

Оглавление

| | |
|--|----|
| 1. Постановка задачи №1 ----- | 3 |
| 2. Структурные требования ----- | 3 |
| 3. Описание работы----- | 2 |
| 3.1. Используемые переменные----- | 2 |
| 3.2. Псевдокод программы----- | 2 |
| 3.3. Блок схема алгоритма ----- | 2 |
| 3.4. Листинг программной реализации----- | 3 |
| 3.5. Результат ----- | 4 |
| 3.6. Тесты ----- | 4 |
| 1. Постановка задачи №2 ----- | 5 |
| 2. Структурные требования ----- | 5 |
| 3. Описание работы ----- | 5 |
| 3.1. Используемые переменные----- | 5 |
| 3.2. Псевдокод программы----- | 5 |
| 3.3. Блок схема алгоритма ----- | 5 |
| 3.4. Листинг программной реализации----- | 6 |
| 3.5. Результат ----- | 7 |
| 3.6. Тесты ----- | 7 |
| 1. Постановка задачи №3----- | 8 |
| 2. Структурные требования ----- | 8 |
| 3. Описание работы----- | 8 |
| 3.1. Используемые переменные----- | 8 |
| 3.2. Псевдокод программы----- | 8 |
| 3.3. Блок схема алгоритма ----- | 9 |
| 3.4. Листинг программной реализации----- | 10 |
| 3.5. Результат ----- | 11 |
| 3.6. Тесты ----- | 12 |
| 1. Постановка задачи №4 ----- | 12 |
| 2. Структурные требования ----- | 12 |
| 3. Описание работы----- | 12 |
| 3.1. Используемые переменные----- | 12 |
| 3.2. Псевдокод программы----- | 12 |
| 3.3. Блок схема алгоритма ----- | 13 |
| 3.4. Листинг программной реализации----- | 13 |
| 3.5. Результат ----- | 13 |
| 3.6. Тесты ----- | 13 |
| 1. Постановка задачи №6 ----- | 17 |
| 2. Структурные требования ----- | 17 |
| 3. Описание работы ----- | 17 |
| 3.1. Используемые переменные ----- | 14 |
| 3.2.Псевдокод программы ----- | 15 |

| | |
|--|----|
| 3.3. Блок схема алгоритма ----- | 15 |
| 3.4. Листинг программной реализации----- | 16 |
| 3.5. Результат----- | 16 |
| 3.6. Тесты ----- | 16 |
| 1. Постановка задачи №7(а) ----- | 17 |
| 2. Структурные требования ----- | 17 |
| 3. Описание работы ----- | 17 |
| 3.1. Используемые переменные----- | 17 |
| 3.2. Псевдокод программы----- | 17 |
| 3.3. Блок схема алгоритма ----- | 17 |
| 3.4. Листинг программной реализации----- | 18 |
| 3.5. Результат ----- | 19 |
| 3.6. Тесты ----- | 19 |
| 1. Постановка задачи №7(б)----- | 19 |
| 2. Структурные требования ----- | 19 |
| 3. Описание работы----- | 19 |
| 3.1. Используемые переменные----- | 19 |
| 3.2. Псевдокод программы----- | 19 |
| 3.3. Блок схема алгоритма ----- | 19 |
| 3.4. Листинг программной реализации----- | 21 |
| 3.5. Результат ----- | 22 |
| 3.6. Тесты ----- | 22 |
| Список литературы ----- | 26 |

1. Постановка задачи №1

1. Дан радиус окружности. Найти длину окружности, её диаметр и площадь круга.

2. Структурные требования

$R > 0$.

3. Описание работы

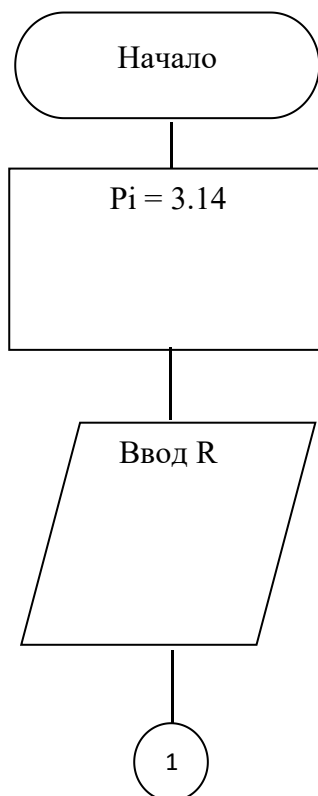
3.1. Используемые переменные

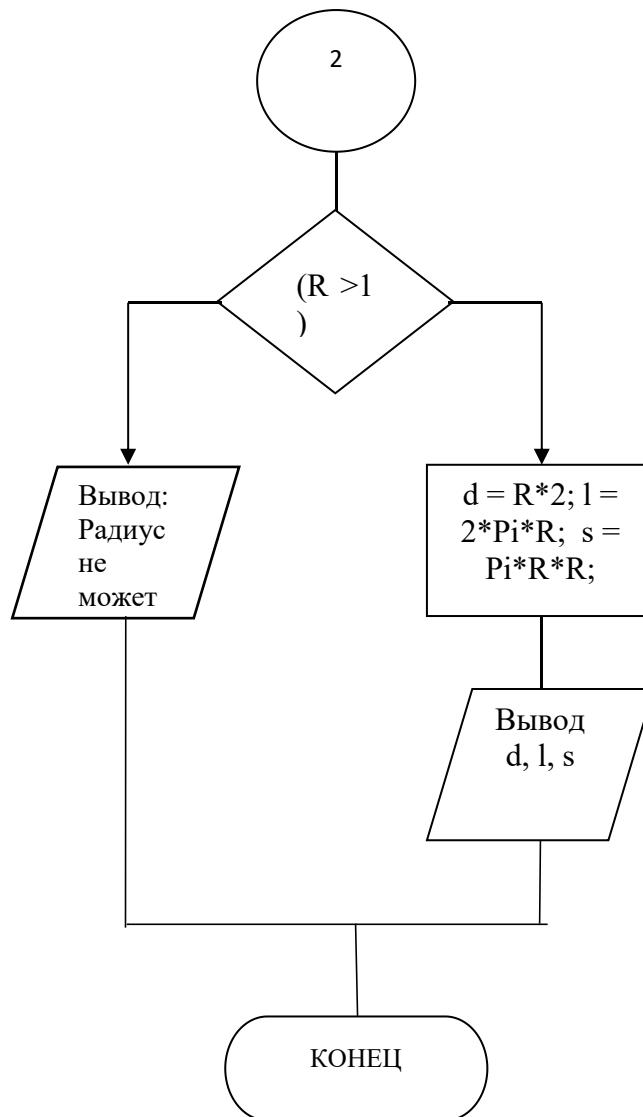
P_i , SS , D , R , S

3.2. Псевдокод программы

Пользователь вводит R . Далее идет проверка если $R > 0$, так как радиус не может быть отрицательным, тогда происходит подсчет $D = R * 2$, $SS = 2 * P_i * R$, $S = P_i * R * R$. После этого выводится D , SS , S . Если радиус, введенный пользователем, не подходит под условие, выводится сообщение “Радиус не может быть отрицательным”

3.3. Блок - схема алгоритма





3.4. Листинг программной реализации

```
#include <stdio.h>
```

```
int main() {
```

```
    const float Pi = 3.14;
```

```
    float SS, S;
```

```
    int R, D;
```

```
    printf("Введите радиус окружности: ");
```

```
    scanf("%d", &R);
```

```
    if (R > 0) {
```

```
        D = R*2;
```

```
        SS = 2*Pi*R;
```

```
        S = Pi*R*R;
```

```
        printf("%d\n%f\n%f", D, SS, S);
```

```
    }
```

```
    else{
```

```
        printf("Радиус не может быть отрицательным");
```

```

    }
    return 0;
}

```

3.5. Результат

```

20 }
input
6
ct 18.840000
28.260000
...Program finished with exit code 0
Press ENTER to exit console.

