Funkcije - Zadaci za vežbanje

1.

Napisati program za konverziju binarnog u oktalni brojni sistem korišćenjem funkcije.

2.

Napisati program koji proverava da li je uneti broj palindrom, korišćenjem funkcije.

3.

Napisati program koji izračunava eksponent broja korišćenjem rekurzivne funkcije.

U nastavku dokumenta nalaze se rešenja ovih zadataka.

Rešenja pogledati ako samostalan rad ne daje rezultate.

Rešenja

1.

```
#include <iostream>
#include <cmath>

using namespace std;

int b2o(long long);

int main()
{
    long long binaryNumber;

    cout << "Uneti binarni broj: ";
    cin >> binaryNumber;

    cout <<"[" << binaryNumber << "](2) = [" << b2o(binaryNumber) << "](8)\n";

    return 0;
}

int b2o(long long binaryNumber)
{
    int octalNumber = 0, decimalNumber = 0, i = 0;

    while(binaryNumber != 0)
    {
        decimalNumber += (binaryNumber%10) * pow(2,i);
        ++i;
        binaryNumber/=10;
    }

    i = 1;

    while (decimalNumber != 0)
    {
        octalNumber += (decimalNumber % 8) * i;
        decimalNumber /= 8;
        i *= 10;
    }

    return octalNumber;
}
```

```
Uneti binarni broj: 110011
[110011](2) = [63](8)

Process returned 0
```

2.

```
#include<iostream>
using namespace std;

void palindrom(int);

int main()
{
        int n;
        cout<<"Unesite broj: ";
        cin>>n;
        palindrom(n);
}

void palindrom(int n)
{
        int n1,d,rn=0;
        n1=n;

        while(n>0)
        {
                d=n%10;
                rn=(rn*10)+d;
                n/=10;
        }

        if(n1==rn)
        cout<<"\nBroj ["<<n1<<"] jeste palindrom.\n";
        else
        cout<<"\nBroj ["<<n1<<"] nije palindrom.\n";

}
```

```
Unesite broj: 123321

Broj [123321] jeste palindrom.

Process returned 0
```

3.

```
#include <iostream>
using namespace std;

int myPow(int, int);

int main()
{
    int base, powerRaised, result;

    cout << "Unesite osnovu: ";
    cin >> base;
    cout << "Unesite eksponent: ";
    cin >> powerRaised;

    result = myPow(base, powerRaised);

    cout <<"["<< base << "^" << powerRaised << "] = " << result<<"\n";

    return 0;
}

int myPow(int base, int powerRaised)
{
    if (powerRaised != 0)
        return (base * myPow(base, powerRaised-1));
    else
        return 1;
}
```

```
Unesite osnovu: 2
Unesite eksponent: 5
[2^5] = 32

Process returned 0
```