**Тема лабораторной работы:** Процедуры и функции

**Цель л/р:** Решение ДЦВП при помощи пользовательской процедуры

**Используемое оборудование:** draw.io(блок-схемы), PascalABC(код программы)

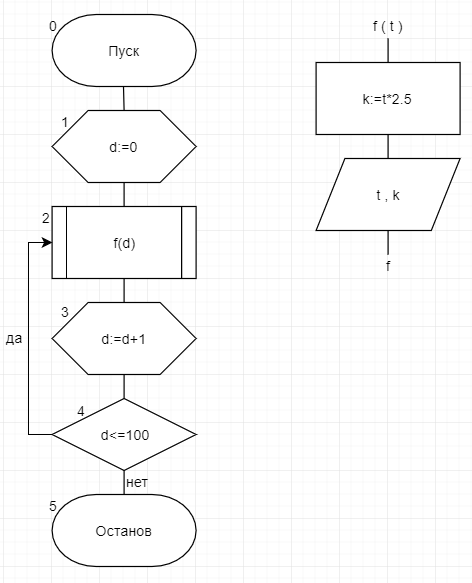
**Постановка задачи:**

Перевести дюймы в сантиметры от 0 до 100 дюймов (1 дюйм=2.5 см). Результаты вывести в виде таблицы. Операторы для формирования вывода таблицы оформить в виде пользовательской функции.

**Математическая модель:**

s = d \* 2,5

**Блок-схема:**



**Список идентификаторов:**

| **Имя** | **Смысл** | **Тип** |
| --- | --- | --- |
| d | параметр цикла | integer |
| k | выводимое процедурой значение | real |
| t | параметр функции | integer |
| f | процедура | real |

**Код программы:**

**program** lr51;

**var** d: integer;

**procedure** f(t: integer);

**var** k:real;

**begin**

k:=t\*2.5;

writeln(t,' | ',k);

**end**;

**begin**

writeln('in | cm');

**for** d := 0 **to** 100 **do**

f(d);

**end**.

**Результат выполненной работы:**

in | cm

0 | 0

1 | 2.5

2 | 5

3 | 7.5

4 | 10

5 | 12.5

6 | 15

7 | 17.5

8 | 20

9 | 22.5

10 | 25

11 | 27.5

12 | 30

13 | 32.5

14 | 35

15 | 37.5

16 | 40

17 | 42.5

18 | 45

19 | 47.5

20 | 50

21 | 52.5

22 | 55

23 | 57.5

24 | 60

25 | 62.5

26 | 65

27 | 67.5

28 | 70

29 | 72.5

30 | 75

31 | 77.5

32 | 80

33 | 82.5

34 | 85

35 | 87.5

36 | 90

37 | 92.5

38 | 95

39 | 97.5

40 | 100

41 | 102.5

42 | 105

43 | 107.5

44 | 110

45 | 112.5

46 | 115

47 | 117.5

48 | 120

49 | 122.5

50 | 125

51 | 127.5

52 | 130

53 | 132.5

54 | 135

55 | 137.5

56 | 140

57 | 142.5

58 | 145

59 | 147.5

60 | 150

61 | 152.5

62 | 155

63 | 157.5

64 | 160

65 | 162.5

66 | 165

67 | 167.5

68 | 170

69 | 172.5

70 | 175

71 | 177.5

72 | 180

73 | 182.5

74 | 185

75 | 187.5

76 | 190

77 | 192.5

78 | 195

79 | 197.5

80 | 200

81 | 202.5

82 | 205

83 | 207.5

84 | 210

85 | 212.5

86 | 215

87 | 217.5

88 | 220

89 | 222.5

90 | 225

91 | 227.5

92 | 230

93 | 232.5

94 | 235

95 | 237.5

96 | 240

97 | 242.5

98 | 245

99 | 247.5

100 | 250

**Анализ результатов вычисления:**

В данном случае процедура удобна тем, что сразу выводит дюймы и сантиметры. Нам остаётся лишь зациклить данную процедуру, при этом параметр цикла, равен значению дюйма.