```

3.6. Тесты

| | |
|------|-----------------|
| R=12 | D=24 |
| | SS=75.360001 |
| | S=452.160004 |
| R=8 | D=16 |
| | lol = 50.240002 |
| | S=200.960007 |
| R=-9 | - |
| | - |
| | - |

1. Постановка задачи №2

Треугольник задан координатами своих вершин. Найти периметр и площадь треугольника.

2. Структурные требования

$((W - X) * (Y - Q) - (S - X) * (A - Q) \neq 0)$ и $((X - S) * (A - Y) - (W - S) * (Q - Y) \neq 0)$

3. Описание работы

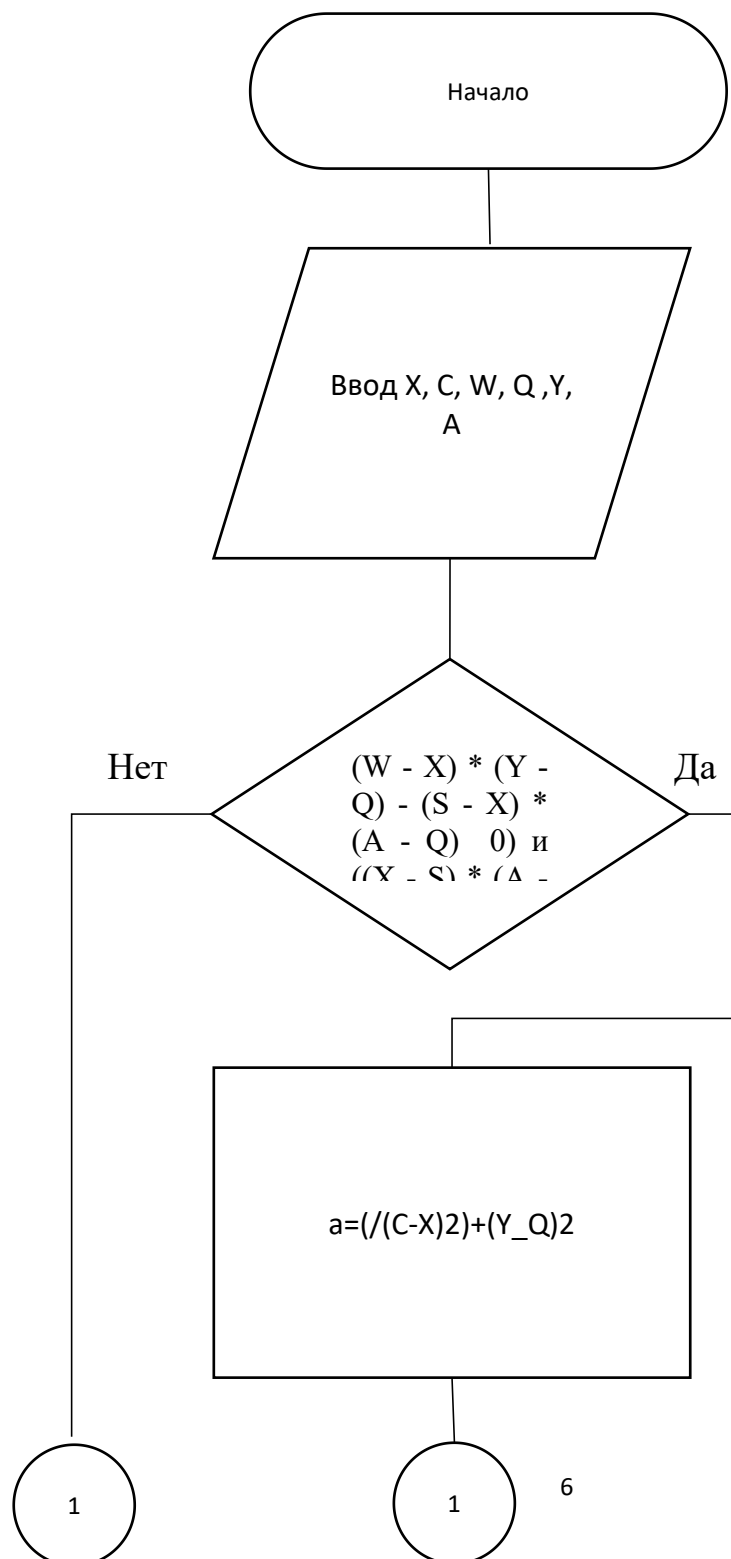
3.1. Используемые переменные

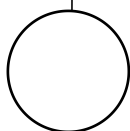
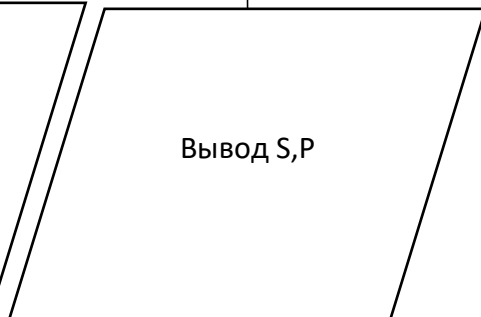
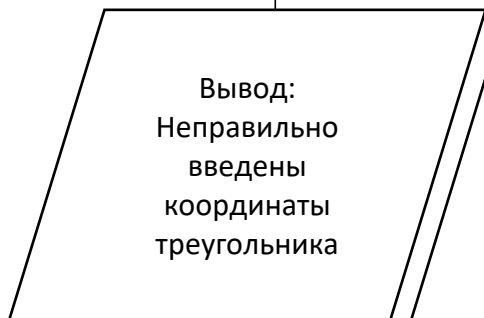
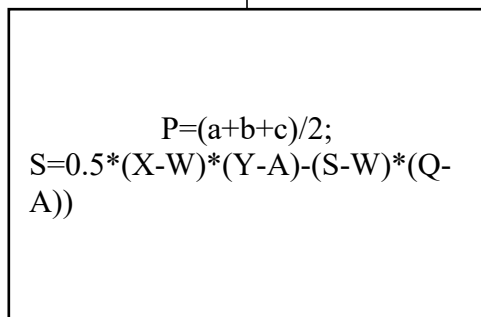
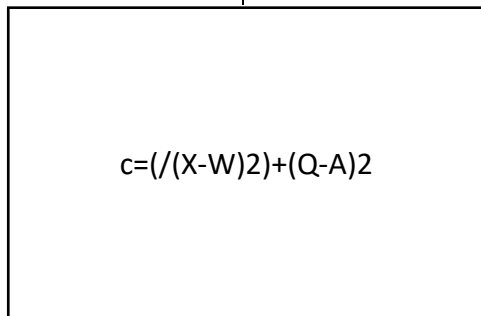
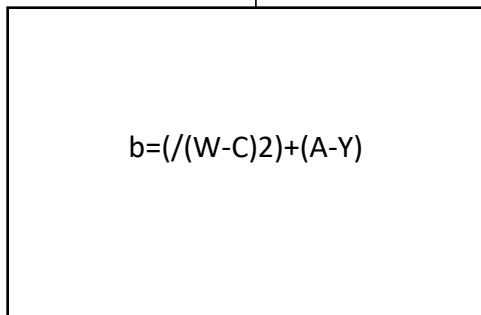
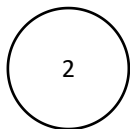
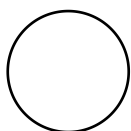
W, S, A, X, Q, Y, P, S, a, b, c

