Ministerul Educaţiei al Republicii Moldova

Universitatea Tehnică a Moldovei

Catedra Informatică Aplicată

**RAPORT**

Lucrarea de laborator nr.1

Programarea Orientata pe Obiecte

A efectuat:

st. gr. C-171 D. Melniciuc

A verificat:

dr., conf.univ. M. Oșovschi

Chişinău 2018

**Tema:** Reprezentarea tipurilor de date ale limbajului C++ în memoria calculatorului. Operatorii limbajului C++. Construcţiile elementare ale limbajului C++ (instrucţiunile ***for, while,******do-while, if-else,******switch-break,******goto*).** Tipuri de date recursive, operaţii asupra listelor, arborilor. Construirea şi elaborarea programelor recursive. Fişierele.

**Scopul lucrării**: familiarizarea studenţilor cu reprezentarea tipurilor de date ale limbajului C++ în memoria calculatorului, operatorii limbajului C++, construcţiile elementare ale limbajului C++ (instrucţiunile ***for, while,******do-while, if-else,******switch-break,******goto*),** tipuri de date recursive, operaţii asupra listelor, arborilor, construirea şi elaborarea programelor recursive, lucrul cu fişierele.

**Varianta 4:**

**a.** Scrieţiun program care converteşte un întreg într-un număr hexazecimal.

**b.** Scrieţi un program care permite crearea unui arbore binar şi traversarea lui în inordine, preordine, postordine

***Codul programul I:***

*#include <iostream>*

*#include <limits>*

*using namespace std;*

*int main(){*

*int val;*

*cout << "INT\_MAX: " << std::numeric\_limits<int>::max() <<endl;*

*cout << "introduceti valoare negativa p/u a iesi din program\n\n";*

*for(;;){*

*cout << "introduceti numar\n>> ";*

*cin >> val;*

*if (val < 0) exit(1);*

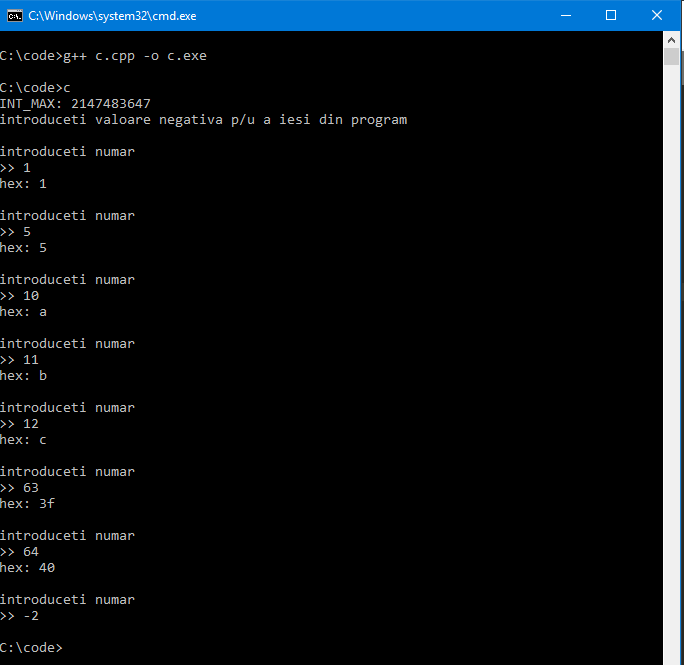
*cout << hex << "hex: "<<val << "\n\n";*

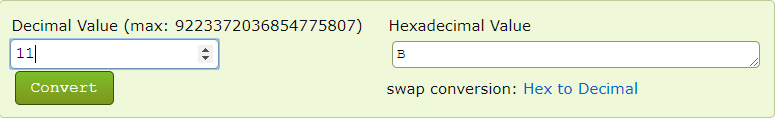
*}*

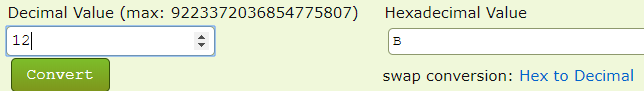
*return 0;*

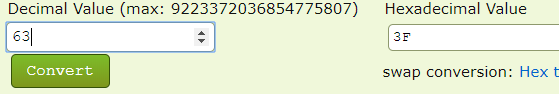
*}*

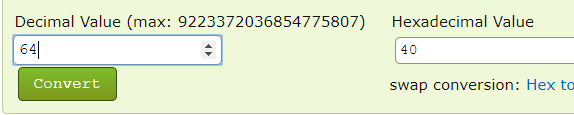
***ScreenShoturi:***











***Codul programul II:***

*# include <iostream>*

*# include <cstdlib>*

*using namespace std;*

*struct nod*

*{*

*char name[40];*

*struct nod \*left,\*right;*

*}nod;*

*struct nod \*root=NULL;*

*struct nod\* creat\_rsd(struct nod \* parent)*

*{ struct nod \*c;*

*int f;*

*c=(struct nod\*)malloc( sizeof(\*c) );*

*if(!c) return NULL;*

*cout <<"\nIntroduceti datele\n";*

*cin >> c->name;*

*cout <<"\nDoriti sa creati copilul sting al nodului ["<< c->name <<"] (1/0)?: " ;*

