

SNTP

Для того чтобы компьютеры поддерживали актуальное время, они могут обращаться к серверам точного времени SNTP (Simple Network Time Protocol). К сожалению, компьютер не может просто получить время у сервера, потому что информация по сети передаётся не мгновенно: пока сообщение с текущим временем дойдёт до компьютера, оно потеряет свою актуальность. Протокол взаимодействия клиента (компьютера, запрашивающего точное время) и сервера (компьютера, выдающего точное время) выглядит следующим образом:

- 1) Клиент отправляет запрос на сервер и запоминает время отправления A (по клиентскому времени).
- 2) Сервер получает запрос в момент времени B (по точному серверному времени) и отправляет клиенту сообщение, содержащее время B .
- 3) Клиент получает ответ на свой запрос в момент времени C (по клиентскому времени) и запоминает его. Теперь клиент, из предположения, что сетевые задержки при передаче сообщений от клиента серверу и от сервера клиенту одинаковы, может определить и установить себе точное время, используя известные значения A, B, C .

Вам предстоит реализовать алгоритм, с точностью до секунды определяющий точное время для установки на клиенте по известным A, B и C . При необходимости округлите результат до целого числа секунд по правилам арифметики (в меньшую сторону, если дробная часть числа меньше $\frac{1}{2}$, иначе в большую сторону).

Возможно, что, пока клиент ожидал ответа, по клиентскому времени успели наступить новые сутки, однако известно, что между отправкой клиентом запроса и получением ответа от сервера прошло менее 24 часов.

Программа получает на вход три временные метки A, B, C , по одной в каждой строке. Все временные метки представлены в формате «hh:mm:ss», где «hh» – это часы, «mm» – минуты, «ss» – секунды. Часы, минуты и секунды записываются ровно двумя цифрами каждое (возможно, с дополнительными нулями в начале числа).

Программа должна вывести одну временную метку в формате, описанном во входных данных, – вычисленное точное время для установки на клиенте. В выводе не должно быть пробелов, пустых строк в начале вывода.

Пример входных и выходных данных

Ввод	Вывод	Примечание
15:01:00 18:09:45 15:01:40	18:10:05	Клиент отправил запрос в 15:01:00 по своим часам, сервер получил запрос в 18:09:45 по своим часам. Клиент получил ответ в 14:01:40, в этот момент точное время будет 18:10:05.

Ответ:

```
program class10v2;  
var
```

```
A,B,C : string[8];
Asec,Bsec,Csec,netDelay : longint;
```

```
function StrToSec (s:string[8]) : longint;
var
  str : string[2];
  Sec : longint;
  i,err : integer;
begin
  val(copy(s,1,2),i,err);
  if err = 0 then begin
    Sec:=i*3600;
    val(copy(s,4,2),i,err);
    if err = 0 then begin
      Sec:=Sec+i*60;
      val(copy(s,7,2),i,err);
      if err = 0 then
        StrToSec:=Sec+i
      else StrToSec:=-1;
      end
    else StrToSec:=-1;
    end
  else StrToSec:=-1;
end;
```

```
function SecToStr(Sec:longint):string[8];
var
  hh,mm,ss:string;
  h,m,s:integer;
begin
  h:=Sec div 3600;
  if h>=24 then h:=h-24;
  Sec:=Sec mod 3600;
  str(h,hh);
  if h<10 then hh:='0'+hh;
  m:=Sec div 60;
  str(m,mm);
  if m<10 then mm:='0'+mm;
  s:=Sec mod 60;
  str(s,ss);
  if s<10 then ss:='0'+ss;
  SecToStr:=hh+':'+mm+':'+ss;
end;
```

```
begin
  readln(A);
  readln(B);
  readln(C);
  Asec:=StrToSec(A);
  Bsec:=StrToSec(B);
  Csec:=StrToSec(C);
  if (Asec<>-1)and(Bsec<>-1)and(Csec<>-1) then begin
```

```
if Csec<Asec then
  netDelay:=round(0.5*(24*3600+Csec-Asec))
else
  netDelay:=round(0.5*(Csec-Asec));
writeln(SecToStr(BSec+netDelay));
end
else
  writeln('Ошибка ввода!');
end.
```