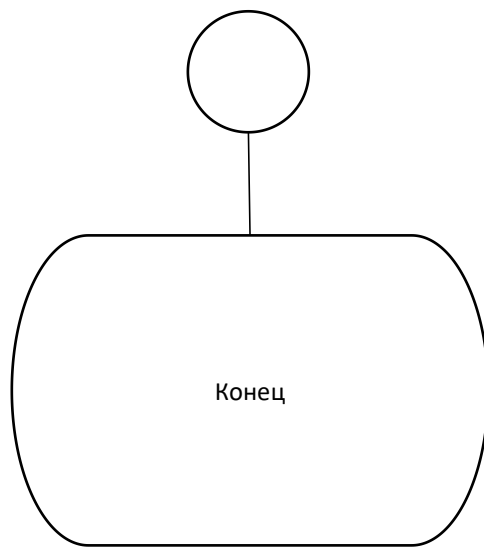
3.2. Псевдокод программы

Пользователь вводит координаты треугольника: X, Q, S, Y, W, A. Далее. происходит подсчет сторон по формулам: $a = \sqrt{(C - X)^2 + (Y - Q)^2}$, $b = \sqrt{(W - C)^2 + (A - Y)^2}$, $c = \sqrt{(X - W)^2 + (Q - A)^2}$. Далее мы можем высчитать P и S: $P = (a + b + c)/2$ и $S = 0.5 * ((X - W) * (Y - A) - (S - W) * (Q - A))$

3.3. Блок - схема алгоритма







3.4. Листинг программной реализации

```

#include<stdio.h>
#include<math.h>
int main()
{
    int X, C, W, Q, Y, A;
    float a, b, c, S, P;
    printf("Введите координаты треугольника: X, Q, S, Y, W, A \n");
    scanf("%d %d %d %d %d %d", &X, &Q, &S, &Y, &W, &A);

    if (((W - X) * (Y - Q) - (C - X) * (A - Q) != 0) && ((X - S) * (A - Y) -
    (W - S) * (Q - Y) != 0) && ((S - X) * (A - Q) - (W - X) * (Y - Q) != 0))
    {
        a = sqrt(pow((C-X), 2) + pow((Y-Q), 2));
        b = sqrt(pow((W-C), 2) + pow((A-Y), 2));
        c = sqrt(pow((X-W), 2) + pow((Q-A), 2));
        P = a + b + c;
        S = 0.5*(fabs((X-W)*(Y-A)-(C-W)*(Q-A)));
        printf("площадь = %f периметр = %f", S, P);
    }
    else{
        printf("Не правильно введены координаты треугольника");
    }
}
  
```

3.5. Результат

```

cmd
Введите координаты треугольника: X, Q, S, Y, W, A
3
4
5
6
7
8
площадь = 10.000000 периметр = 16.542515
...Program finished with exit code 0
Press ENTER to exit console.
  
```


3.6. Тесты

| | |
|-----|-------------|
| X=2 | S=7.500000 |
| Q=4 | |
| S=5 | P=22.037979 |
| Y=7 | |

| | |
|------|-------------|
| X=10 | S=4.000000 |
| Q=2 | |
| S=5 | P=21.714602 |
| Y=6; | |

1. Постановка задачи №3

Возраст Тани - X лет, а возраст Мити - Y лет. Найти их средний возраст, а также определить, на сколько отличается возраст каждого ребенка от среднего значения.

2. Структурные требования

Во-первых, возраст каждого не может быть чисто логически отрицательным, поэтому мы это указываем для пользователя – $Z > -1$ и $D > -1$. Также мы ведем подсчет месяцев, поэтому указываем, что $G1 > -1$ и $G1 < 13$ и $a1 > -1$ и $a1 < 13$.

3. Описание работы

3.1. Используемые переменные

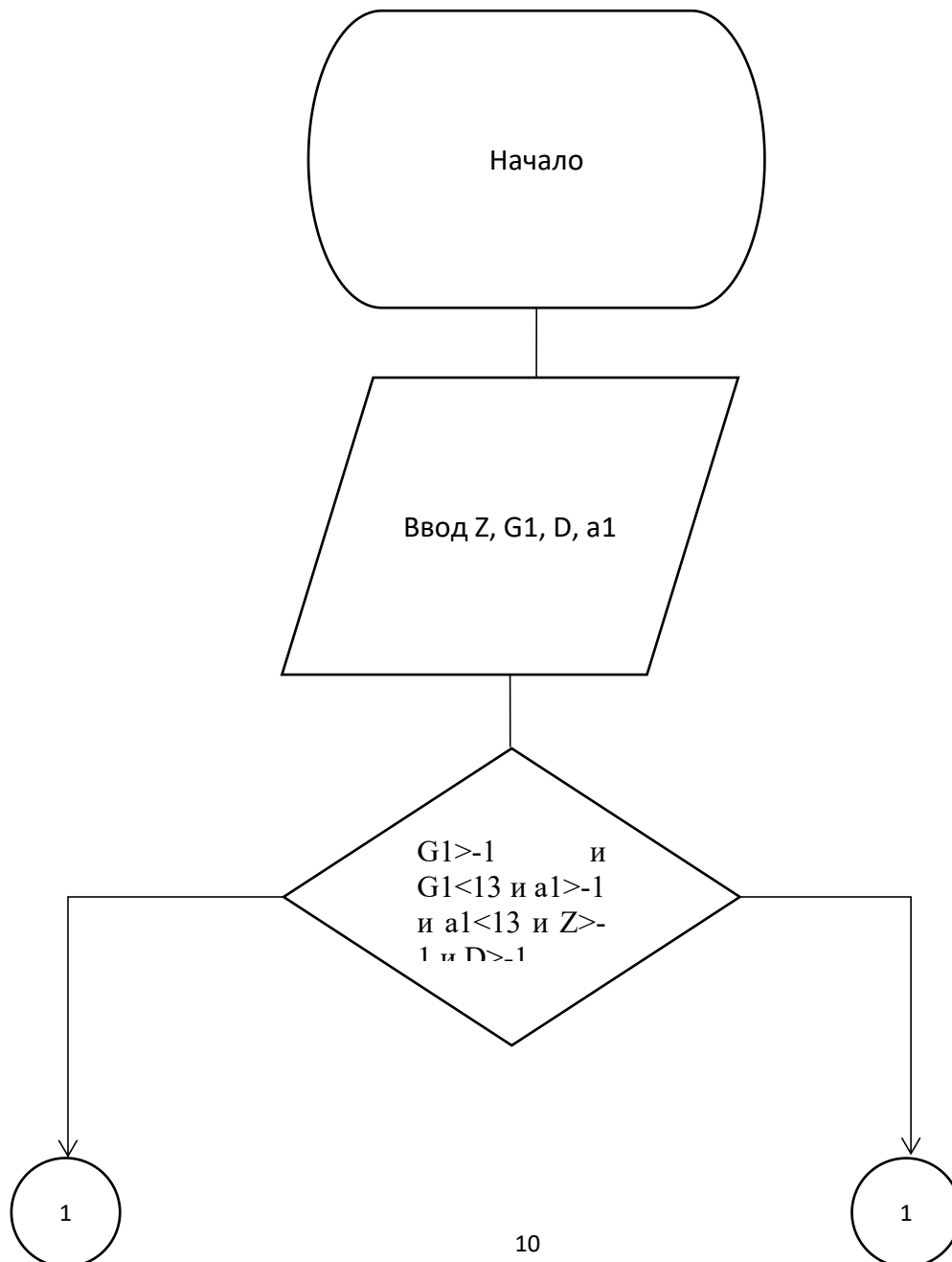
Z, D, G, G1, a1, S, D1, Z1, a, S1, S2

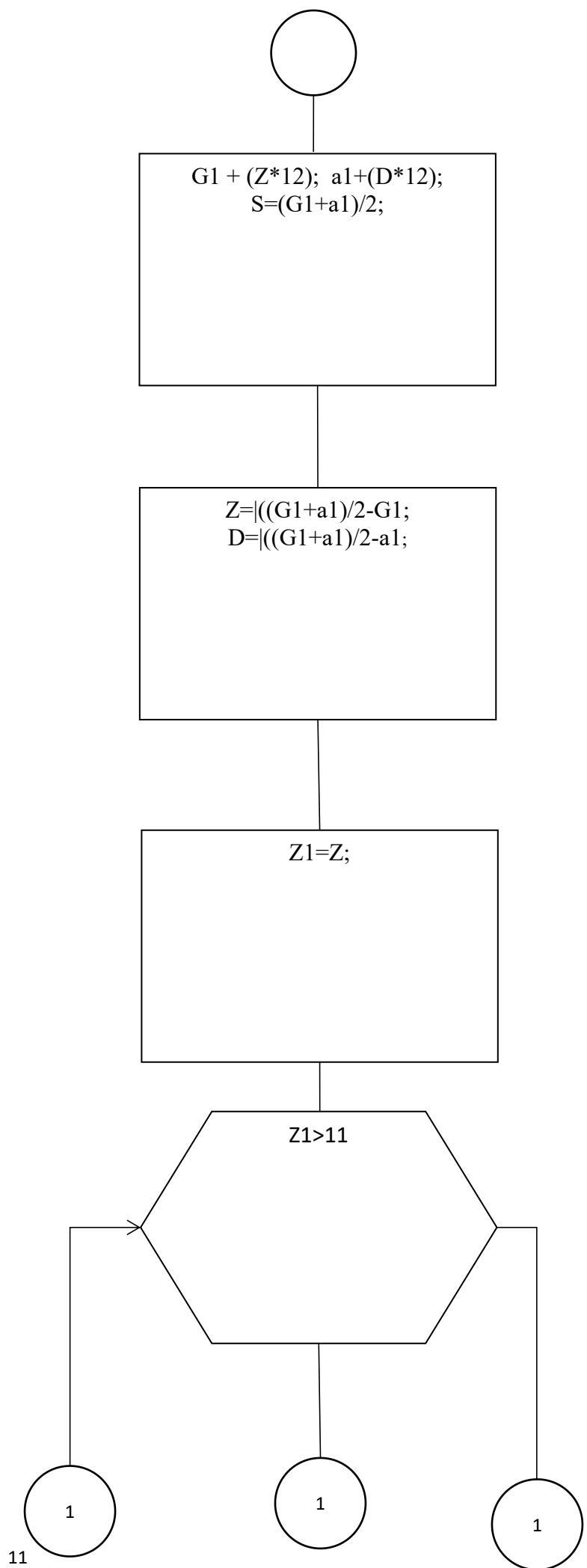
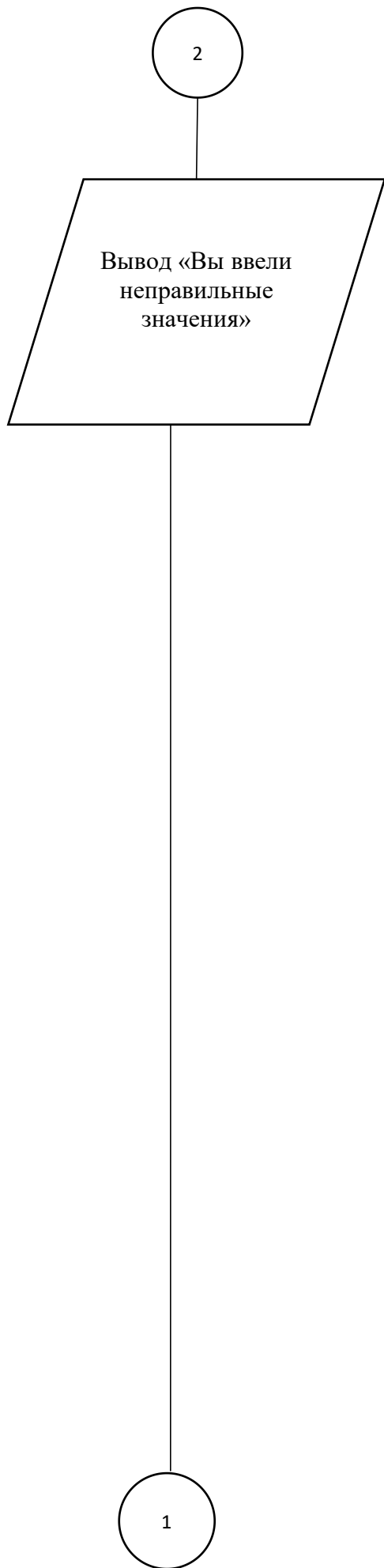
3.2. Псевдокод программы

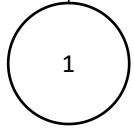
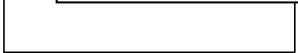
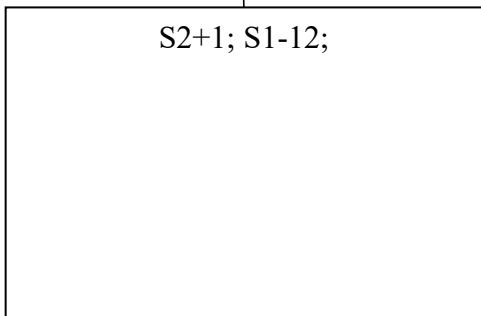
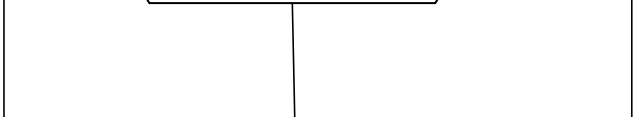
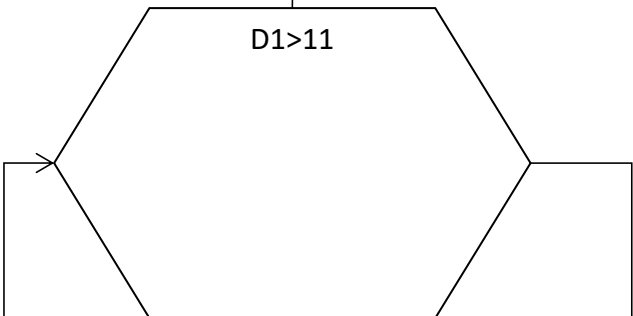
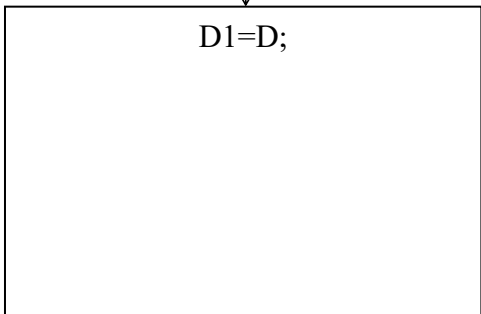
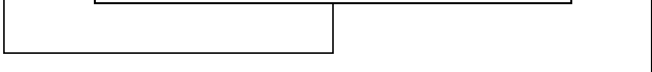
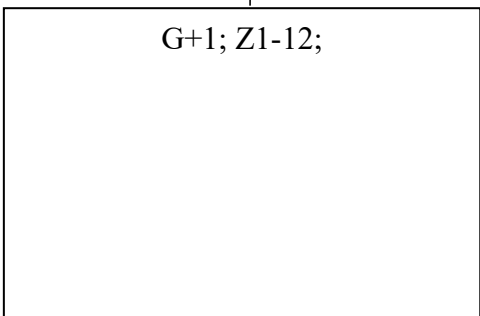
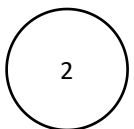
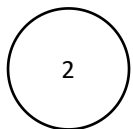
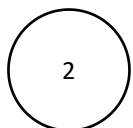
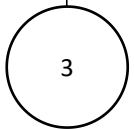
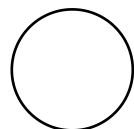
Введите возраст Тани и Мити. Год и месяц каждого. Если $Z = 0$ и $D = 0$ и $G1 = 0$ и $a1 = 0$, тогда выводится сообщение о том, что введены неправильные значения. Иначе если $G1 > -1$ и $G1 < 13$ и $a1 > -1$ и $a1 < 13$ и $Z > -1$ и $D > -1$, тогда $G1 = G1 + (Z * 12)$ – месяцы Тани. $a1 = a1 + (D * 12)$ – месяцы Мити. Далее высчитывается среднее значение $S = (G1 + a1) / 2$ – среднее значение месяцев. Далее $Z = \text{abs}((G1 + a1) / 2 - G1)$ – среднее значение по годам у Тани. $D = ((G1 + a1) / 2 - a1)$ – среднее

