Цикл с условием

# Зависимости.

## Для выполнения данной работы требуются следующие библиотеки:

1. #include <iostream>

# Задание 1.

1. **void** task\_1(){
2. /\*Мистер Пипкин и Мистер Липкин одновременно внесли в свои
3. банки одинаковые вклады по V долларов. Банк мистера Пипкина
4. начисляет своим клиентам 10% годовых, а банк мистера Липкина – 7%.
5. Во сколько раз больше денег окажется через 10 лет на счете у
6. мистера Пипкина, чем у мистера Липкина?\*/
8. **int** years = 10;
9. **float** present\_1 = 0.1, present\_2 = 0.07;
11. **int** V; cout << "How many you puts on depozite? : "; cin >> V;
12. **double** divid\_1 = V, divid\_2 = V;
14. **int** i = 0;
15. **while**( i < years ){
16. i++;
17. divid\_1 += divid\_1 \* present\_1;
18. divid\_2 += divid\_2 \* present\_2;
19. }
20. cout << divid\_1 << endl;
21. cout << divid\_2 << endl;
23. cout << "Dividents Ms. Pipkin > Dividents Ms. Lipkin in: " << divid\_1 / divid\_2 << endl;
24. }

# Задание 2

1. **void** task\_2(){
2. /\*В водоеме 100 т рыбы. Каждый год рыболовецкая бригада вылавливает 15 т.
3. Воспроизводство рыбы 5 % в год.
4. Для сохранения воспроизводства необходимо прекращать лов,
5. когда в водоеме ее остается менее 5 т.
6. Через сколько лет лов рыбы должен быть прекращен?\*/
8. **double** fish = 100;
9. **int** fishers = 15, preset = 0.05;
11. **int** i = 0;
12. **while**(fish > 5){
13. fish += fish \* preset;
14. fish -= fishers;
15. i++;
16. }
18. i++;
19. cout << i << " years ago" << endl;
20. }

# Задание 3

1. **void** task\_3(){
2. /\*Малыш и Карлсон живут в прямоугольной комнате.
3. Как им подсчитать, сколько понадобится квадратных
4. ковриков со стороной 1х1 метр, чтобы полностью покрыть
5. пол комнаты (ни Малыш, ни Карлсон не умеют ни делить, ни умножать).\*/
7. **int** a, b, S = 0; cout << "Input S (width and length: "; cin >> a >> b;
9. **int** i = 0;
10. **while**(b > 0){
11. S += a; b--;
12. }
14. cout << "Them need " << S << " kovrocov" << endl;
15. }

# Задание 4

1. **void** task\_4(){
2. /\*Царевна-лягушка съедает ежедневно на К% комаров больше,
3. чем в предыдущий день, и еще два комара.
4. Составить алгоритм, определяющий, через сколько дней
5. количество съеденных комаров превысит М, если в первый
6. день было съедено Х комаров.\*/
8. **double** K;
10. **int** M, X; cout << "Input M, X and K: "; cin >> M >> X >> K;
11. K \*= 0.01;
13. **int** d = 0;
14. **while**(X <= M){
15. d++; X += X \* K + 2;
16. }
17. cout << d << " day ago..." << endl;
18. }

# Дополнительное задание 1

1. **void** add\_task\_1(){
2. /\*Натуральное число из n цифр является числом Армстронга,
3. т.е. сумма его цифр, возведенная в n степень,
4. равна самому числу (153=1\*1\*1+5\*5\*5+3\*3\*3).
5. Получите все числа Армстронга для n=3 и n=4.\*/
6. **int** num, max, Amst;
7. **int** n; cout << "Input n (3 or 4): "; cin >> n;
9. **bool** f = **false**;
10. **if**(n == 3){ num = 100;  max = 1000;  f = **true**;}
11. **if**(n == 4){ num = 1000; max = 10000;          }
13. **if**(n < 3 || n > 4){ cout << "You input are wrong" << endl; }
15. **int** a,b,c,d;
16. **while**(num < max){
17. **if**(f){
18. a = num / 100;
19. b = num / 10 % 10;
20. c = num % 10;
22. Amst = (a \* a \* a) + (b \* b \* b) + (c \* c \* c);
23. }
24. **else**{
25. a = num / 1000;
26. b = num / 100 % 10;
27. c = num % 100 / 10;
28. d = num % 10;
30. Amst = (a \* a \* a \* a) + (b \* b \* b \* b) + (c \* c \* c \* c) + (d \* d \* d \* d);
31. }
33. **if** (num == Amst){ cout << num << endl; }
34. num++;
35. }
36. }