*cin >>f;*

*if(f) c->left=creat\_rsd(c);*

*else c->left=NULL;*

*cout <<"\nDoriti sa creati copilul drept al nodului [" <<c->name <<"] (1/0)?: " ;*

*fflush(stdin); cin >>f;*

*if(f) c->right=creat\_rsd(c);*

*else c->right=NULL;*

*return c;*

*}*

*void show\_rsd(struct nod \*c)*

*{*

*if(!c) return;*

*cout <<c->name<<" ";*

*show\_rsd(c->left);*

*show\_rsd(c->right);*

*}*

*void show\_rds(struct nod \*c)*

*{*

*if(!c)*

*return;*

*cout <<c->name<<" ";*

*show\_rds(c->right);*

*show\_rds(c->left);*

*}*

*void show\_srd(struct nod \*c)*

*{*

*if(!c) return;*

*show\_srd(c->left);*

*cout <<c->name<<" ";*

*show\_srd(c->right);*

*}*

*void show\_drs(struct nod \*c){*

*if(!c){*

*return;*

*}*

*show\_drs(c->right);*

*cout <<c->name<<" ";*

*show\_drs(c->left);*

*}*

*void show\_sdr(struct nod \*c)*

*{*

*if(!c)*

*return;*

*show\_sdr(c->left);*

*show\_sdr(c->right);*

*cout <<c->name<<" ";*

*}*

*void show\_dsr(struct nod \*c)*

*{*

*if(!c)*

*return;*

*show\_dsr(c->right);*

*show\_dsr(c->left);*

*cout <<c->name<<" ";*

*}*

*void freemem\_rsd(struct nod \*c)*

*{*

*if(!c) return;*

*freemem\_rsd(c->left);*

*freemem\_rsd(c->right);*

*free(c);*

*}*

*int height\_rsd(struct nod \*c)*

*{*

*int l=0,r=0;*

*if(!c) return -1;*

*l=1+height\_rsd(c->left);*

*r=1+height\_rsd(c->right);*

*if(l>r) return l;*

*else return r;*

*}*

*int main()*

*{*

*char fname[100];*

*int gxr = 0;*

*do{*

*start:*

*system("cls");*

*cout<<"[1] Creare\n";*

*cout<<"[2] Afisarea parcurgerilor\n";*

*cout<<"[0] Exit\n";*

*cout <<">> ";*

*cin >>gxr;*

*cout<<" ";*

*switch (gxr)*

*{*

*case 1: {*

*root = creat\_rsd(NULL);*

*system("PAUSE");*

*break;*

*}*

*case 2: {*

*cout <<"RSD\n";*

*show\_rsd(root);*

*cout << "\n";*

*cout <<"RDS\n";*

*show\_rds(root);*

*cout << "\n";*

*cout <<"SRD \n";*

*show\_srd(root);*

*cout << "\n";*

*cout <<"DRS \n";*

*show\_drs(root);*

*cout << "\n";*

*cout <<"SDR \n";*

*show\_sdr(root);*

*cout << "\n";*

*cout <<"DSR \n";*

*show\_dsr(root);*

*cout << "\n";*

*system("PAUSE");*

*break;*

*}*

*case 0:{*

*finish:*

*system("pause");*

*system("cls");*

*return 0;*

*}*

*default:{*

*int zx = 0;*

*cout <<"[-] Gresit\n";*

*cout<<"[1] Continuam programul\n[2] Iesim\n>> ";*

*cin >> zx;*

*if (zx == 1) goto start;*

*else goto finish;*

*system("cls");*

*break;*

*}*

*}*

*}*

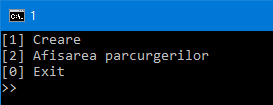
*while(gxr!=0);*

*system("pause");*

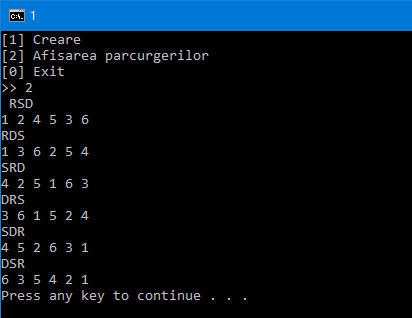
*return 0;*

*}*

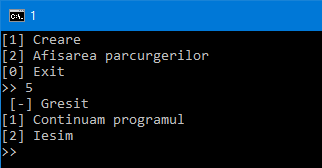
***ScreenShoturi:***

***Meniu:***

***Afisarile parcurgerillor:***



***Introducerea unei optiuni gresite:***



***Concluzie:***

*Dupa efectuarea laborotorului am dobintid experienta in limbajul de programere C++ facind exercitiile propuse*