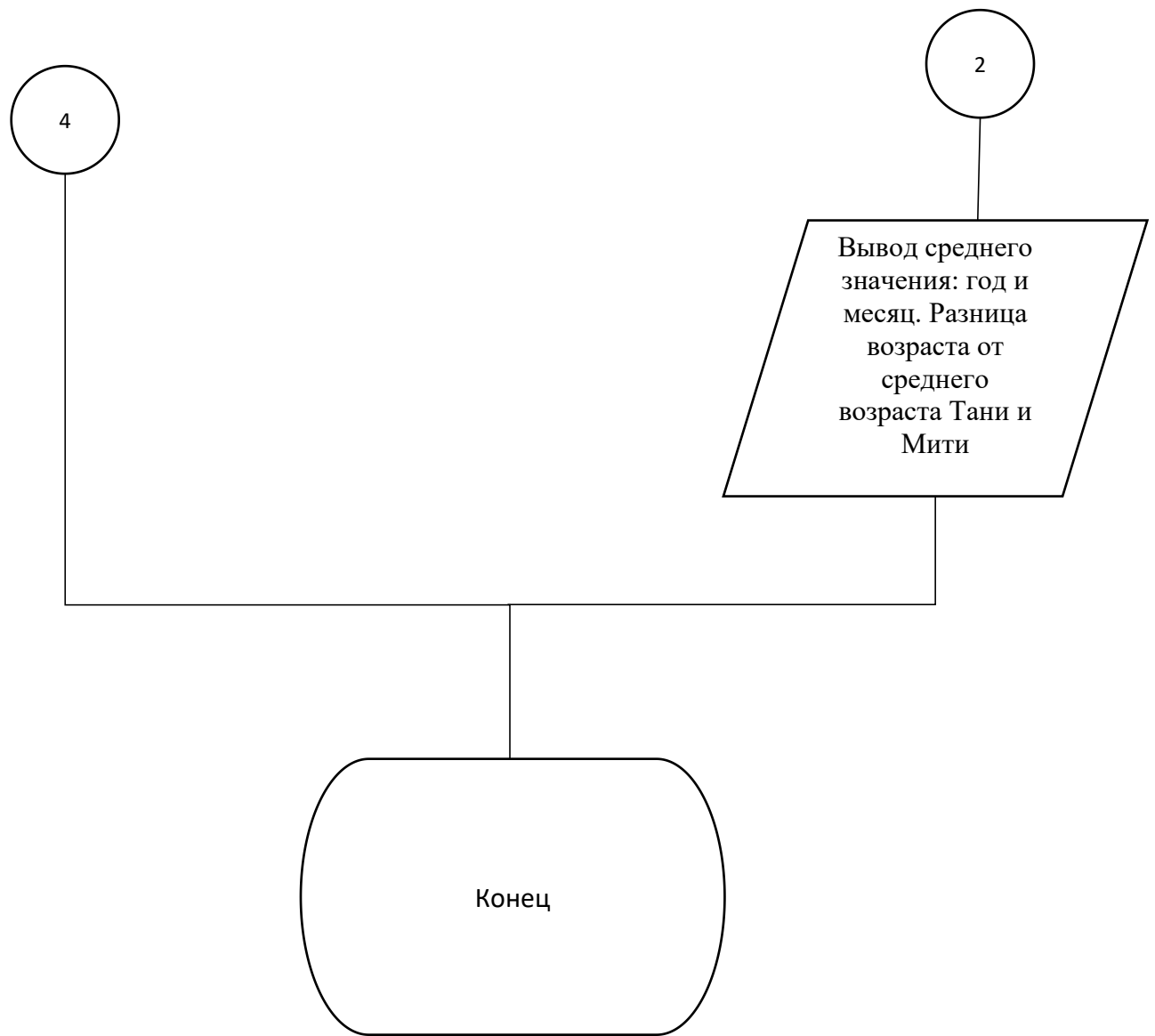
значение по годам у Мити. Далее $Z1=Z$. Пока $Z1 > 11$, $G=G+1$ и $Z1=Z1-12$. $D1=D$. Пока $D1 > 11$, $a=a+1$ и $D1=D1-12$. $S1=S$. Пока $S1 > 11$, $S2=S+1$ и $S1=S1-12$. Далее выводится среднее значение в году и месяце и разницы возраста от среднего значения Тани и Мити в годах и месяцах.

3.3. Блок - схема алгоритма









3.4. Листинг программной реализации

```
#include <stdio.h>
```

```
int main()
{
    int Z, D, G1, a1, S, D1, Z1, G=0, a=0, S1=0, S2=0;
    printf("Возраст тани и мити, сначала год тани потом месяц ");
    scanf("%i%i %i %i", &Z, &G1, &D, &a1);
    if(G1 == 0 && a1 == 0 && Z == 0 && D == 0)
        printf("Введенное значение не должно равняться: 0");
    else {
        if (G1 > -1 && G1 < 13 && a1 > -1 && a1 < 13 && Z > -1 && D > -1)
        {
```

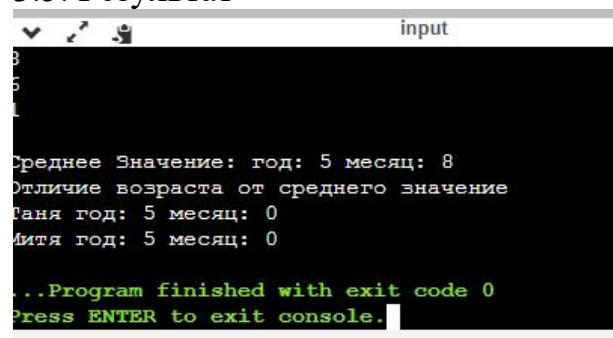
```

G1 += Z * 12; //месяцев тани
a1 += D * 12; //месяцев мити
S = (G1 + a1) / 2; //среднее значение
Z = abs((G1 + a1) / 2 - G1);
D = abs((G1 + a1) / 2 - a1);
Z1 = Z;
while (Z1 > 11)
{
    G += 1;
    Z1 -= 12;
}
D1 = D;
while (D1 > 11)
{
    a += 1;
    D1 -= 12;
}
S1 = S;
while (S1 > 11)
{
    S2 += 1;
    S1 -= 12;
}

printf("\nСреднее Значение: год: %i месяц: %i\nОтличие возраста
от среднего значения\nТаня год: %i месяц: %i\nМитя год: %i
месяц: %i", S2, S1, Z1, G, D1, a); //выводит результат
}
else
printf("Ведены некорректные значения");
}
}

```

3.5. Результат



```

input
Среднее Значение: год: 5 месяц: 8
Отличие возраста от среднего значения
Таня год: 5 месяц: 0
Митя год: 5 месяц: 0
...Program finished with exit code 0
Press ENTER to exit console.

```

3.6. Тесты

| | | | |
|------|---------|------------------------------------|---------------------------|
| Митя | Лет = 3 | Сред. значение = 3 года и 1 мес | |
| | Мес=6 | Отличие возраста | |
| Таня | Лет = 2 | Таня = 5 лет и 0 мес | Митя 4 года и 0 мес |
| | Мес=9 | | |

| | | | |
|------|----------|------------------------------------|---------------------------|
| Митя | Лет = 9 | Сред. значение = 12 лет и 9 мес | |
| | Мес=4 | Отличие возраста | |
| Таня | Лет = 16 | Таня = 5 лет и 3 мес. | Митя 6 лет и 3 мес. |
| | Мес=3 | | |

1. Постановка задачи №4

Вычислить значение логического выражения при следующих значениях логических величин I, L, Ll: I = Истина, L = Ложь, Ll= Ложь.

- а) I или L
- б) I и L
- в) L или Ll.

2. Структурные требования

- а) I или L
- б) I и L
- в) L или Ll.

