Код программы без switch

#include <iostream> // Подключение стандартной библиотеки

#include <string>

using namespace std;

int numArr[1000], cnt;

int main() //Основное тело программы

{

char arr[1000] = { ' ' };

int currNum = 0;

int k = 1;

cout << "Enter a numeric string: "; // Вывод сообщения

char c = getchar();

while (c != '\n') {

arr[k++] = c;

c = getchar();

}

cout << '\n'; // Перевод каретки

int q = 1;

for (int i = k; i >= 0; i--) {

if (isdigit(arr[i])) {

currNum += q \* (arr[i] - '0');

q \*= 10;

}

else if (q != 1) {

numArr[cnt++] = currNum;

currNum = 0;

q = 1;

}

}

}

Код программы c switch

#include <iostream> // Подключение стандартной библиотеки

#include <string>

using namespace std;

int numArr[1000], cnt;

int main() //Основное тело программы

{

char arr[1000] = { ' ' };

int currNum = 0;

int k = 1;

cout << "Enter a numeric string: "; // Вывод сообщения

char c = getchar();

while (c != '\n') {

arr[k++] = c;

c = getchar();

}

cout << '\n'; // Перевод каретки

int q = 1;

for (int i = k; i >= 0; i--) {

if (isdigit(arr[i])) {

currNum += q \* (arr[i] - '0');

switch (currNum)

{

case 1:

cout << "1";

break;

case 2:

cout << "2";

break;

case 3:

cout << "3";

break;

default:

cout << "0";

break;

}

q \*= 10;

}

else if (q != 1) {

numArr[cnt++] = currNum;

currNum = 0;

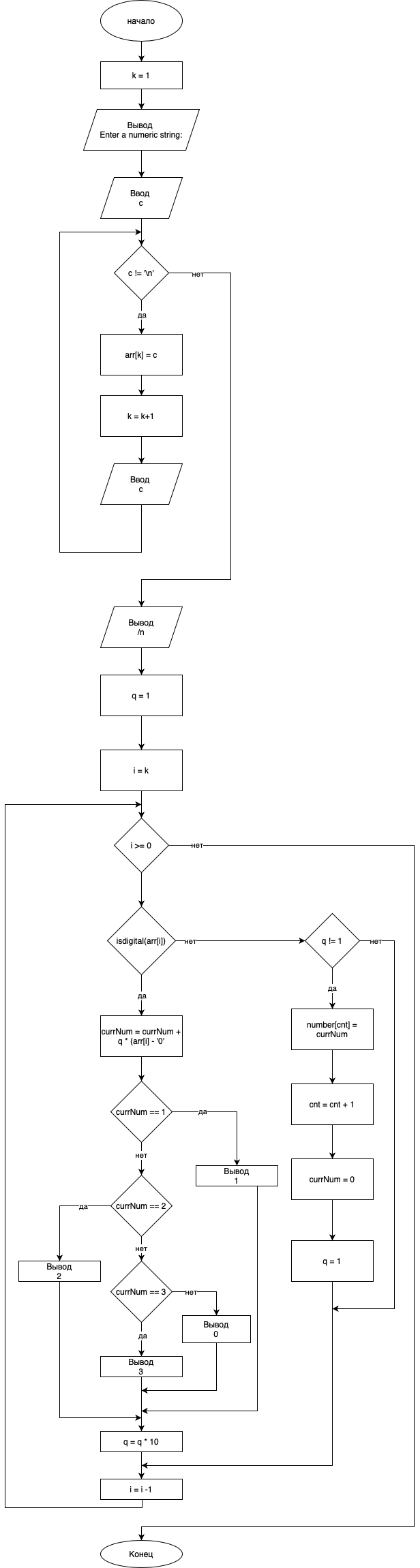
q = 1;

