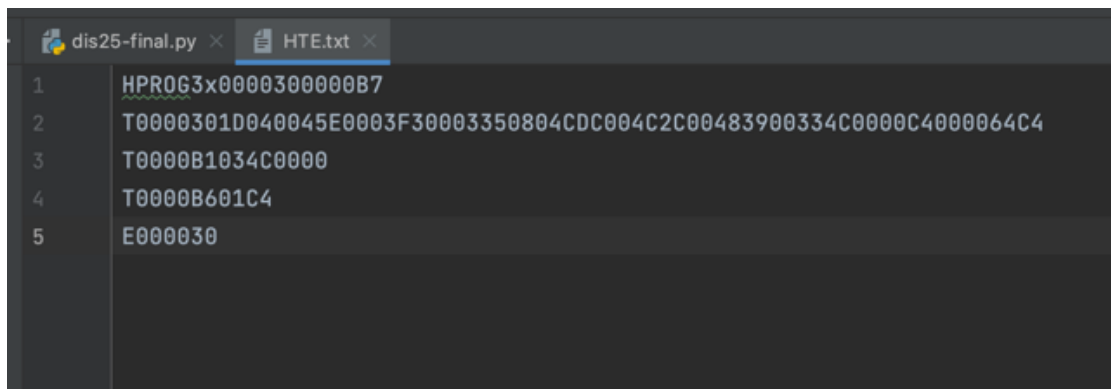


Disassembler Systems Project

Design problems:

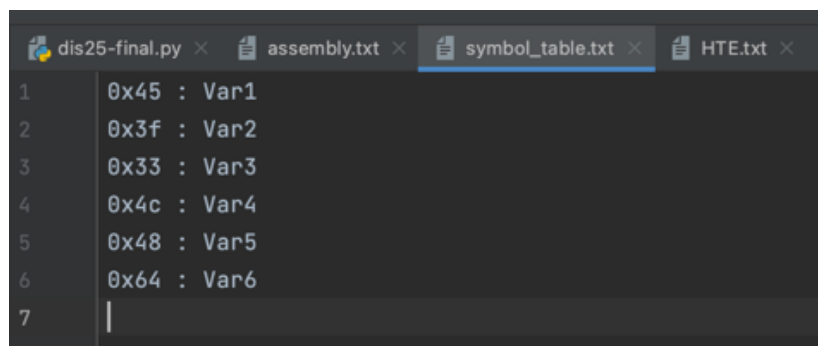
- Bytes, as the byte value may correspond an opcode.
- Word values may correspond labels/location counters.
- Distinguish between RESW/RESB, can only assume it is one type.
- Giving the labels meaningful names.
- There is no way to know what the immediate value in the modified format 3 instructions is, therefore, they are assumed in our code to be '4'.

Sample run:



A screenshot of a disassembler window. The window has two tabs: 'dis25-final.py' and 'HTE.txt'. The 'HTE.txt' tab is active. The window displays a list of assembly instructions with line numbers 1 through 5 on the left. The instructions are: 1. `HPROG3x0000300000B7`, 2. `T0000301D040045E0003F30003350804CDC004C2C00483900334C0000C4000064C4`, 3. `T0000B1034C0000`, 4. `T0000B601C4`, and 5. `E000030`.

```
1  HPROG3x0000300000B7
2  T0000301D040045E0003F30003350804CDC004C2C00483900334C0000C4000064C4
3  T0000B1034C0000
4  T0000B601C4
5  E000030
```



A screenshot of a disassembler window showing a symbol table. The window has four tabs: 'dis25-final.py', 'assembly.txt', 'symbol_table.txt', and 'HTE.txt'. The 'symbol_table.txt' tab is active. The window displays a list of symbols with line numbers 1 through 7 on the left. The symbols are: 1. `0x45 : Var1`, 2. `0x3f : Var2`, 3. `0x33 : Var3`, 4. `0x4c : Var4`, 5. `0x48 : Var5`, 6. `0x64 : Var6`, and 7. `|`.

```
1  0x45 : Var1
2  0x3f : Var2
3  0x33 : Var3
4  0x4c : Var4
5  0x48 : Var5
6  0x64 : Var6
7  |
```

	LOCATION	LINE_LABEL	INSTRUC	TARGET_LABEL	OPCODE
1					
2		PROG3x	START	0030	
3	0030		LDX	Var1	040045
4	0033	Var3	TD	Var2	E0003F
5	0036		JEQ	Var3	300033
6	0039		LDCH	Var4,x	50804C
7	003C		WD	Var4	DC004C
8	003F	Var2	TIX	Var5	2C0048
9	0042		JLT	Var3,4	390033
10	0045	Var1	RSUB		4C0000
11	0048	Var5	FIX		C4
12	0049		LDA	Var6	000064
13	004C	Var4	FIX		C4
14	004D		RESB	100	
15	00B6		FIX		C4
16	00B7		END		
17					