3. Описание работы

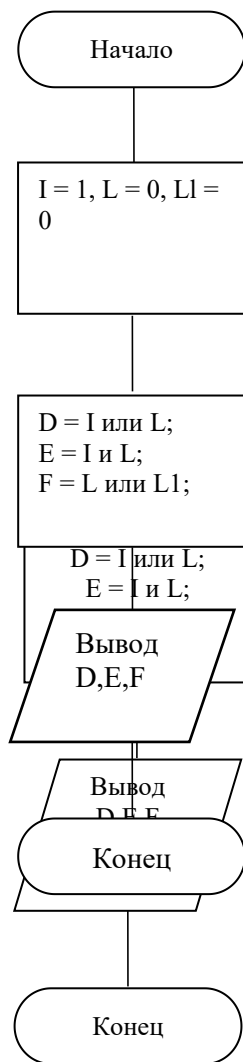
3.1. Используемые переменные

I, Ll, L, D, E, F

3.2. Псевдокод программы

Есть переменные I, L, Ll. I=1; L=0; Ll=0; D=A||B (A или B), E=A&&D (A и B), F= B||C (B или C). Вывод D, E, F.

3.3. Блок - схема алгоритма



3.4. Листинг программной реализации

```

#include <stdio.h>

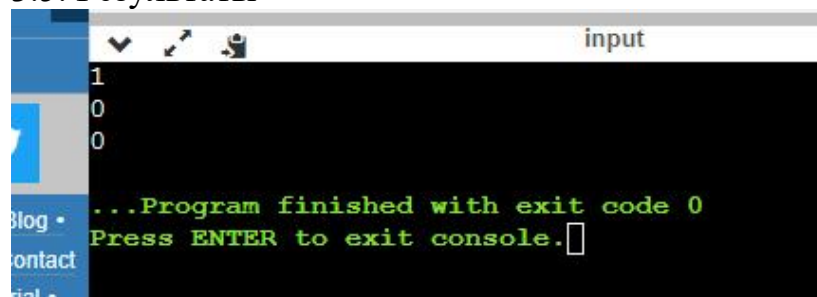
int main()
{

int I , L, Ll, D, E, F;
I = 1;
L = 0;
Ll = 0;
D = I||L;
E = I&&L;
F = L||Ll;
printf("%i\n%i\n%i", D, E, F);

return 0;
}

```


3.5. Результаты



```
input
1
0
0
...Program finished with exit code 0
Press ENTER to exit console.
```

3.6. Тесты

| Условие | Ответ |
|---------|-------|
| 0 | 0 |
| 0 | 0 |
| 0 | 0 |

1. Постановка задачи №6

Пусть в прямоугольной системе координат задана точка, для которой . Найти номер четверти плоскости, которой принадлежит заданная точка. Считать, что номер определяется по формуле

$$N = \begin{cases} 1, & \text{если } x > 0, y > 0, \\ 2, & \text{если } x < 0, y > 0, \\ 3, & \text{если } x < 0, y < 0, \\ 4, & \text{если } x > 0, y < 0. \end{cases}$$

2. Структурные требования

$Z > 0, A > 0$ (первая четверть); - $Z < 0, A > 0$ (вторая четверть); - $Z < 0, A < 0$ (третья четверть); - $Z > 0, A < 0$ (четвертая четверть).

3. Описание работы

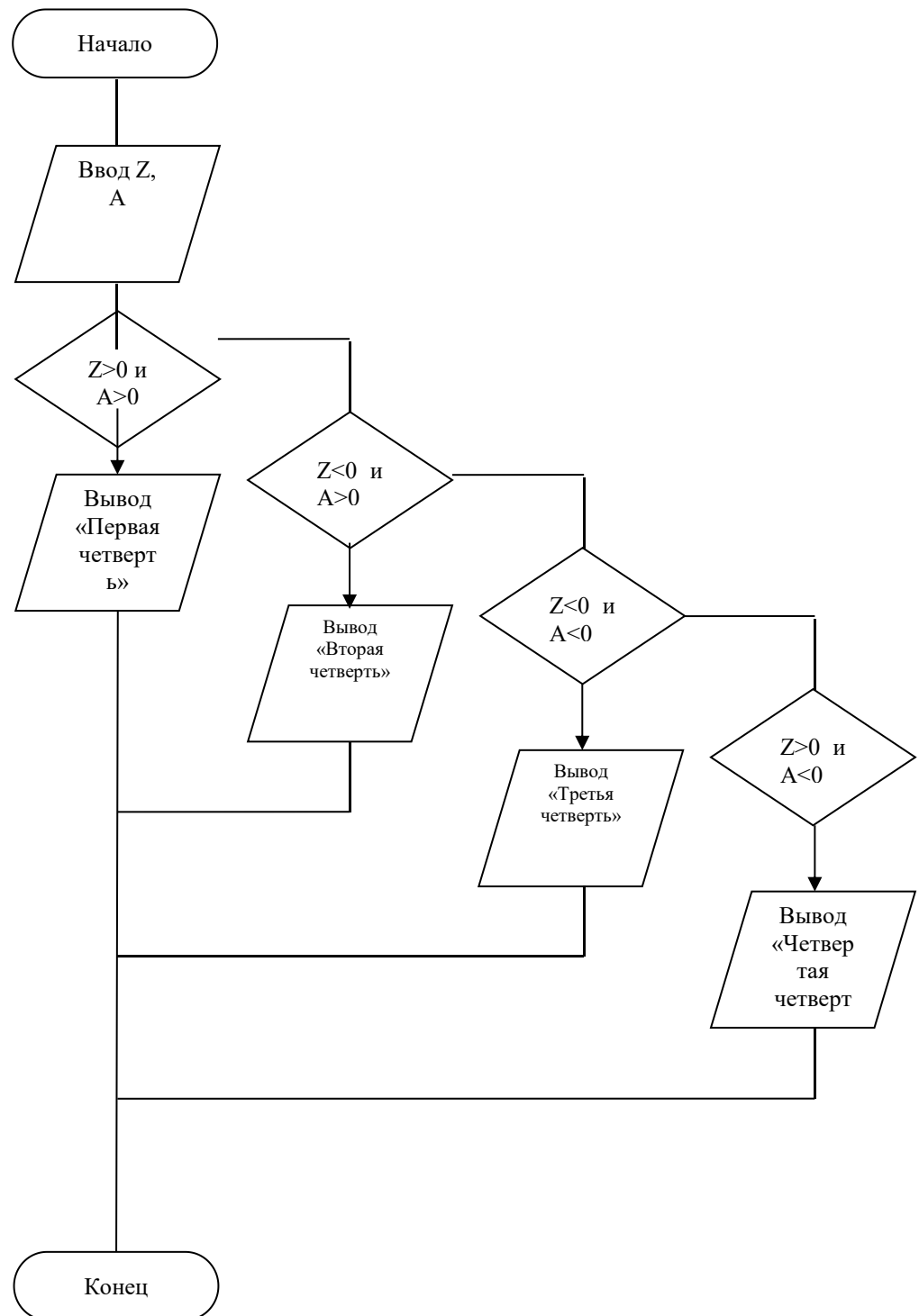
3.1. Используемые переменные

Z, A

3.2. Псевдокод программы

Пользователь вводит x и y. Далее происходит расчет четверти, в которой находится точка. Если $Z > 0$ и $A > 0$, то выводится сообщение «Первая четверть». Если $Z < 0$ и $A > 0$, то выводится сообщение «Вторая четверть». Если $Z < 0$ и $A < 0$, то выводится сообщение «Третья четверть». Если $Z > 0$ и $A < 0$, то выводится сообщение «Четвертая четверть».

