Министерство образования Республики Беларусь

Учреждение образования:

«Белорусский государственный университет информатики

и радиоэлектроники»

 Специальность «Программная инженерия»

Кафедра инженерной психологии и эргономики

Учебная дисциплина «Основы алгоритмизации и программирования»

Отчет

по лабораторной работе №3

«Циклические алгоритмы»

Преподаватель: Усенко Ф.В.

Сделал: Гр.410902 Дятко Е.М.

Минск 2024

**Цель работы:** сформировать умения разрабатывать программы с использованием операторов выбора, цикла, передачи управления.

Задание

14.1 Дано натуральное п, действительное х. Вычислить: 

14.2 Вычислить сумму ряда с заданной степенью точности α:



Код к заданиям можно посмотреть ниже:

1. #include <iostream>
2. #include <cmath>
3. using namespace std;
4. constexpr auto PI = 3.14159265;
5. void first\_part\_laba3();
6. void second\_part\_laba3();
7. int main() {
8. int n;
9. cout << "wich part you would like to see?: "; cin >> n;
10. if (n == 1) {
11. first\_part\_laba3();
12. }
13. else if (n == 2) {
14. second\_part\_laba3();
15. }
16. else {
17. cout << "You can`t play outside the rules!!!!";
18. }
19. system("pause");
20. return 0;
21. }
22. void first\_part\_laba3() {
23. int n;
24. float x, cosinus;
25. cout << "inpur n: "; cin >> n;
26. cout << "input x (in degrees): "; cin >> x;
27. x = x \* PI / 180;
28. cosinus = cos(fabs(x));
29. double resault = 0.5 - cosinus;
30. if (n < 2 && n > 0) {
31. cout << "resault: " << resault;
32. }
33. if (n <= 0) {
34. cout << "its not gonna work!";
35. }
36. else {
37. for (int i = 2; i <= n; i++) {
38. resault \*= i / (i + 1) - pow(cosinus, i);
39. }
40. cout << "resault: " << resault<<endl;
41. }
42. }
43. void second\_part\_laba3() {
44. double resault = 0.0, a, n, old\_value = 0.0, pres\_value = 0.0;
45. a = 0.001;
46. n = 1;
47. while (fabs(fabs(pres\_value) - fabs(old\_value)) > a || pres\_value == 0.0) {
48. old\_value = pres\_value;
49. pres\_value = pow(-1, n) \* 1 / (2 \* n);
50. resault += pres\_value;
51. n++;
52. }
53. cout << "resault: " << resault << endl;
54. }

**Комментарии к программе**

1. В строках 6-7 идёт создание скелета функций.
2. В 10,11 строке инициализируются и задаются переменная для определения какую функцию хочет запустить пользователь.
3. В строках 12-20 идёт проверка на то, какую часть пользователь хочет увидеть, если число что ввёл пользователь не является 1 или 2, то программа заканчивается.
4. 25 строка – функция для 14.1.
5. 26,27 строка инициализация переменных нужных для вычислений.
6. 28,29 строка ввод пользователем переменных созданных ранее.
7. 30-32 строки нахождение значения косинуса угла и первоначального значения программы.
8. 33-44 строки в зависимости от кол-во множителей введённые пользователем рассчитывается результат формулы.
9. 47 строка – функция для 14.2.
10. 48-50 строки инициализация переменных нужных в 14.2.
11. 51-56 строки – выполнение программы до того момента пока разница между предыдущим значением итерации и текущим его значением разница не меньше чем a(точность).
12. 57 строка – вывод результата на консоль пользователя.

**Примеры работы программы**

14.1 :



Рисунок 1



Рисунок 2

14.2:



Рисунок 3

**Блок схема кода**

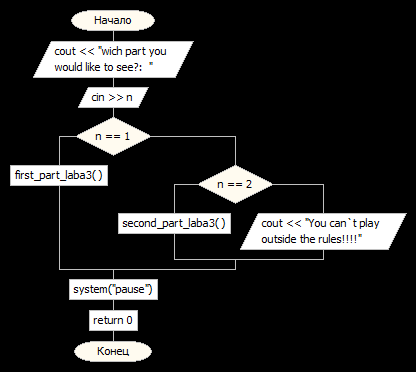


Рисунок 4

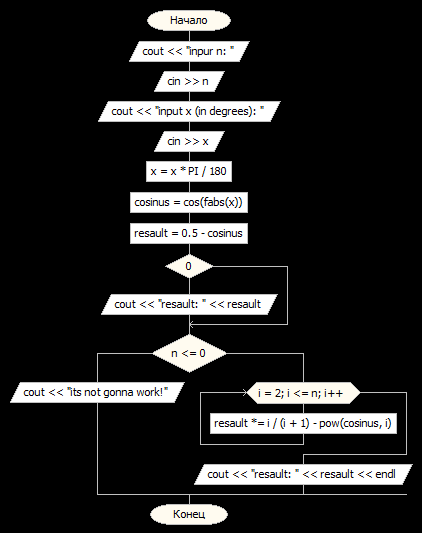


Рисунок 5

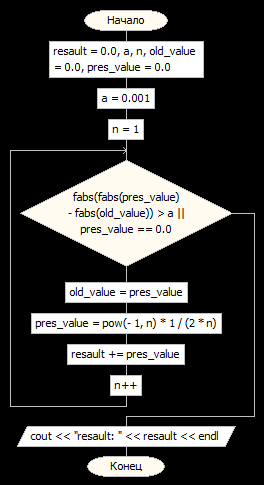


Рисунок 6

**Вывод:**

В ходе выполнения данной лабораторной работе мы разработали программу с использованием операторами выбора, цикла и передачи управления.