IST-ASM Retake Exam — Fall 2024

Name:

- First, write your name in the box above. Then, have a quick read through all 7 questions.
- In the end, you will write up your answers on this paper.
  - But please make a draft elsewhere first. Only hand in something readable. Really.
- This is an open-book open-laptop exam: you may work on scrap paper and/or on your screen.
- Each question is independent from others, except stated otherwise.

Question 1 For each acronym below, give the full unabbreviated expression.

INSA	Institut National des Sciences Appliquées
CPU	
LR	
PC	
SP	

Central Processing Unit, Link Register, Program Counter, Stack Pointer

Question 2 Perform the addition 73 + 97 in binary notation: convert both numbers to binary, then compute the sum entirely in binary. Show the details of your work.

decimal 73 as binary:

decimal 97 as binary:

addition:

```
  1001001
+ 1100001
 1      1
---------
 10101010 = 128 + 32 + 8 + 2 = 170
```

Question 3 In the table below, encode your last name in ASCII (if some letters are missing, use the closest equivalent e.g. É↦E). Write each byte as a hexadecimal number (i.e. “42” will be read as 0x42, not “decimal 42”).

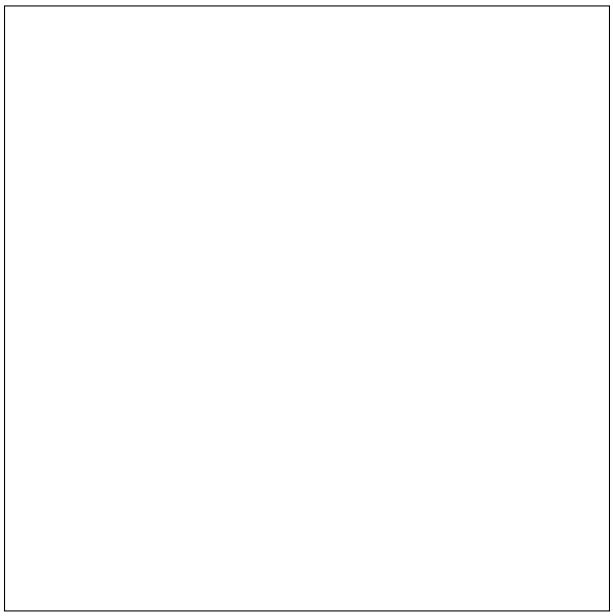
Letter												
ASCII (hex)												

SALAGNAC = 53 41 4C 41 47 4E 41 43
MOREL = 4D 4F 52 45 4C

Question 4 Give the ASM notation for instruction word `0x31abfff4`. Explain the meaning of each field separately, then a sentence to explain what the whole instruction does.

`bne R10, R11, -12` → jump three instructions back if R10 and R11 hold different values
3: instruction type = conditional jump
1: comp code = branch if not equal
A: Rd = R10
B: Rs = R11
FFF4: offset = -12

Question 5 Write a program which sets register R1 to value `0xdeadbeef`. You cannot use the `leti` instruction.



```
addi R1, R0, 0xdead ; gets sign extented
lsli R1, R1, 16
addi R2, R0, 0xbeef
lsli R2, R2, 16 ; shift left to eliminate ones
lsri R2, R2, 16 ; *logical* right shift brings back the beef but not the ones
add R1, R1, R2
halt
```

Question 6 Write a program which, given an array of known length, walks over all values and replaces negative numbers with zeros.

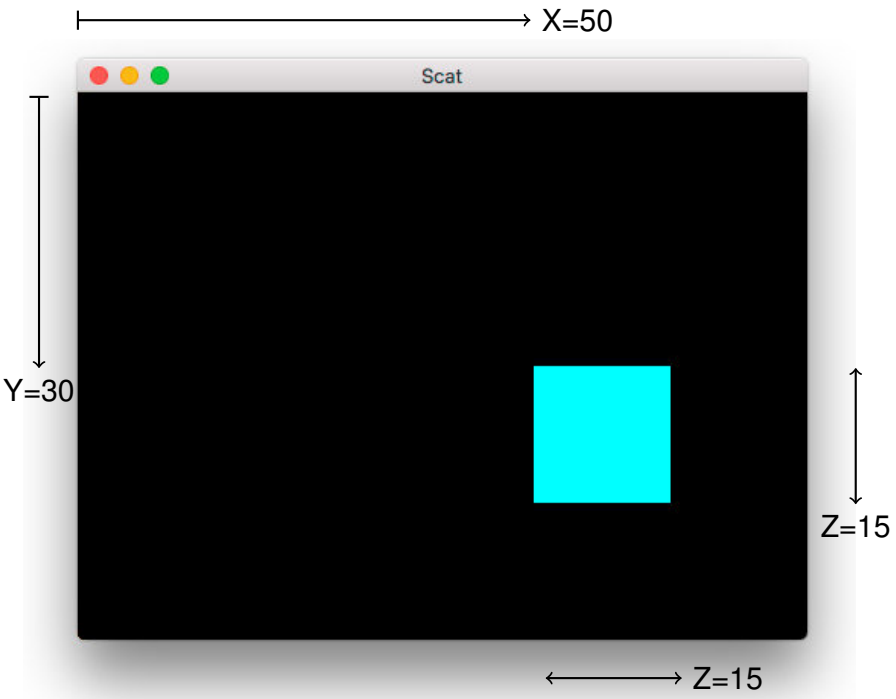
```
start:
   jmp main

data: .word 21, -25, 7, 34, 49, 31, 34, -4, -5, 23, 8, -38, -28, 19, 18
len:  .word 15

main:
```

XXX TODO

Question 7 Write a program which draws a cyan square with top-left corner at X, Y, and side length Z, like illustrated on the right. Parameters X, Y and Z are received in registers R1, R2 and R3, respectively, as illustrated below. Your program should work for any values of X, Y, and Z as long as they are meaningful. In other words, you may assume that X and Y are non-negative, that Z is strictly positive, that the square fits entirely on the screen, etc. You should not implement any error checking.




```
   leti R1, 50 ; X
   leti R2, 30 ; Y
   leti R3, 15 ; Z
main:
```