

Увод в алгоритмите и програмирането

Домашна работа №4

Никола Николов, F106501

Задача 1: Да се състави алгоритъм (схемата на управление) и програма (в съответствие със схемата) за пресмятане по два начина на функцията $\sin(x)$ с точност 0.000001, както е дадена в лекциите и на семинар (развита в ред на Маклорен). Първи начин – с пресмятане на факториел (в знаменателя) и втори начин – без пресмятане на факториел. Обърнете внимание, че аргументът е въл в радиани. Прегледайте материала от семинара, схемите са дадени. Приложете към домашното код и снимки на екран с входа и изхода.

Решение:

$$\sin(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots = \sum_{i=1}^{\infty} (-1)^{i+1} \frac{x^{2i-1}}{(2i-1)!}$$

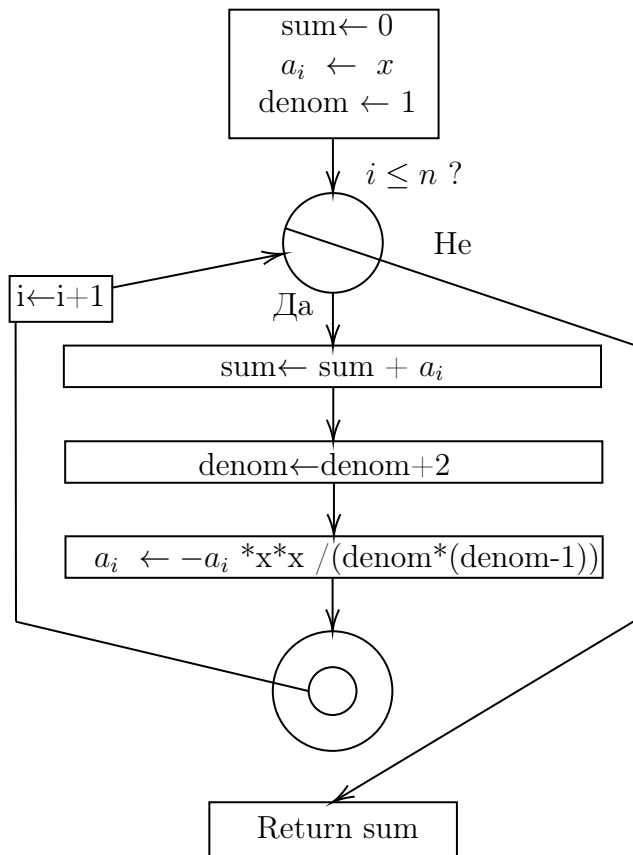
```
nn@NNPC:~/Programming/NBU/Algoritmi/Week4/HW4$ ./zad1
3.14 10
0.00159268
0.00159265
nn@NNPC:~/Programming/NBU/Algoritmi/Week4/HW4$ ./zad1
0.785 10
0.706825
0.706825
nn@NNPC:~/Programming/NBU/Algoritmi/Week4/HW4$ ./zad1
1.57 10
1
1
```

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

double sin(double x, int n);
unsigned long long int factorial(int n);
double sinFact(double x, int n);
```

```
int main()
{
    double x=0;
    int n=0;
    std::cin >> x >> n;
    std::cout << sin(x,n) << '\n';
    std::cout << sinFact(x,n)
                << '\n';

    return 0;
}
```



```

double sin(double x, int n)
{
    double sum=0;
    double member = x;
    int denom = 1;
    double tmp=sum;

    for(int i=1;i<=n;i++){
        sum += member;
        denom += 2;
        member = -member * x * x /
                    (denom * (denom-1));
        if(fabs(tmp-member)<=0.000001) break;
        tmp=member;
    }

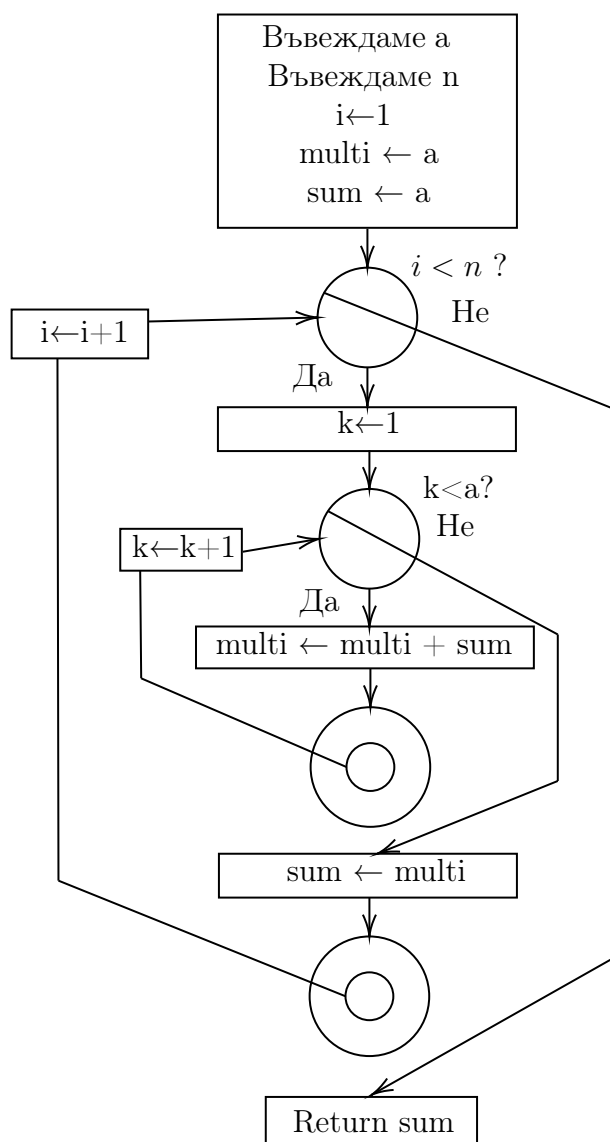
    return sum;
}

double sinFact(double x, int n)
{
    double sum=0;
    int sign = 1;
    double member = 0, tmp=0;
    for(int i=1;i<=n;i++){
        tmp = member;
        member = (sign*pow(x,2*i-1))/
                    (factorial(2*i-1));
        sum += member;
        sign = -sign;
        if(fabs(tmp-member)<=0.000001) break;
    }
    return sum;
}

unsigned long long int factorial(int n)
{
    unsigned long long int sum = 1;
    for(int i=1;i<=n;i++){
        sum *= i;
    }
    return sum;
}

```

Задача 2: Да се състави алгоритъм (схемата на управление) и програма (в съответствие със схемата) за повдигане на цяло положително число a на цяла положителна степен n , при използване само на адитивни операции. Прегледайте още веднъж материала от семинара. Там и схемите са дадени.



```

#include <iostream>

int power(int a, int n);

int main()
{
    int a=0 , n=0;
    std::cin >> a >> n;
    std::cout << power(a,n);

    return 0;
}

int power(int a, int n)
{
    int sum=a , multi=a;
    for(int i=1; i<n; i++){
        for(int k=1; k<a; k++){
            multi += sum;
        }
        sum = multi;
    }
    return sum;
}
  
```

Задача 3: *Предайте екселски файл с графика на траектория, като заместите в дадения към темата файл-модел вашите данни.*

Решение:

