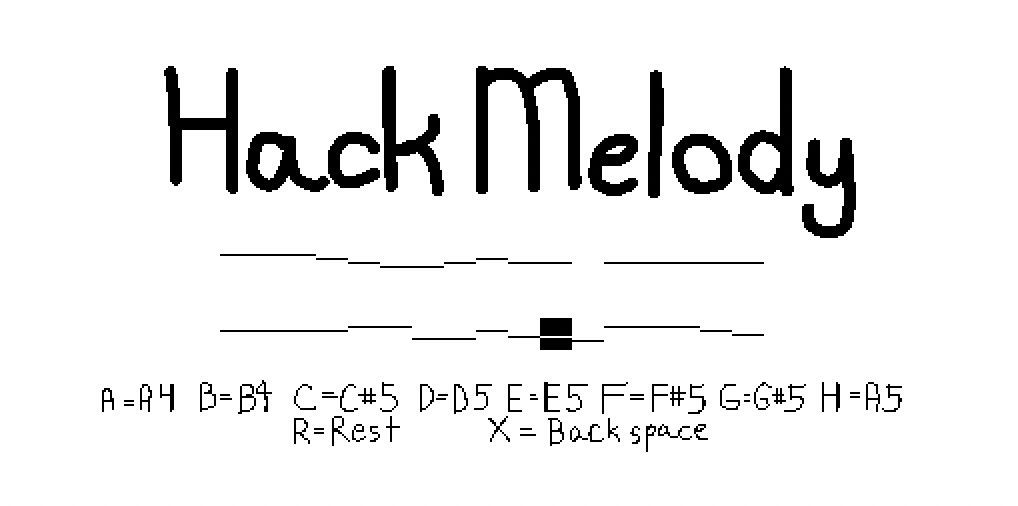
HackMelody

Systems 1 Final Project by Michael Affirim and Cadence Boyce



A mockup sketch

Planning materials spreadsheet: [HackMelody planning](https://docs.google.com/spreadsheets/d/1v16xEc5lpBq9C9Z8J4-9aqDD0OvryGggpTFT6Mba32Q/edit?usp=sharing)

We have 16k memory to work with, 16384 registers. 16000 registers will create 4 seconds of mono 8000 sample rate 8 bit PCM audio (a low quality type of uncompressed audio that is used for telephones) and then 384 registers to work with generating it

So first we make a header right before we start adding samples. That’s 44 bytes (22 registers) that are predefined. <http://soundfile.sapp.org/doc/WaveFormat/>

120 bpm is 2 beats per second so 4000 samples per beat, let’s split it up into 16th notes so that each note is 1000 samples, 500 registers. Take keyboard input and every time someone types in something corresponding to a note, you go from your starting address to 500 addresses after that with the pattern corresponding to the note, and then you change your starting address to be after what you just wrote.

You type in notes by their letter name, and it provides feedback on the screen that is modeled after a piano roll.

A=A4, B=B4, C=C#5, D=D5, E=E5, F=F#5, G=G#5, H=A5, R=rest, X=backspace

This generates square waves because that is the easiest pattern to generate.

The amplitude of the sound is a variable. I don’t intend this to be changeable by the user but we should be able to change it easily to debug it. This number changes how loud the sound is.

A4= 440 Hz. 8000/440=18.182, so about 18 samples per cycle. 9 of these samples will be @amplitude and 9 will be 0. Two samples fit in each register. Set the registers to:

@amplitude @amplitude

@amplitude @amplitude

@amplitude @amplitude

@amplitude @amplitude

@amplitude 0

0 0

0 0

0 0

0 0

And repeat until you have written 1000 samples to memory, 500 registers.

B4= ~494 Hz. 8000/494=16.194 so about 16 samples per cycle. 8 will be @amplitude and 8 will be 0

C#5 = ~554 Hz. 8000/554=14.44 so about 14 samples per cycle. 7 will be @amplitude and 7 will be 0

D5= ~587 Hz per cycle. 8000/587=13.629 so about 13 samples per cycle. 7 will be @amplitude and 6 will be 0

E5 = ~659 Hz. 8000/659=12.14 so about 12 samples per cycle. 6 will be @amplitude and 6 will be 0

F#5 = ~740 Hz. 8000/740=10.811 so about 11 samples per cycle. 6 will be @amplitude and 5 will be 0

G#5 = ~831 Hz. 8000/831=9.627 so about 10 samples per cycle. 5 will be @amplitude and 5 will be 0

A5 = 880 Hz. 8000/880=9.091 so about 9 samples per cycle. 5 will be @amplitude and 4 will be 0

a rest would be all 0s

How to process CPU emulator output to create an audio file once you have completed your melody and stopped the program:

Set the RAM to show as binary

Click on the RAM and do Cmd+A to copy all of it

Paste into a text editor, edit to only be registers 251-16304, and remove register numbers and whitespace

Put this binary data into a binary file, set the file extension to .wav, then open in a compatible sound player application, like Audacity. **Set your computer volume very low so you don’t risk blowing your ears or speakers out!!!**

Pattern that creates an A note

1111 1111 1111 1111

1111 1111 1111 1111

1111 1111 1111 1111

1111 1111 1111 1111

1111 1111 0000 0000

0000 0000 0000 0000

0000 0000 0000 0000

0000 0000 0000 0000

0000 0000 0000 0000

1111 1111 1111 1111

1111 1111 1111 1111

1111 1111 1111 1111

1111 1111 1111 1111

1111 1111 0000 0000

0000 0000 0000 0000

0000 0000 0000 0000

0000 0000 0000 0000

0000 0000 0000 0000

Memory allocation:

| 0 | 15 | R0-R15 |
| --- | --- | --- |
| 16 | 250 | other variables |
| 251 | 272 | header |
| 273 | 773 | note 1 |
| 774 | 1274 | note 2 |
| 1275 | 1775 | note 3 |
| 1776 | 2276 | note 4 |
| 2277 | 2777 | note 5 |
| 2778 | 3278 | note 6 |
| 3279 | 3779 | note 7 |
| 3780 | 4280 | note 8 |
| 4281 | 4781 | note 9 |
| 4782 | 5282 | note 10 |
| 5283 | 5783 | note 11 |
| 5784 | 6284 | note 12 |
| 6285 | 6785 | note 13 |
| 6786 | 7286 | note 14 |
| 7287 | 7787 | note 15 |
| 7788 | 8288 | note 16 |
| 8289 | 8789 | note 17 |
| 8790 | 9290 | note 18 |
| 9291 | 9791 | note 19 |
| 9792 | 10292 | note 20 |
| 10293 | 10793 | note 21 |
| 10794 | 11294 | note 22 |
| 11295 | 11795 | note 23 |
| 11796 | 12296 | note 24 |
| 12297 | 12797 | note 25 |
| 12798 | 13298 | note 26 |
| 13299 | 13799 | note 27 |
| 13800 | 14300 | note 28 |
| 14301 | 14801 | note 29 |
| 14802 | 15302 | note 30 |
| 15303 | 15803 | note 31 |
| 15804 | 16304 | note 32 |