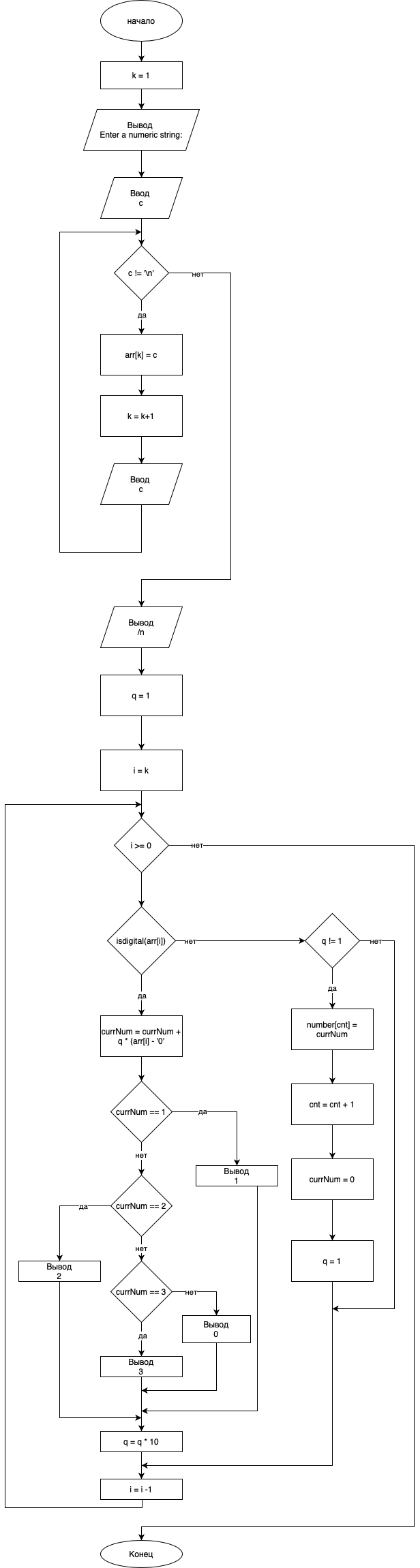
}

}

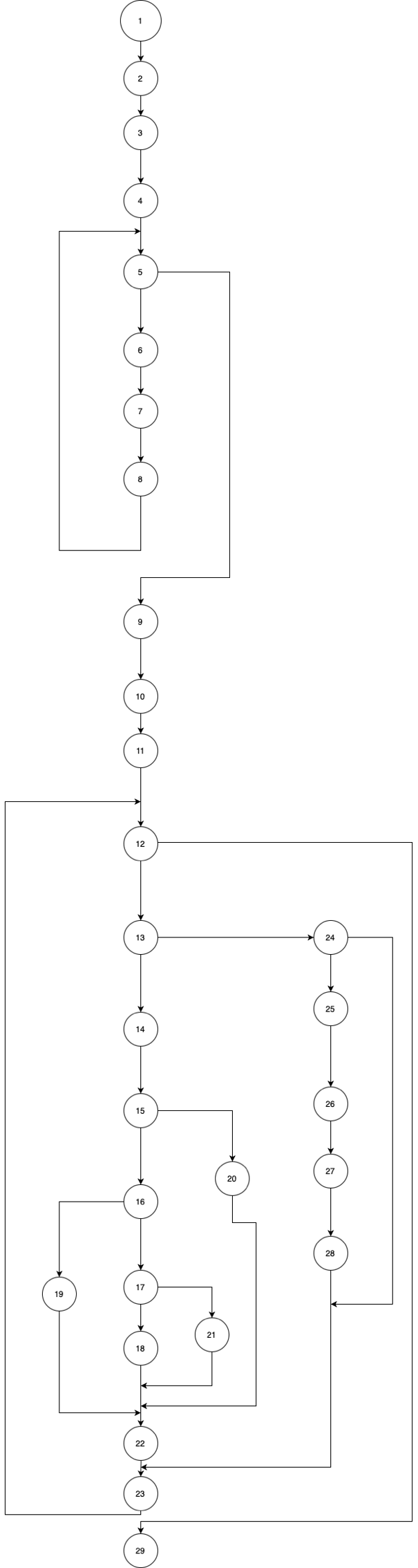
}

**БЛОК-СХЕМА**





**ГРАФ**



**Метрика Маккейба**

путь: 1-2-3-4-5(нет)-9-10-11-12(нет) -29

Количество вершин(v) = 29;

Количество дуг(e) = 35;

Компонент связности(p) = 1;

Z(G) = e – v + 2p = 35-29 + 2\*1 = 8(базисных независимых путей);

**Метрика Джилба**

CL=7(количество условных операторов, характеризующее абсолютную сложность программы)

сl=CL/(общее число операторов)=7/27=0.259(насыщенность программы условными операторами, характеризующее относительную сложность программы)

CLI=4(вершина 13 вложена в 12), (вершина 15 вложена в 13), (вершина 16 вложена в 15), (вершина 17 вложена 16). Максимальный уровень вложенности условного оператора.