# Дополнительное задание 2

1. **void** add\_task\_2(){
2. /\*Дано натуральное число А.
3. Как наименьшим количеством монет можно выплатить А рублей.
4. Предполагается, что в достаточном количестве имеются монеты
5. достоинством в 1, 3, 5, 10, 15, 20 и 50 рублей.\*/
7. **double** A; cout << "Input cash: "; cin >> A;
8. **int** one = 50, summ = 0;
9. **while** (A > 0){
10. **switch** (one){
11. **case** 50:
12. summ += A / one;
13. cout << " 50-th money: " << A / one << " tg" << endl;
14. A -= one \* (A / 50); one = 20;
15. **break**;
17. **case** 20:
18. summ += A / one;
19. cout << one << "-th money: " << A / one << " tg" << endl;
20. A -= one \* (A / one); one = 15;
21. **break**;
23. **case** 15:
24. summ += A / one;
25. cout << one << "-th money: " << A / one << " tg" << endl;
26. A -= one \* (A / one); one = 10;
27. **break**;
29. **case** 10:
30. summ += A / one;
31. cout << one << "-th money: " << A / one << " tg" << endl;
32. A -= one \* (A / one); one = 5;
33. **break**;
35. **case** 5:
36. summ += A / one;
37. cout << one << "-th money: " << A / one << " tg" << endl;
38. A -= one \* (A / one); one = 3;
39. **break**;
41. **case** 3:
42. summ += A / one;
43. cout << one << "-th money: " << A / one << " tg" << endl;
44. A -= one \* (A / one); one = 1;
45. **break**;
47. **case** 1:
48. summ += A / one;
49. cout << one << "-th money: " << A / one << " tg" << endl;
50. A -= one \* (A / one); one = 1;
51. **break**;
52. }
53. }
54. cout << "All " << summ << " money" << endl;
55. }

# Сложное задание

1. **void** hard\_task(){
2. /\*Незнайка считает, что операция «увеличить» число
3. означает увеличить каждую его цифру,
4. а может быть еще при этом приписать одну ненулевую цифру слева.
5. У Незнайки сегодня четверг (он по четвергам занимается математикой) и
6. он хочет увеличить все увиденные им числа в 4 раза,
7. используя свой способ «увеличения». Помогите Незнайке.\*/
8. **int** sub = 10;
9. **int** num; cout << "Input num: "; cin >> num;
11. **int** i = 0;                          // Определение длины числа
12. **while**(**true**){
13. **if**( num % sub == num ){ i++; **break**; }
14. **else**{ i++; sub \*= 10; }
15. }
17. i++; **int** ms[i], l = 1; sub = num;    // Разбиение числа на цифры
18. **while**(i >= l){
19. ms[l] = sub % 10;
20. sub /= 10;
21. l++;
22. }
24. l = 1; **int** add = 0; **bool** f = **true**;  // Умножение
25. **while** (i >= l){
26. **if** ( ms[l] >= 5 ) { cout << "nevojmojno" << endl; f = **false**; **break**; }
27. **else** {
28. **if** ( ms[l] \* 4 > 9 && l == i ){
29. ms[l] = (ms[l] \*= 4) % 10 + add;
30. l++;
31. ms[l] = 1;
32. **break**;
33. }
35. **if** ( ms[l] \* 4 > 9 && l != i ){
36. ms[l] = (ms[l] \*= 4) % 10 + add;
37. l++;
38. add = 1;
39. **continue**;
40. }
42. ms[l] = (ms[l] \* 4) + add;
43. l++;
44. }
45. }
47. **if**(f){
48. l = i;                              // Цикл вывода информации
49. **while** (l >= 1){
50. **if**(l == i && ms[l] == 0){ l--; **continue**; }
51. cout << ms[l];
52. l--;
53. }
54. }
55. cout << endl;
56. }