

## ARP-X Code Description – Works on Vice 3.1 Emulator and C64 Hardware with UII+ cartridge

### Variables:

C – Border Color (53280)

S – Sound Voice 1 Low Note (54272)

Y – Data read variable for arrays (0-9)

H(Y) – High Note array

L(Y) – Low Note array

P – Pattern Choice

I – Iterations Choice

D – Speed Choice

DV – Delay Value based on Speed Choice

R\$ - Keyboard Input

R – Numerical Value of R\$

Z – Iteration Loop

B – Arpeggio Loop

M – Score

T – Note delay Loop

Line 0: Clear Screen: Set C to border color memory location: Set S to Sound Voice 1 Low Note memory location: Set Voice 1 decay to 9: Set Volume to 15: Start Read Data Loop: Read High and Low Note Data into H and L arrays with variable Y: Set Border color to High note Data: Loop back for Data Read Loop: Set border color to 14

Line 1: Clear Screen: Set Character color to 14: Display program title after 17 spaces: Line Space: Display Pattern Choice with variable P: Limit P to 1 or 2

Line 2: Set Background color to Pattern number minus 1 (black or white): Display Iteration choice with variable I: Limit I to 1 through 4: Data for first 5 sets of High and Low Note values

Line 3: Set Character color to Iteration number plus 2: Display Speed Choice with variable D: Limit D to 1 through 9: Data for last 5 sets of High and Low Note values

Line 4: Set DV to  $D^2$  (DV is the delay value between notes): Display Game Instructions: Set Character color to 14: Print Score label: Line Space

Line 5: Wait for input from keyboard with variable R\$: Set R to the numerical value of R\$: Branched Goto based on value of R plus 1

Line 6: Start Iteration Loop: Start Arpeggio Loop: Set value of Y based on keyboard input (R) and step value of Arpeggio loop (B): Move cursor up one line and right one space for adjusting score: Gsub to Note, Color and Score change: Loop back for Arpeggio Loop: Check for Pattern 2 and set Y to Y minus 2 to play fourth note in Pattern 2 (If Pattern 1 is selected, the next two commands are skipped): Move cursor up one line and right one space: Gsub to Note, Color and Score change

Line 7: Loop back for Iteration Loop: Goto Line 5 to wait for another key press to start a new arpeggio iteration or quit: Greetz

Line 8: Set Note High: Set Note Low: Turn on Note with Sawtooth Waveform: Change Border color to Y value: Calculation to increase Score: Print Score: Begin Note Duration Loop: Loop back for Note duration: Turn off Sawtooth Waveform: Return to Gsub in Line 6

Line 9: Set Volume to 0: Set Border to 14: Set Background to 6: Line Space: Display Victory: End