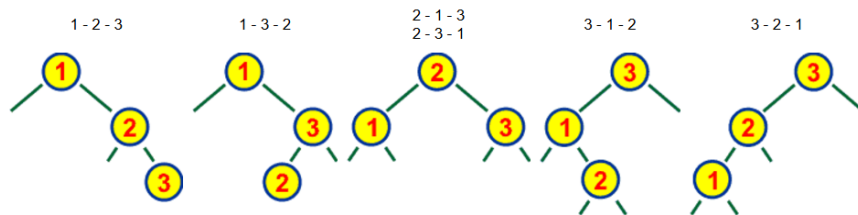
Dla każdego węzła w drzewie BST zachodzą następujące własności 2.2:

- Wartości kluczy węzłów leżących w prawym poddrzewie węzła są większe lub równe wartości klucza danego węzła.
- Wartości kluczy węzłów leżących w lewym poddrzewie węzła są mniejsze lub równe wartości klucza danego węzła.

Ważnym jest że zależności od kolejności wprowadzania danych do drzewa BST mogą powstać różne konfiguracje węzłów 2.3.



Rys. 2.2. node schemat



Rys. 2.3. node różnica

```

1
2
3 int BSTminkey(BSTNode * root)
4 {
5     BSTNode * x = root;
6
7     while((x->left) x = x->left;
8
9     return x->key;
10 }
```

Listing 1. BTS

wyszukiwanie rozpoczynamy od korzenia drzewa BST, przechodzimy przez poszczególne węzły drzewa BST zaawsze wybierając lewo (listninig:1)

### 3. Projektowanie

## **4. Implementacja**

## 5. Wnioski



## Bibliografia

- [1] *Strona internetowa BTS*. URL: [https://eduinf.waw.pl/inf/utills/002\\_roz/mp001.php](https://eduinf.waw.pl/inf/utills/002_roz/mp001.php).

## Spis rysunków

2.1. node schemat . . . . .	4
2.2. node schemat . . . . .	5
2.3. node różnica . . . . .	5

## Spis listingów

1.	BTS . . . . .	5
----	---------------	---