**1.3.1.9**

**Состав данных**

**Входные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| Z | вещественный | п/п | Начальное состояние счета |
| Y | вещественный | п/п | Величина заработной платы |
| X | вещественный | п/п | Величина коммунальных платежей |
| p | вещественный | п/п | Процент для остатка счета |

**Выходные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| i | целый | п/п | Счетчик месяцев |
| S | вещественный | п/п | Состояние счета в начале каждого месяца |

**Входная форма**

|  |
| --- |
| Введите изначальное состояние счета:  <Z>  Введите заработную плату:  <Y>  Введите коммунальные платежи:  <X>  Введите процент для остатка: <p> |

**Выходная форма**

|  |
| --- |
| Месяц Баланс счета  <i1> <S1>  <i2> <S2>  … …  <i12> <S12> |

**Код программы**

#include <iostream>

using namespace std;

int main()

{

setlocale(0, "");

float Z, Y, X, p, S;

cout << "Введите изначальное состояние счета:\n";

cin >> Z;

cout << "Введите заработную плату:\n";

cin >> Y;

cout << "Введите коммунальные платежи:\n";

cin >> X;

cout << "Введите процент для остатка:\n";

cin >> p;

cout << "Месяц\t\tБаланс счета\n" << endl;

S = Z + Y - X;

int i = 1;

cout << i << "\t\t" << S << " руб." << endl;

S = S + S \* p / 100;

for (int i = 2; i <= 12; i += 1) {

S = S + Y - X;

cout << i << "\t\t" << S << " руб." << endl;

S = S + S \* p / 100;

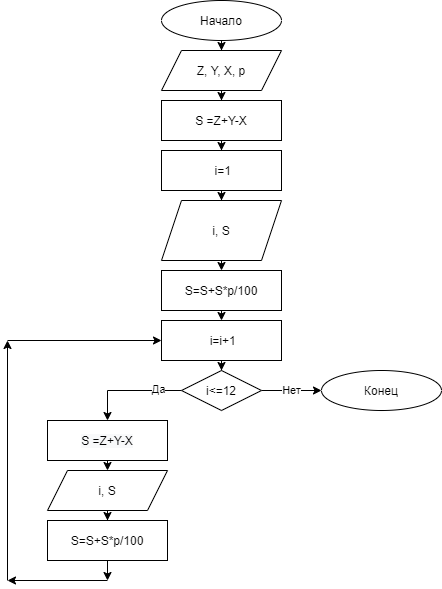
}

system("pause");

return 0;

}

**Блок-схема**

****

**1.3.2.9**

**Состав данных**

**Входные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| Z | вещественный | п/п | Начальное состояние счета |
| Y | вещественный | п/п | Величина заработной платы |
| X | вещественный | п/п | Величина коммунальных платежей |
| p | вещественный | п/п | Процент для остатка счета |

**Выходные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| i | целый | п/п | Счетчик месяцев |
| S | вещественный | п/п | Состояние счета в начале каждого месяца |

**Входная форма**

|  |
| --- |
| Введите изначальное состояние счета:  <Z>  Введите заработную плату:  <Y>  Введите коммунальные платежи:  <X>  Введите процент для остатка: <p> |

**Выходная форма**

|  |
| --- |
| Месяц Баланс счета  <i1> <S1>  <i2> <S2>  … …  <i12> <S12> |

**Код программы**

#include <iostream>

using namespace std;

int main()

{

setlocale(0, "");

float Z, Y, X, p, S;

cout << "Введите изначальное состояние счета:\n";

cin >> Z;

cout << "Введите заработную плату:\n";

cin >> Y;

cout << "Введите коммунальные платежи:\n";

cin >> X;

cout << "Введите процент для остатка:\n";

cin >> p;

cout << "Месяц\t\tБаланс счета\n" << endl;

S = Z + Y - X;

int i = 1;

cout << i << "\t\t" << S << " руб." << endl;

S = S + S \* p / 100;

for (int i = 2; i <= 12; i += 1) {

X = X \* 1.01;

S = S + Y - X;

cout << i << "\t\t" << S << " руб." << endl;

S = S + S \* p / 100;

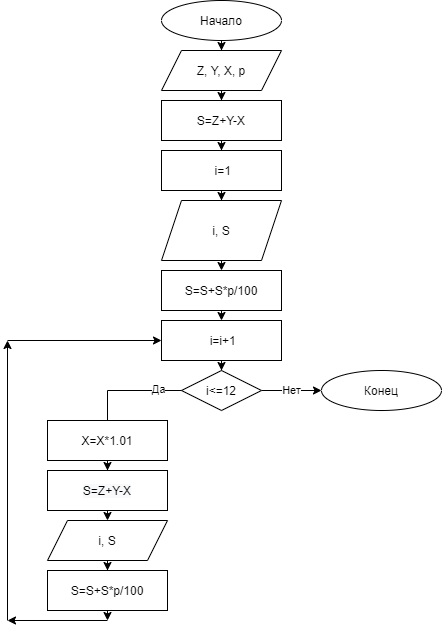
}

system("pause");

return 0;

}

**Блок-схема**

****

**1.3.3.9**

**Состав данных**

**Входные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| Z | вещественный | п/п | Заработная плата работника |
| p | вещественный | п/п | Процент для остатка счета |

**Выходные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| S | вещественный | п/п | Баланс счета после года |

**Промежуточные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Тип | Структура | Смысл |
| i | целый | п/п | Счетчик месяцев |

**Входная форма**

|  |
| --- |
| Введите заработную плату: <Z>  Введите процентную ставку: <p> |

**Выходная форма**

|  |
| --- |
| Счет в конце года:  <S> |

**Код программы**

#include <iostream>

using namespace std;

int main()

{

setlocale(0, "");

float Z, p, S = 0;

cout << "Введите заработную плату:\n";

cin >> Z;

cout << "Введите процентную ставку:\n";

cin >> p;

for (int i = 1; i <= 12; i += 1) {

S = S + Z / (i + 1);

S = S + S \* p / 100;

}

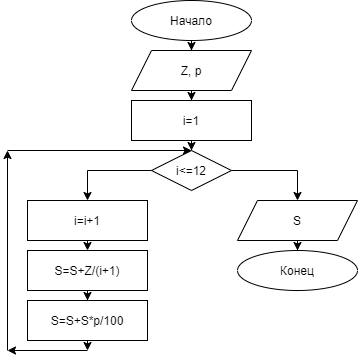
cout << "Счет в конце года: "<< endl << S << endl;

system("pause");

return 0;

}

**Блок-схема**

****