Thomas Jefferson
Invitational Open in Informatics 2012

Exam in Java

Contest Part I (Theoretical—Short Answer)

Answer Document

Team # ____

Do not open until told to do so.

Rules and Instructions:

1. You will have 60 minutes to complete this section. This may not be enough time to complete the entire contest. Do not be alarmed if you do not finish every problem.
2. For each problem, write your answer clearly, neatly, and legibly in the space provided in the answer document.
3. For code analysis problems, assume that all necessary headers and libraries have been included, and that there are no compile or run-time errors.
4. Each problem will be weighted differently, for a total of 50 points. Partial credit may be awarded for partially-correct answers.
5. Work on these problems within your team. Place all of your final answers in this answer document.
6. Please direct all non-content-related questions to the proctors.
7. Good luck, and have fun!

Short Answer Problem 0 __/2

Replace the line marked by an asterisk with:

A) `return pascal(row - 1, col) + pascal(row + 1, col);`

B) `return pascal(row - 1, col) * pascal(row - 1, col + 1);`

C) `return pascal(row - 1, col - 1) + pascal(row - 1, col);`

D) `return pascal(row + 1, col - 1) + pascal(row + 1, col);`

E) `return pascal(row - 1, col - 1) + pascal(row - 1, col + 1);` (no change)

(2.0 pts) Answer: **C**

Short Answer Problem 1 __/3

Output: $\underline{7, 5, 5, 4, 4}$
$\geq \frac{3}{5}$ correct $\rightarrow$ (1.0 pts)
All correct $\rightarrow$ (3.0 pts)

Short Answer Problem 2 __/3

	Alex (A)	Brian (B)	Chloe (C)	Doug (D)
Coming?	**YES**	**NO**	**YES**	**YES**

(0.5 pts) for each correct
(1.0 pts) bonus for all correct

Short Answer Problem 3 __/4

a) Output: (2.0 pts) 120

b) Output: (2.0 pts) -4

Short Answer Problem 4 __/3

Replace the line marked by the asterisk with:

A) `x = a * (a - c);`

B) `a = x * (x - c);`

C) `x = x * (a - c);`

D) `x = x * (x - a);`

E) `a = a * (a - c);` **(no change)**

(3.0 pts) Answer: C

Short Answer Problem 5 __/4

Find when the following expressions evaluate to TRUE. Express your answers as ordered tuples.

a) **(1.0 pts)** (0,1,1), (1,0,0), (1,0,1), (1,1,0), (1,1,1)

b) **(1.0 pts)** (0,0,0), (0,0,1), (0,1,0), (1,0,0), (1,0,1), (1,1,0), (1,1,1)

c) **(2.0 pts)** (1,0,1), (1,1,0), (1,1,1)

Short Answer Problem 6 __/4

a) **(0.5 pts)** 1000000_2

b) **(0.5 pts)** 10011111100000_2

c) **(0.5 pts)** $2A8152E1A_{16}$

d) **(0.5 pts)** 101110_2

e) **(0.5 pts)** 15477_8

f) **(0.5 pts)** 101010_2

g) **(1.0 pts)** 1000000000000000_{16}

Short Answer Problem 7 __/4

a) **Overall winner: (2.0 pts)** Rhonum

b) **Player with Highest Total Score: (1.0 pts)** Pickle
Highest Total Score: (1.0 pts) 28 or 19 (problem statement not clear)

Short Answer Problem 8 __/4

function(12)= (0.5 pts) 17

function(16)= (0.5 pts) 18

function(39)= (0.5 pts) 55

function(99)= (1.0 pts) 113

function(100000000)= (1.5 pts) 100000007

Short Answer Problem 9 __/4

Order:

1. **A**
2. **B**
3. **H**
4. **G**
5. **I**
6. **C**
7. **K**
8. **D**
9. **J**
10. **n/a or blank**
11. **n/a or blank**

For each column: One wrong → (0.5 pts)
All 3 → (1.0 pts)

Short Answer Problem 10 __/5

a)