**Задание 2**

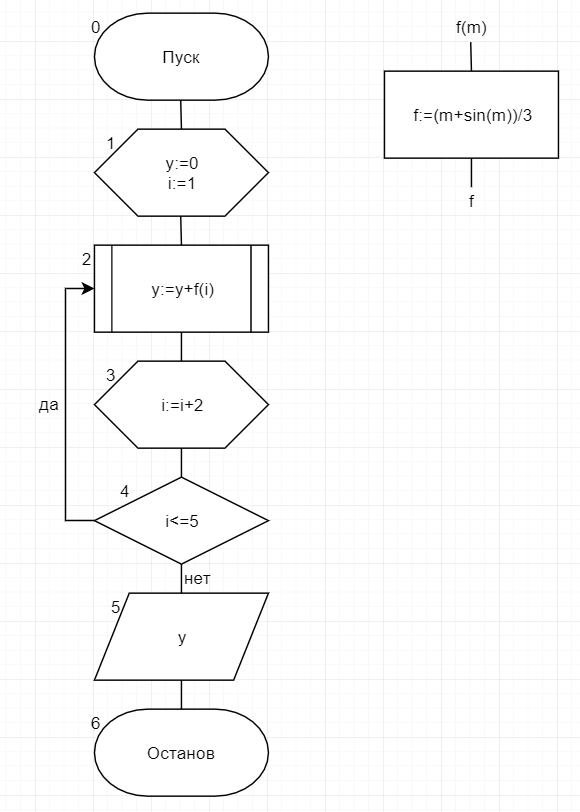
**Постановка задачи:**

Рассчитать значение y, определив и использовав необходимую функцию

**Математическая модель:**



**Блок-схема:**



**Список идентификаторов:**

| **Имя** | **Смысл** | **Тип** |
| --- | --- | --- |
| y | результат | real |
| m | переменная функции | integer |
| i | параметр цикла | integer |
| f | функция | real |

**Код программы:**

**program** lr52;

**var** y: real; i:integer;

**function** f(m: integer): real;

**begin**

f := (m + sin(m)) / 3;

**end**;

**begin**

y:=0;

i:=1;

**while** i<=5 **do begin**

y:=y+f(i);

i:=i+2;

**end**;

writeln(y:1:5);

**end**.

**Результат выполненной работы:**

3.00789

**Анализ результатов вычислений:**

С помощью функции высчитываем всю дробь, а затем находим сумму функций, образованных от трёх разных констант.

**Задание 3**

**Постановка задачи:**

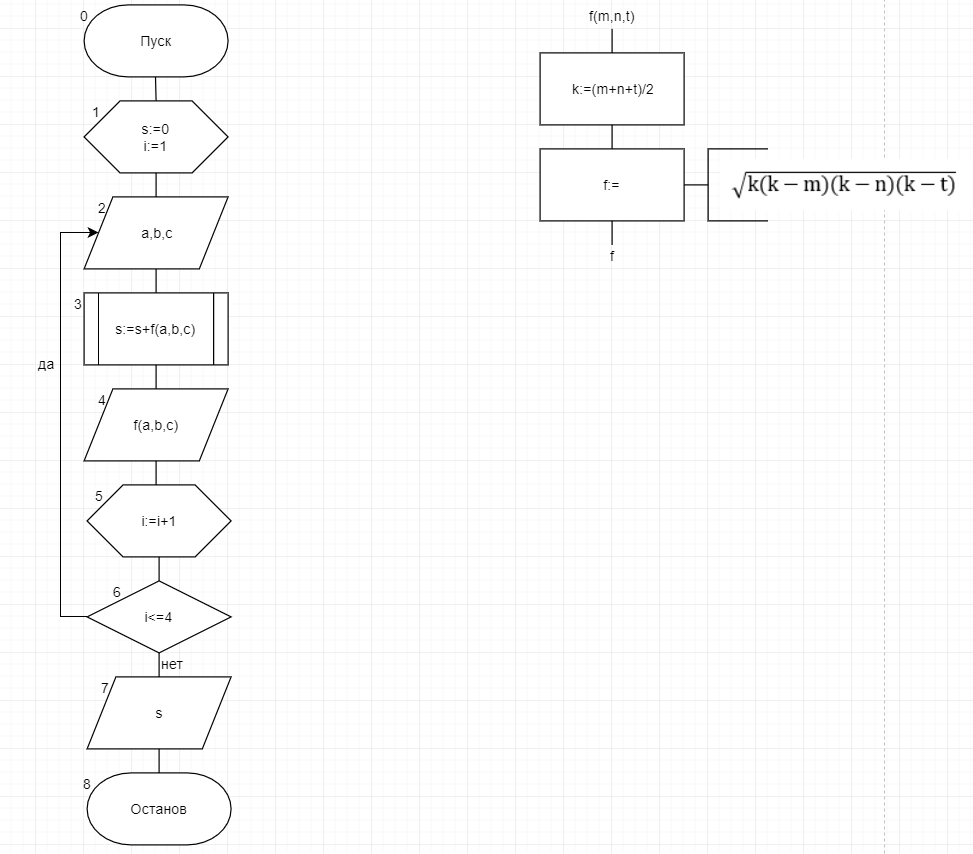
Вычислить площадь фигуры, заданной сторонами. Фигура не является прямоугольником, а треугольники, которые ее составляют, не являются прямоугольными.

