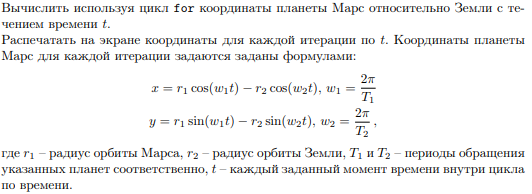
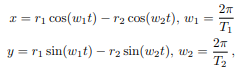
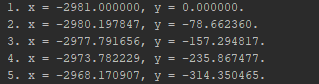
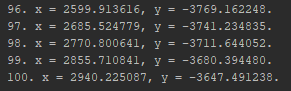
Лабораторная работа № 3. Организация циклов в Си

Задание 1

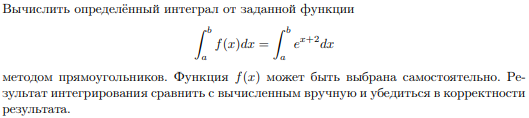
1. 
2. 

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| r1 | Радиус орбиты Марса | int |
| r2 | Радиус орбиты Земли | int |
| w1 | Угловая скорость Марса | double |
| w2 | Угловая скорость Земли | double |
| x | Координаты по x | double |
| y | Координаты по y | int |
| Phi | Смещение | double |
| t1 | Период обращения Марса | int |
| t2 | Период обращения Земли | int |
| t | Шаг по времени | int |

#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int t1 = 687, t2 = 365, r1 = 3390, r2 = 6371, t, Phi = 0;  
 double x, y, w1, w2;  
 w1 = 2 \* M\_PI / t1;  
 w2 = 2 \* M\_PI / t2;  
 for(t = 0; t <= 99; t++) {  
 x = r1 \* cos(w1 \* t + Phi) - r2 \* cos(w2 \* t + Phi);  
 y = r1 \* sin(w1 \* t + Phi) - r2 \* sin(w2 \* t + Phi);  
 printf("%d. x = %f, y = %f.\n", t + 1, x, y);  
 }  
 return 0;  
}

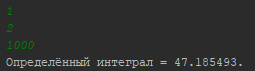
1.  

Задание 2

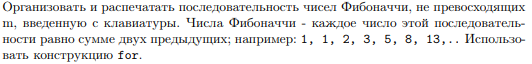
1. 
2. 

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| a | Левая граница интервала | double |
| b | Правая граница интервала | double |
| h | Шаг | double |
| S | Текущая площадь под графиком от a до a+n\*h | double |
| n | Кол-во разбиений | Int |
| I | Результат | double |
| x |  | double |

#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int n;  
 double S, I, a, b, h, x;  
 scanf("%lf%lf%d", &a, &b, &n);  
 S = 0;  
 h = (b - a) / n;  
 a = x;  
 for(x; x <= b - h; x += h) {  
 S = S + exp(x + 2);  
 }  
 I = S \* h;  
 printf("Определённый интеграл = %f.", I);  
}

1. 

Задание 3

1. 
2. .

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| m | Значение, вводимое пользователем | int |
| previous | Предыдущее значение | int |
| current | Текущее значение | int |
| fib | Значение fib | int |

#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int m, fib, current = 1, previous = 1;  
 printf("Введите число: ");  
 scanf("%d", &m);  
 printf("%d ", previous);  
 for(fib = previous + current; fib <= m; current = fib) {  
 fib = previous + current;  
 previous = current;  
 printf("%d ", previous);  
 }  
}

1. 

Задание 4

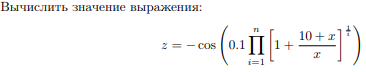
1. 
2. —

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| num | Значение, до которого нужно выводить натуральные числа | int |
| i | Счетчик цикла | int |

#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int i, num;  
 printf("Введите число: ");  
 scanf("%d", &num);  
 for(i = 1; i <= num; i++) {  
 printf("%d ", i);  
 }  
}

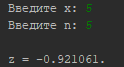
1. 

Задание 5

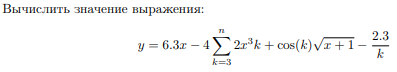
1. 
2. 

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| n | Количество перемножений | int |
| x | Переменная x, которую задает пользователь | double |
| z | Искомое значение функции | double |
| mp | Результат перемножения | double |
| i | Счетчик цикла | int |

#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int i, num;  
 printf("Введите число: ");  
 scanf("%d", &num);  
 for(i = 1; i <= num; i++) {  
 printf("%d ", i);  
 }  
}

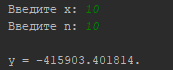
1. 

Задание 6

1. 
2. 

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| x | Переменная x | double |
| n | Количество сложений | int |
| y | Искомое значение функции | double |
| sum | Общая сумма | double |
| k | Счетчик цикла, переменная k | int |

#include <stdio.h>  
#include <math.h>  
  
int main() {  
 int k, n;  
 double x, y, sum = 0;  
 printf("Введите x: ");  
 scanf("%lf", &x);  
 printf("Введите n: ");  
 scanf("%d", &n);  
 for(k = 3; k <= n; k++) {  
 sum = sum + (2 \* pow(x, 3) \* k + cos(k) \* pow(x + 1, 0.5) - 2.3 / k);  
 }  
 y = 6.3 \* x - 4 \* sum;  
 printf("\ny = %f.", y);  
}

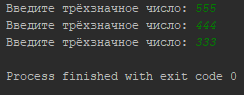
1. 

Задание 7

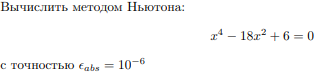
1. 
2. , если n1 + n2 + n3 > 10 ввести число еще раз.

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| num | Вводимое пользователем трехзначное число | Int |
| sum | Сумма всех цифр числа | int |

#include <stdio.h>  
  
int main() {  
 int num, sum = 0;  
 do {  
 printf("Введите трёхзначное число: ");  
 scanf("%d", &num);  
 sum = num % 10 + (num / 10 % 10) + num / 100;  
 } while(sum > 10);  
 return 0;  
}

1. 

Задание 8

1. 
2. 

|  |  |  |
| --- | --- | --- |
| Имя | Значение | Тип |
| eps | Заданная точность | #define |
| x1 | Результат вычисленной функции | double |
| x0 | Переменная x0 | double |
| fx | Вычисляемая функция | double |
| dfx | Производная функции | double |

#include <stdio.h>  
#include <math.h>  
  
#define eps 0.000001  
double fx(double x) { return pow(x, 4) - 18 \* x \* x + 6;}  
double dfx(double x) { return 4 \* pow(x, 3) - 36 \* x;}  
  
typedef double(\*function)(double x);  
  
double solve(function fx, function dfx, double x0) {  
 double x1 = x0 - fx(x0) / dfx(x0);  
 while (fabs(x1 - x0) > eps) {  
 x0 = x1;  
 x1 = x1 - fx(x1) / dfx(x1);  
 }  
 return x1;  
}  
  
int main () {  
 printf("%f\n", solve(fx, dfx, 4));  
 return 0;  
}

1. 