

Instructions. This exam has 7 questions, worth 10 points each. You have 50 minutes.

Resources. You may reference your optional two-sided 8.5-by-11 handwritten "cheat sheet" during this exam. You may not use the textbook, your notes, or any electronic devices. You may not communicate with anyone except the course staff during this exam.

After this exam. Due to travel for extracurriculars and sports, some of your peers will take this exam next week. Do not discuss its contents with anyone who has not taken it yet.

This paper. Do not remove this copy of the exam from the exam room. You may fill in this page now.

SOLUTION

NAME: _____

NETID: _____

PRECEPT: _____

EXAM ROOM: _____

"I pledge my honor that I will not violate the Honor Code during this examination."

SIGNATURE: _____

Fill in the blanks in the table below by converting the given values between number systems.

decimal	8-bit two's complement	hexadecimal
126	0111 1110	7E
-126	1000 0010	82
-1	1111 1111	FF
26	0001 1010	1A
-8	1111 1000	F8
-2	1111 1110	FE

Let $L = \{ bbaaa, abbaa, bbbaa, abaaa \}$. The alphabet for L is $\{a, b\}$.

For each of the following regular expressions, choose one of the following:

- NONE -- Matches no strings in L .
- SOME -- Matches some, but not all, strings in L .
- EXACT -- Matches all strings in L and no other strings.
- MORE -- Matches all strings in L and some other strings.

	NONE	SOME	EXACT	MORE
1.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
2. .(ab)*	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. (a* b*)(aa)*	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. ab*a*b*a*a*bab*	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. (ab bb)(baa aaa)	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
6. (a ab bbb)(bba abb a)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7. (a b)(ba bb)(a b)a	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
8. (bbb bba abb aba) ..	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
9. (b* ((ab) (ba)))*	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10. ((a b)* (aba* ..a))*	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

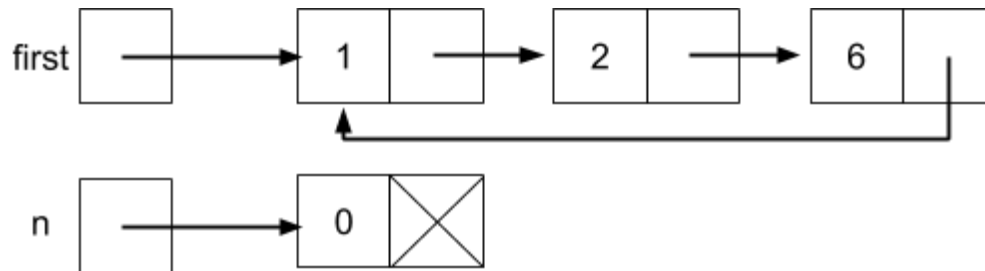
Fill in the blanks in the following TOY program that reads two inputs from `StdIn`, multiplies them using repeated addition, and prints the result to `StdOut`.

```
10: 8 A F F      reads from StdIn to R[A]
11: 8 B F F      reads from StdIn to R[B]
12: 7 C 0 0      R[C] <- 0000
13: 7 1 0 1      R[1] <- 0001
14: C A 18      HINT: if R[A] == 0, we have our answer!
15: 1 C BC      HINT: add something to R[C]
16: 2 A A1      HINT: subtract something from R[A]
17: C014      HINT: we're not done yet!
18: 9 C F F      prints R[C] to StdOut
19: 0 0 0 0      halt
```

For each cell in the table below, select the response that best represents our current understanding. Assume that, as we suspect, the Church-Turing thesis holds.

	solvable in polynomial time	reduces to SAT in polynomial time	SAT reduces to it in polynomial time
SORT	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input type="radio"/> true <input type="radio"/> false <input checked="" type="radio"/> unknown
FACTOR	<input type="radio"/> true <input type="radio"/> false <input checked="" type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input type="radio"/> true <input type="radio"/> false <input checked="" type="radio"/> unknown
PRIME	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input type="radio"/> true <input type="radio"/> false <input checked="" type="radio"/> unknown
TSP	<input type="radio"/> true <input type="radio"/> false <input checked="" type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown
SAT	<input type="radio"/> true <input type="radio"/> false <input checked="" type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown	<input checked="" type="radio"/> true <input type="radio"/> false <input type="radio"/> unknown
HALTING PROBLEM	<input type="radio"/> true <input checked="" type="radio"/> false <input type="radio"/> unknown	OUTSIDE THE SCOPE OF THIS COURSE	OUTSIDE THE SCOPE OF THIS COURSE

Choose the correct sequence of Java instructions to insert a node into a circularly linked list. You have access to two node variables, `first` and `n`, as diagrammed below. `n.next` is initialized to `null`. Your answers must produce a circularly linked list. Treat each of the four parts below as independent.



Choose one letter per box below, in the correct order, to perform the following operations. You may use each letter once, more than once, or not at all.

- | | |
|---|--|
| A. <code>first = n;</code> | F. <code>n.next = first;</code> |
| B. <code>first.next = n;</code> | G. <code>n.next = first.next;</code> |
| C. <code>first.next.next = n;</code> | H. <code>n.next = first.next.next;</code> |
| D. <code>first.next.next.next = n;</code> | I. <code>n.next = first.next.next.next;</code> |
| E. <code>first.next.next.next.next = n;</code> | J. <code>n.next = first.next.next.next.next;</code> |

1. Insert `n` between the first node and the second node.

G	B
---	---

2. Insert `n` after the last node.

F	D
---	---

3. Insert `n` between the second node and the third node.

H	C
---	---

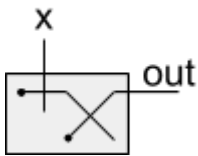
4. Insert `n` before the first node.

D	F	A
---	---	---

Choose the order of growth of each of the following operations.

	1	$\log N$	N	$N \log N$	N^2	N^3
1. Best-case scenario for insertion sort.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. Best-case scenario for mergesort.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Worst-case scenario for insertion sort.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
4. Worst-case scenario for mergesort.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. A program whose running time doubles (exactly) when the input size doubles.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6. A program that performs an operation on all possible pairings of N elements.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
7. Finding the largest element in a linked list.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8. Finding the median element in a sorted array.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9. Finding the average of all the elements in a binary search tree.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10. Finding the largest element in a balanced binary search tree.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

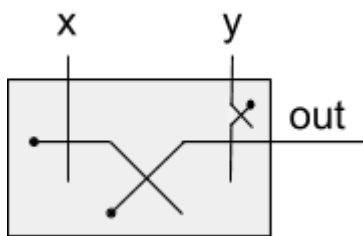
For each circuit below, complete the corresponding truth table and choose the best descriptor.



x	out
0	0
1	1

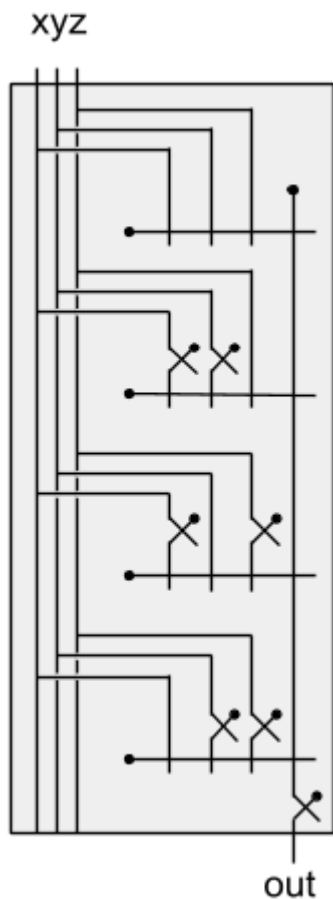
- not
- xor
- and
- maj
- odd
- even
- none of the above

Also accepted MAJ and ODD



x	y	out
0	0	0
0	1	0
1	0	0
1	1	1

- not
- xor
- and
- none of the above
- maj
- odd
- even



x	y	z	out
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

- not
- xor
- and
- none of the above
- maj
- odd
- even

TOY REFERENCE CARD

INSTRUCTION FORMATS

	
Format RR:	opcode d s t	(0-6, A-B)
Format A:	opcode d addr	(7-9, C-F)

ARITHMETIC and LOGICAL operations

1: add	$R[d] \leftarrow R[s] + R[t]$
2: subtract	$R[d] \leftarrow R[s] - R[t]$
3: and	$R[d] \leftarrow R[s] \& R[t]$
4: xor	$R[d] \leftarrow R[s] \wedge R[t]$
5: shift left	$R[d] \leftarrow R[s] \ll R[t]$
6: shift right	$R[d] \leftarrow R[s] \gg R[t]$

TRANSFER between registers and memory

7: load address	$R[d] \leftarrow \text{addr}$
8: load	$R[d] \leftarrow M[\text{addr}]$
9: store	$M[\text{addr}] \leftarrow R[d]$
A: load indirect	$R[d] \leftarrow M[R[t]]$
B: store indirect	$M[R[t]] \leftarrow R[d]$

CONTROL

0: halt	halt
C: branch zero	if $(R[d] == 0)$ PC \leftarrow addr
D: branch positive	if $(R[d] > 0)$ PC \leftarrow addr
E: jump register	PC \leftarrow R[d]
F: jump and link	$R[d] \leftarrow$ PC; PC \leftarrow addr

Register 0 always reads 0.

Loads from M[FF] come from stdin.

Stores to M[FF] go to stdout.

16-bit registers (two's complement)

16-bit memory locations

8-bit program counter

You may tear this page out and use it as scratch paper.

If you do, fill in the blanks below and return it *inside* your exam.

NAME: _____

NETID: _____

PRECEPT: _____

EXAM ROOM: _____