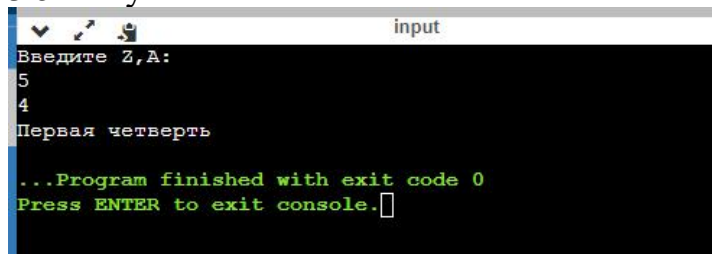
3.3. Блок схема алгоритма



3.4. Листинг программной реализации

```
#include <stdio.h>
int main()
{
    int Z, A;
    printf("Введите Z,A:\n");
    scanf("%i%i", &Z, &A);
    if(Z>0 && A>0)
        printf("Первая четверть");
    if(Z<0 && A>0)
        printf("Вторая четверть");
    if(Z<0 && A<0)
        printf("Третья четверть");
    if(Z>0 && A<0)
        printf("Четвертая четверть");
    return 0;
}
```

3.5. Результат



3.6. Тесты

| | |
|------|-----------------|
| Z=11 | Вторая четверть |
| A=5 | |

1. Постановка задачи №7 (а)

Дано действительное число a . Вычислить $f(a)$, если

$$f(x) = \begin{cases} 0, & \text{если } x \leq 0, \\ x, & \text{если } 0 < x \leq 1, \\ x^3, & \text{в остальных случаях;} \end{cases}$$

2. Структурные требования

Задано число a . Вычислить $f(i)$, если $f(x)$: - 0, если $x \leq 0$; - x , если $0 < x \leq 1$; - , в остальных случаях.

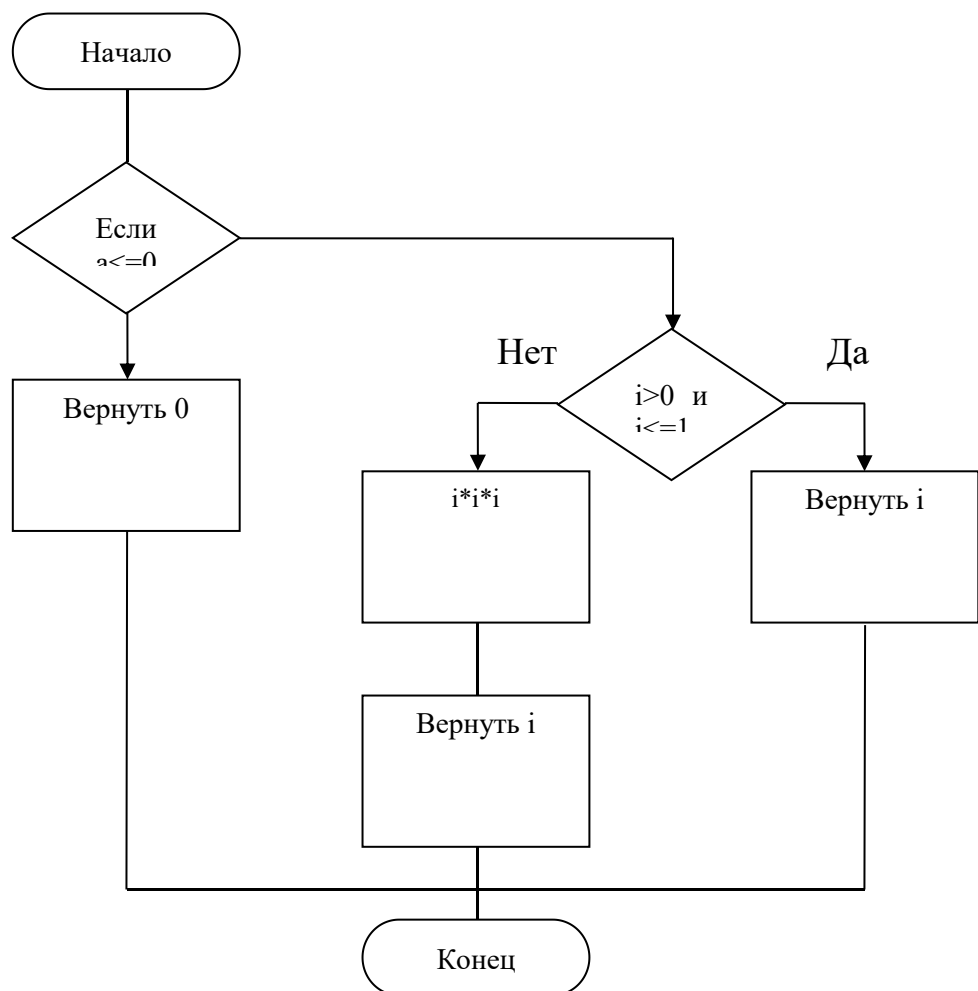
3. Описание работы

3.1. Используемые переменные i

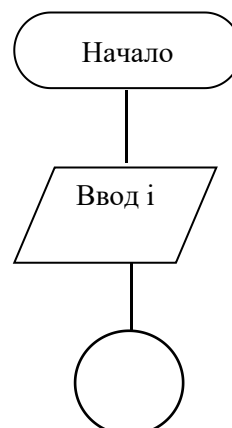
3.2. Псевдокод программы

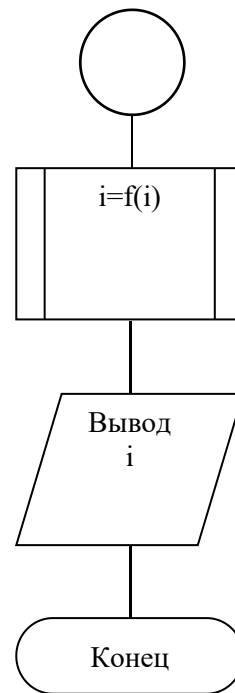
Пользователь вводит i . Если $i \leq 0$, тогда выводится 0. Если $i > 0$ и $i \leq 1$, тогда выводится a , иначе, a возводится в третью степень и выводится.

3.3. Блок-схема



Основная функция:





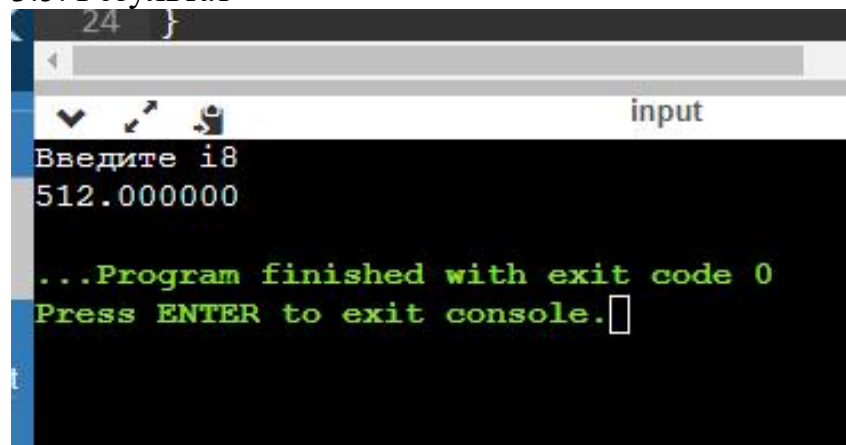
3.4. Листинг Програмной реализации

```
#include <stdio.h>
#include <math.h>
float f (float a);
int main()
{
    float i;
    printf("Введите i");
    scanf("%f", &i);
    printf("%f", f(i));
    return 0;
}

float f(float i){
    if(i<=0){
        return 0;
    }

    if(i>0 && i<=1) {
        return i;
    } else{
        i = pow(i, 3);
        return i;
    }
}
```

3.5. Результат



```
24 }  
input  
Введите i8  
512.000000  
...Program finished with exit code 0  
Press ENTER to exit console.█
```

3.6. Тесты

| | |
|-------------|---------------|
| Введите i=9 | I=729.0000000 |
| Введите i=6 | i=216.0000000 |
| Введите i=3 | i=27.000000 |

1. Постановка задачи №7 (б)

Дано действительное число a . Вычислить $f(a)$, если

$$f(x) = \begin{cases} 0, & \text{если } x \leq 0, \\ x^2 + x, & \text{если } 0 < x \leq 1, \\ x^2 - \sin(\pi x), & \text{в остальных случаях.} \end{cases}$$

2. Структурные требования

Задано число a . Вычислить $f(a)$, если $f(x)$: - 0, если $x \leq 0$; - $+x$, если $0 < x \leq 1$; - $-\sin()$, в остальных случаях.

