Questions from previous years' practice sessions

1. X and Y arrays that contain 8 bit numbers, reside in memory starting from memory addresses $1000 and $1100. Arrays have equal size and their size (n<255) is stored in memory address $0FFF.
   Write a program for instructional CPU that line up array values as X_1, Y_1, X_2, Y_2, …, X_n, Y_n starting from the memory address $1200

2. Write a program for educational CPU that checks whether the square root of an integer is an integer. Square root of an integer can be calculated as:

   ```
   unsigned int x, r;

   /* … */

   x = (x+1);
   x = x/2;
   for(r=0;  x>r; x-=r++);
   ```

   Value of the r calculated after this operations is the square root of x. Integer is in memory address $1000 while the square root should be in $1002. If the square root is an integer, memory address $1001 should be set to $01 otherwise to $00. (Do not use division instruction. r should first be subtracted from x then incremented.)

3. There is an array, A, in the memory that contains integers in two's complementary arithmetic. Starting address of the array is in memory address $100A-$100B, its size (n<255) is in $100C and there are same number (n/2) of even and odd numbers in it. These numbers should be arranged as: (First odd integer in A) (First even integer in A) (second odd integer in A) (second even integer in A) …
   Start address of the new array is in memory address $100D-$100E. Write the program for educational CPU that completes this operation.
   (A should be scanned only once. Values between $100A-$100C should not change at the end of the execution. Only addresses that can be used as temporary storage are $1006, $1007, $1008 and $1009.)

A
E1
O1
O2
O3
E2
O4
:

New Array
O1
E1
O2
E2
O3
E3
: