

```

//Закръгляне
double round(double x)
{ return floor(x+0.5); }

//Време в стандартен формат
char *format(double t,char *r)
{ double h,m,s;
  float p=modf(t,&h);
  p=round(p*3600);
  s=fmod(p,60);
  m=floor(p/60);
  sprintf(r,"%0f:%0f:%0f",h,m,s);
  return r;
}

//Намиране на минимално време
double minTime(void)
{ double m,min,x,st;
  Point R;
  R.x=0;R.y=w/2;
  min=getTime(R);
  st=x=0.1; //"едра" стъпка
  do
  { do
    { R.x=x;
      m=getTime(R);
      if (m<=min) min=m;
      else break;
      x+=st;
    }while (x<=s/2);
    x-=2*st;
    st/=10; // намаляване на стъпката 10 пъти
    R.x=x;
    m=getTime(R);
    if(fabs(m-min)<0.00001) break;//край
    min=m;
  }while(1);
  return 2*min;
}

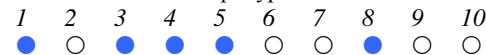
int main (void)
{ char buf[16];
  scanf("%lf %lf %lf %lf",&w,&s,&a,&b);
  A.x=0;A.y=w;
  P.x=s/2;P.y=w/2; //координати на точките А и Р.
  printf("%s\n",format(minTime(),buf));
  return 0;
}

```

Задача С3. ОГЪРЛИЦИ, автор Павлин Пеев

Фирма „Гери и Дани“ има утвърден оригинален патент за мънистени огърлици в класическите два цвята – синьо и бяло. Освен интересната закопчалка на винт, към патента е включен и уникален метод за подредба

на мънистата, който е и гаранция за автентичност. Нека при разкопчана огърлица с дължина n да номерираме мънистата, отляво надясно, с целите числа от 1 до n , както е показано на фигурата:



Патентованата подредба се подчинява на следните две правила:

1. Най-лявото мънисто (това с номер 1) е синьо.
2. Нека сините мъниста с нечетни номера са p на брой, а сините мъниста с четни номера – q на брой. Разглеждаме абсолютната стойност d на разликата на числата p и q , $d = |p - q|$. Тя е цяло неотрицателно число. Патентът гласи, че правилна е тази подредба, за която d се дели на 3 без остатък.

Показаната по-горе огърлица, например, не е правилно подредена: тя изпълнява първото условие (мънисто 1 е синьо), но при нея, както се вижда, $p = 3$, $q = 2$, т. е. $d = 1$ и d не се дели на 3. Напротив, огърлиците и например, отговарят и на двете условия.

От фирмата искат да знаят колко са правилните огърлици с определена дължина (брой мъниста). Помогнете им, като напишете програма **NECKLACE**, която решава задачата.

Вход

Програмата трябва да въведе от единствения ред на стандартния вход дължината n ($1 \leq n \leq 60$) на огърлицата.

Изход

Програмата трябва да изведе броя на различните правилни огърлици с дължина n на стандартния изход.

ПРИМЕР

Вход	Изход
6	11

Обяснение: Правилните огърлици с дължина 6 са: , , , , , , и .

Решение

Очевидна е връзката с двоични числа. Ако кодираме синьото мънисто с 1, а бялото – с 0, то зададените правила описват n -цифрените двоични числа без водещи нули, които се делят на 3. Наистина, ако с a_i означим двоичните цифри на едно число A , то

$A = a_1a_2a_3\dots a_n = a_1 \cdot 2^{n-1} + a_2 \cdot 2^{n-2} + a_3 \cdot 2^{n-3} + \dots + a_n$. Като имаме предвид, че $2 \equiv -1 \pmod{3}$, то четните степени в това представяне могат да се заместят с 1, а нечетните – с -1 по модул 3. Тогава е ясно, че един признак за делимост на 3 в двоична бройна система е делимостта на 3 на алтернативната сума от двоичните цифри $a_1 - a_2 + a_3 - a_4 + \dots$. Това изказване всъщност е еквивалентно на формулировката в задачата. Ще припомним само, че, поради аналогични причини, подобно изглежда и известният признак за делимост на 11 в десетична бройна система. И естествено, защото $11_2 = 3$.