3. Описание работы

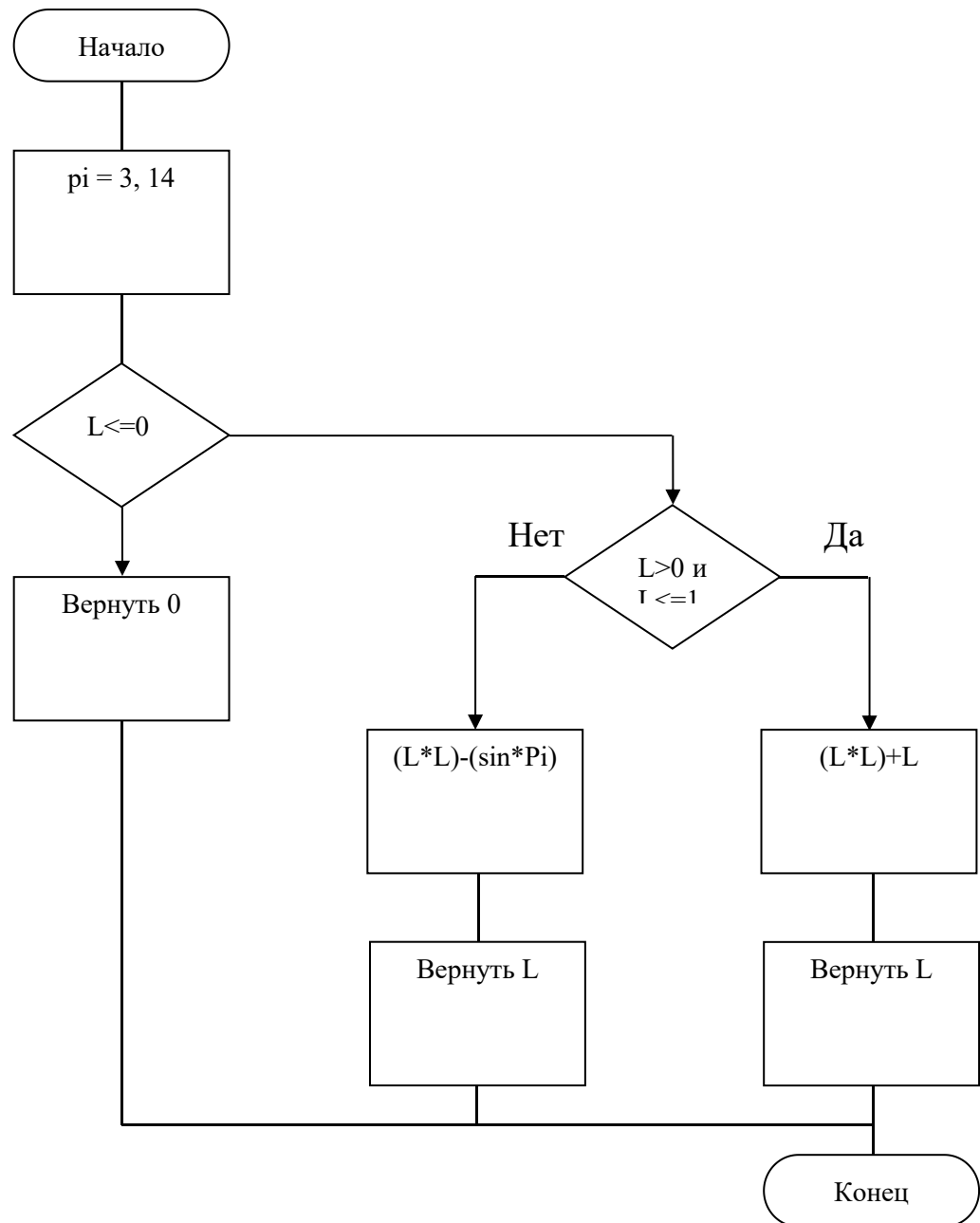
3.1. Используемые переменные

L , Pi

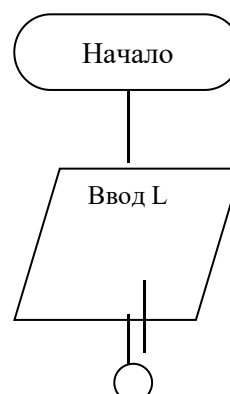
3.2. Псевдокод программы

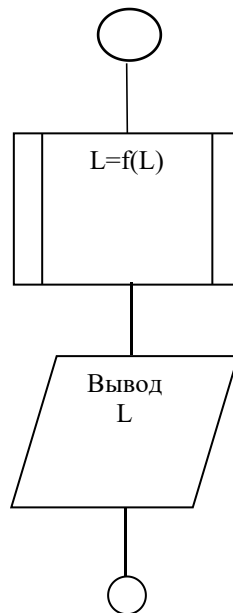
Пользователь вводит a . Если $L \leq 0$, тогда выводится 0. Если $L > 0$ и $L \leq 1$, тогда a возводится в квадрат и еще прибавляется $(L^2 + L)$, иначе, a возводится в квадрат и из a вычитается $\sin(L * Pi)$

3.3. Блок схема алгоритма



Блок-схема основной функции:





3.4. Листинг программной реализации

```

#include <stdio.h>
#include <math.h>
float f (float L);
int main()
{
    float L;
    printf("Введите L");
    scanf("%f", &L);
    printf("%f", f(L));

    return 0;
}

float f(float L){
    const float Pi = 3.14;
    if(L<=0){

        return 0;
    }

    if(L>0 && L<=1) {
        L = pow(L, 2)+L;
        return L;
    }
    else{

```



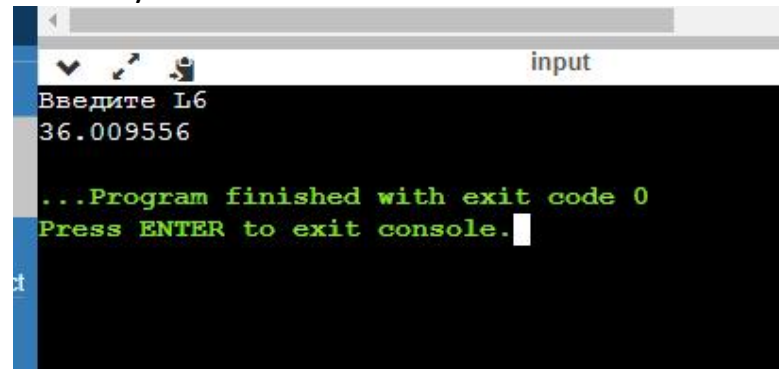
```

        L = pow(L, 2) - sin(L*Pi);

    return L;
}
}

```

3.5. Результат



3.6. Тесты

| | |
|-------------|-------------|
| Введите L=9 | L=80.985664 |
| Введите L=3 | L=8.995222 |

Список литературы

1. Конова Е.А., Поллак Г.А. Алгоритмы и программы. Язык С++: Учебное пособие. - 2-е изд., стер. - СПб.: Издательство "Лань", 2017. - 384 с.
2. Седжвик Роберт. Алгоритмы на С++.: Пер. с англ. - М.: ООО "И.Д. Вильямс", 2011. - 1056 с.
3. Лафоре Р. Объектно-ориентированное программирование в С++. Классика Computer Science. 4-е изд. - СПб.: Питер, 2015. - 928 с.
4. Орлов С.А. Теория и практика языков программирования: Учебник для вузов. Стандарт 3-го поколения. - СПб.: Питер, 2014. - 688 с.
5. Павловская Т.А. С/С++. Процедурное и объектно-ориентированное программирование: Учебник для вузов. Стандарт 3-го поколения. - СПб.: Питер, 2015. - 496 с.
6. Павловская Т.А. С#. Программирование на языке высокого уровня: Учебник для вузов. - СПб.: Питер, 2014. - 432 с.
7. Плаксин М.А. Тестирование и отладка программ - для профессионалов будущих и настоящих. - М.: БИНОМ. Лаборатория знаний, 2013. - 167 с.