The **focus** of this homework assignment is learning more of RISC-V assembly language programming and, in particular, implementing procedures (functions in C) that use nonlinear recursion using RISC-V ISA features.

```
1   #include <stdio.h>
2   #include <stdint.h>
3   #include <inttypes.h>
4   #define show(x) printf("0x%04" PRIx32 "\t%" PRIu32 "\n", x, x);
5   uint32_t kapow(uint32_t x, uint32_t y) {
6     for(int i = 0; i < y; i++) printf("..");
7     printf(" %" PRIu32 "\n", y);
8     if(y > 1) return kapow(x, y/2) * kapow(x, y / 2 + y % 2);
9     else return (y > 0 ? x : 1);
10  }
11  int main() {
12    uint32_t a, b;
13    scanf("%" SCNu32 "%" SCNu32, &a, &b);
14    show(kapow(a, b));
15    return 0;
16  }
```

As did the previous assignments, this one will also use the $RARSM^1$ environment. The most closely related portions of the textbooks are Chapter 5 of the $RVAP^2$ book and Section 2.8 of the $COAD^3$ book.

The **main programming task** to write an assembly language program corresponding to the above C program (using discussions in class and the discussion forum to clarify details of the correspondence).

Input-output: Mirroring this aspect of the previous assignment, the `hw04` program should read its input from the *standard input* stream and write its output to the *standard output* stream. Optional diagnostics may be written to the *standard error* stream. It is very important that the program read its input only from the standard input stream and that it write nothing except the specified output to the standard output stream.

The **packaging and submission** requirements are similar to those in the previous homework assignments, using the `hw04` tag instead of `hw02`. Similarly, the rules for using additional **resources** are the same as before.

---

[1] Jean Privat and others, RARSM—RISC-V Assembler and Runtime Simulator (iMproved), `https://github.com/rarsm/rars`, 2024.

[2] Robert Winkler, *RISC-V Assembly Programming* (Robert Winkler, 2024).

[3] David A Patterson and John L Hennessy, *Computer Organization and Design RISC-V Edition*, 2nd edition (Morgan Kaufmann, 2020).