**Математическая модель:**

k =

f =

**Блок-схема:**



**Список идентификаторов:**

| **Имя** | **Смысл** | **Тип** |
| --- | --- | --- |
| s | накопительная переменная | real |
| i | параметр цикла | integer |
| a | сторона треугольника | real |
| b | сторона треугольника | real |
| c | сторона треугольника | real |
| k | полупериметр | real |
| m | переменная функции | integer |
| n | переменная функции | integer |
| t | переменная функции | integer |
| f | функция | Real |

**Код программы:**

**program** lr53;

**var** a, b, c, i: integer; s, k : real;

**function** f(m, n, t: integer): real;

**begin**

k := (m + n + t) / 2;

f := sqrt(k \* (k - m) \* (k - n) \* (k - t));

**end**;

**begin**

**for** i := 1 **to** 4 **do**

**begin**

writeln('Введите стороны ', i , ' треугольника');

readln(a, b, c);

s := s + f(a, b, c);

**end**;

writeln('Площадь фигуры равна ', s:1:3);

**end**.

**Результат выполненной работы:**

Введите стороны 1 треугольника

7

8

5

Площадь 1 треугольника - 17.321

Введите стороны 2 треугольника

6

7

4

Площадь 2 треугольника - 11.977

Введите стороны 3 треугольника

6

7

4

Площадь 3 треугольника - 11.977

Введите стороны 4 треугольника

4

5

6

Площадь 4 треугольника - 9.922

Площадь фигуры равна 51.195

**Анализ результатов вычислений:**

Площадь фигуры высчитывается путём суммирования площадей четырёх треугольников, вычисленных с помощью функции.Следует заметить, что стороны треугольника, вводимые с клавиатуры, должны быть положительны.

**Задание 4**

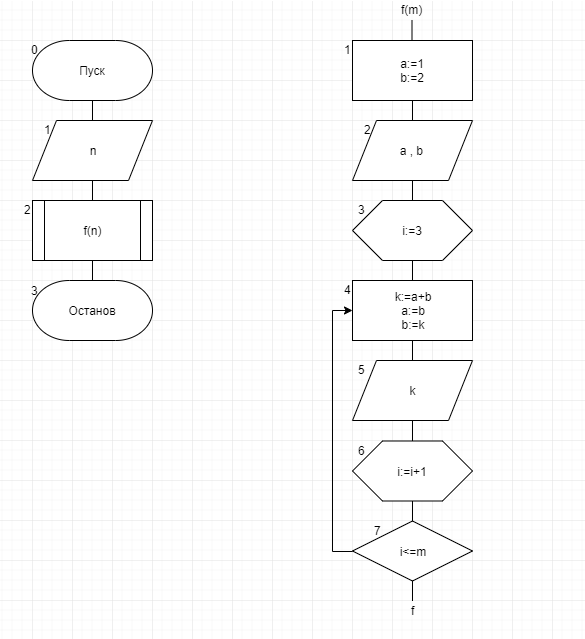
**Постановка задачи:**

С клавиатуры вводится число. Вывести на экран столько элементов ряда Фибоначчи, сколько указал пользователь. Вычисление ряда организовать в функцию. Например, если на ввод поступило число 6, то вывод должен содержать шесть первых чисел ряда Фибоначчи: 1 2 3 5 8 13.

**Математическая модель:**

k = a + b

**Блок-схема:**



**Список идентификаторов:**

| **Имя** | **Смысл** | **Тип** |
| --- | --- | --- |
| n | вводимое число | integer |
| a | первое знач. ряда | integer |
| b | второе знач. ряда | integer |
| k | след. знач. ряда | integer |
| i | параметр цикла | integer |
| f | функция | integer |
| m | переменная функции | integer |

**Код программы:**

**program** lr54;

**var** n: integer;

**function** f(m: integer): integer;

**var** i,a,b,k: integer;

**begin**

a := 1;

b := 2;

write(a,' ',b);

**for** i := 3 **to** m **do**

**begin**

k := a + b;

a := b;

b := k;

write(' ', k);

**end**;

**end**;

**begin**

writeln('Введите число');

read(n);

f(n);

**end**.

**Результат выполненной работы:**

Введите число

7

1 2 3 5 8 13 21

**Анализ результатов вычисления:**

Ряд Фибоначчи вычисляется таким образом, что последующее значение ряда будет равно сумме двух предыдущих значений. Поскольку начальные значения должны быть заданы и выведены на экран, то первые два элемента ряда Фибоначчи выводятся перед циклом.

**Вывод:**

Была изучена реализация ДЦВП с использованием пользовательской процедуры и функции при помощи компилятора Pascal.