Raju forgot his ID card on the Xth floor of the building he is presently on the Yth floor. There are lifts which can only be boarded on the ground floor which move one floor up every S seconds (assume ground floor to be floor zero). Raju can climb up or down a floor every T seconds what is the least amount of time it will take Raju to get to his ID card.

**Input Format**

The first line and only line consists of 4 space separated integers X Y S T

**Constraints**

1<=X<=10000 1<=Y<=10000 1<=S<=1000 1<=T<=1000

**Output Format**

Output consists of a single integer the minimum time it will take Raju to get to his ID card

**Sample Input 0**

15 3 3 6

**Sample Output 0**

63

**Explanation 0**

It is optimal to walk down 3 floors and then take the lift up to the 15th floor giving us the output 63

#include <cmath>

#include <cstdio>

#include <vector>

#include <iostream>

#include <algorithm>

using namespace std;

int main() {

int x,y,s,t,u,u1,u2,t1,t2;

cin>>x>>y>>s>>t;

u=abs(x-y);

u1=x\*s;

u2=y\*t;

t1=u1+u2;

t2=t\*u;

if(t1<t2)

cout<<t1;

else

cout<<t2;

return 0;

}