Prefix: `+ + + 1 3 2 8`
Infix: `1 + 3 + 2 + 8`
Postfix: `1 3 + 2 + 8 +`

b)

Prefix: `/ + 1 3 + 2 8`
Infix: `(1 + 3)/(2 + 8)`
Postfix: `1 3 + 2 8 + /`

c)

Prefix: `+ - - 1 4 6 7`
Infix: `1 - 4 - 6 + 7`
Postfix: `1 4 - 6 - 7 +`

d)

Prefix: `↑ + 1 - 4 9 * 2 3`
Infix: `(1 + 4 - 9) ↑ 2 * 3`
Postfix: `1 4 9 - + 2 3 * ↑`

e)

Prefix: `+ + 1 / 3 2 8`
Infix: `1 + 3 / 2 + 8`
Postfix: `1 3 2 / + 8 +`

f)

Prefix: `- + * ↑ 1 2 3 4 5`
Infix: `1 ↑ 2 * 3 + 4 - 5`
Postfix: `1 2 ↑ 3 * 4 + 5 -`

g)

Prefix: `/ + * 1 3 5 7`
Infix: `(1 * 3 + 5) / 7`
Postfix: `1 3 * 5 + 7 /`

h)

Prefix: `↑ * 1 3 - 5 7`
Infix: `(1 * 3) ↑ (5 - 7)`
Postfix: `1 3 * 5 7 - ↑`

i)

Prefix: `+ - 1 2 * 3 ↑ 4 5`
Infix: `1 - 2 + 3 * 4 ↑ 5`
Postfix: `1 2 - 3 4 5 ↑ * +`

j)

Prefix: `- 1 / ↑ 4 3 5`
Infix: `1 - 4 ↑ 3 / 5`
Postfix: `1 4 3 ↑ 5 / -`

k)

Prefix: `/ * ↑ 1 * 2 3 4 5`
Infix: `(1 ↑ (2 * 3)) * 4 / 5`
Postfix: `1 2 3 * ↑ 4 * 5 /`

l)

Prefix: `↑ 3 * / 4 7 5`
Infix: `3 ↑ (4 / 7 * 5)`
Postfix: `3 4 7 / 5 * ↑`

Each correct is worth $\frac{1}{8}$ pts. Round the total to the nearest (0.5 pts).

m) (0.5 pts) Circle one: Prefix Infix **Postfix**
Evaluate: (0.5 pts) $\frac{3}{2}$ **or** 1.5

n) (0.5 pts) Circle one: **Prefix** **Infix** **Postfix**
Evaluate: (0.5 pts) $\frac{1}{40}$ **or** 0.025

Short Answer Problem 11 __/5

a) **Shortest Path Length: (0.5 pts)** 2

b) **Shortest Path Length: (0.5 pts)** 4

c) **Shortest Path Length: (1.0 pts)** 7

d) **Shortest Path Length: (1.0 pts)** 2

e) **Shortest Path Length: (2.0 pts)** 12

Short Answer Problem 12 __/5

a) **(1.0 pts) Only works for $b = 2^k$ where k is a non-negative integer, and $b = 0$, because of integer division. (Or something similar).**

b) **(1.0 pts)** $(a, b > 1000) = (*, 0$ **or** $2^k)$**.** a **can be anything, but** b **must either be** 0 **or a power of** 2**.**

c)
```
public static int pow(int a, int b){
    if (b <= 0){
        return 1;
    }
    else if (b == 1){
        return a;
    }
    else{
        int tmp = pow(a, b / 2);
        return tmp * tmp;
    }
}
```

Answer: (2.0 pts)
Change `return tmp * tmp;` **with:**

```
if(b % 2 == 0)
    return tmp * tmp;
else
    return tmp * tmp * a;
```

d) **Runtime Complexity: (1.0 pts)** $O(\log_2 b)$ **or** $O(\log b)$ **or** $O(\log_2 N)$ **or** $O(\log N)$
Or correct for above implementation, if different from given solution.