

Zestaw 2B

Ocena:

ocena 3.0	Ocena 4.0	Ocena 5.0
Zad. 1 lub Zad. 2 lub Zad. 3	Zad. 1, Zad. 2 lub Zad. 1, Zad. 3 lub Zad. 2, Zad. 3	Zad. 1, Zad. 2, Zad. 3

Ramówka programu na następnej stronie. Proszę z niej skorzystać. W razie potrzeby proszę odpowiednio zmodyfikować listy argumentów funkcji i typy wartości zwracanych przez funkcje.

Zadanie 1:

Zaimplementuj operacje enqueue i dequeue dla kolejki FIFO przechowującej znaki. Dowolna reprezentacja kolejki. Należy umożliwić także wyświetlenie stanu kolejki. Implementacja optymalna.

Zadanie 2:

Zaimplementuj operację wyszukiwania w drzewie BST, przechowującym znaki, węzła przechowującego znak o podanej wartości. Wynikiem ma być indeks węzła lub -1 jeśli nie istnieje. Znak '0' powinien reprezentować NULL. Reprezentacja tablicowa. Implementacja optymalna.

Zadanie 3:

Zaimplementuj operację dodawania, z zachowaniem uporządkowania, elementu o podanej wartości klucza do uporządkowanej niemalejąco listy jednokierunkowej. Lista przechowuje znaki. Należy także umożliwić wyświetlenie stanu listy przed i po dodaniu elementu. Implementacja optymalna.

Ramówka program 2B:

```
#include <cstdlib>
#include <iostream>

using namespace std;

//Zadanie 1

void enqueue(char z){}
char dequeue (){}
void print(){} //wydruk stanu kolejki
void zadanie1(){
cout<<"Zadanie 1"<<endl;
enqueue('a'); print(); enqueue('e'); print();enqueue('i'); print();enqueue('n'); print();enqueue('e'); print();
cout<< dequeue();print(); cout<< dequeue();print();
enqueue('r'); print();enqueue('x'); print();
cout<< dequeue(); print(); cout<< dequeue(); print(); cout<< dequeue(); print();
cout<< dequeue(); print(); cout<< dequeue(); print(); cout<< dequeue(); print(); cout<< dequeue(); print();
cout<<"Zadanie 1 - koniec"<<endl;
}

//Zadanie 2

int search(char bst[], char s){}
void zadanie2(){
cout<<"Zadanie 2"<<endl;
//drzewo BST użyte w zadaniu powinno być wynikiem dodawania kolejno następujących znaków: l d h p j a o
char bst[] ={}; //należy w odpowiedni sposób zainicjalizować tablicę podanymi znakami
cout<<search(bst,'j'); cout<<search(bst,'h'); cout<<search(bst,'p'); cout<<search(bst,'z');
cout<<"Zadanie 2 - koniec"<<endl;
}

//Zadanie 3

void dodaj(char z){}
void printL(){}
void zadanie3(){
cout<<"Zadanie 3"<<endl;
dodaj('d'); dodaj('j'); dodaj('k'); dodaj('h'); dodaj('a'); dodaj('a'); printL();
cout<<"Zadanie 3 - koniec"<<endl;
}

int main(int argc, char *argv[])
{
    zadanie1();
    zadanie2();
    zadanie3();

    system("PAUSE");
    return EXIT_SUCCESS;
}
```