1. This exercise is meant to be started during the class meeting and completed individually outside class. **Please work in groups of three to six people.** (If you need help then let me know and I will assist.) List the members of your group below. Underline your name.

2. Briefly explain what the following C program does. In particular, clearly state and explain its output as precisely as possible.

```
1    #include <stdio.h>
2    #include <stdint.h>
3    #include <inttypes.h>
4    int main() {
5      uint32_t x = 9 << 26, y = (5 << 29) | 1;
6      uint64_t z = (uint64_t) x * y;
7      uint32_t zl = (uint32_t) z, zu = (uint32_t) (z >> 32);
8      printf("0x%08" PRIx32 "\n0x%08" PRIx32 "\n", zu, zl);
9      uint64_t d = x, p = 0;
10     uint32_t r = y;
11     for(int i = 0; i < 32; i++, d <<= 1, r >>= 1) if (r & 1) p += d;
12     printf("0x%08" PRIx32 "\n0x%08" PRIx32 "\n", (uint32_t) (p>>32),
            (uint32_t) p);
13     return 0;
14   }
```

3. Provide a RV32I RISC-V Assembly Language program, assuming a RARS(M) environment, that corresponds to the program of Question 2 as closely as possible. (Note: *not RV32M*, so no `mul` and related instructions.) Briefly explain the key portions of the program.

4. Provide the machine code (text and data segments) corresponding to the programs of Question 3. Produce the code by hand first, then compare with the results using RARS(M).

5. Depict the states of the relevant words of the data segment and registers before and after the code of Question 4 is executed, as well as after half the iterations of the loop have been executed. Explain your answer.