**International University of Business Agriculture and Technology**

**Mid Term Assignment**

**Course Code: CSC-461**

**Submitted To:**

**Aminun Nahar**

**Lecturer**

**Department of CSE, IUBAT**

**Submitted By:**

**Md. Shabiul Alam**

**ID: 18103326**

**Program: BCSE**

**Section: G**

**Date of Submission: April 17, 2021**

Ans: to the Ques: NO: 01

(a)

```
if (a == b)
    b = 2 * a + 1
else
    b = 2 * a
    { b > 1 }
```

$2 * a + 1 > 1 \rightarrow 2 * a > 0 \rightarrow a > 0$

$2 * a > 1 \rightarrow a > \frac{1}{2}$

The weakest precondition is $\{ a > 1/2 \}$

(b)

```
if (x > y)
    x = x + 1
else
    x = 3 * x
```

$x + 1 < 0$

$\therefore x < -1$

$3 * x < 0$

$\therefore x < 0$

The weakest precondition is $\{x < 0\}$

Ans: to the Ques: NO: 02

String: C = B * (C * (A+B))

Grammar:

<assign> → <id> = <expr>

<id> → A | B | C

<expr> → <expr> + <term> | <term>

<term> → <term> * <factor> | <factor>

<factor> → (<expr>) | <id>

| Stack | Input string | Action |
|---|---|---|
| $ | C = B * (C * (A+B)) $ | |
| $C | = B * (C * (A+B)) $ | Shift C |
| $ <id> | = B * (C * (A+B)) $ | <id> → C |

| Stack | Input String | Action |
|---|---|---|
| $ <factor> | = B * (C * (A+B)) $ | <factor> → <id> |
| $ <term> | = B * (C * (A+B))$ | <term> → <factor> |
| $ <expr> | = B * (C * (A+B))$ | <expr> → <term> |
| $ <id> | = B * (C * (A+B))$ | <id> ⇒ <expr> |
| $ <id> | = B * (C * (A+B))$ | shift = |
| $ <id> = B | B * (C * (A+B))$ | shift B |
| $ <id> = <id> | * (C * (A+B))$ | <id> → B |
| $ <id> = <factor> | * (C * (A+B))$ | <factor> → <id> |
| $ <id> = <term> | * (C * (A+B))$ | <term> → <factor> |
| $ <id> = <expr> | * (C * (A+B))$ | <expr> → <term> |
| $ <id> = <expr> * | (C * (A+B))$ | shift * |
| $ <id> = <expr> * ( | C * (A+B))$ | shift ( |
| $ <id> = <expr> * (C | * (A+B))$ | shift C |
| $ <id> = <expr> * (<id> | * (A+B))$ | <id> → C |

| Stack | Input string | Action |
|---|---|---|
| $ <id> = <expr> * (<factor> | * (A+B))$ | <factor> → <id> |
| $ <id> = <expr> * (<term> | * (A+B))$ | <term> → <factor> |
| $ <id> = <expr> * (<expr> | * (A+B))$ | <expr> → <factor> |
| $ <id> = <expr> * (<expr> * | (A+B))$ | ~~<expr> →~~ Swift * |
| $ <id> = <expr> * (<expr> * ( | A+B)$ | Swift ( |
| $ <id> = <expr> * (<expr> * (A | +B)$ | Swift A |
| $ <id> = <expr> * (<expr> * (<id> | +B)$ | <id> → A |
| $ <id> = <expr> * (* <expr> * (<factor> | +B)$ | <factor> → <id> |
| $ <id> = <expr> * (<expr> * ( | +B) $ | <term> → <factor> |
| $ <id> = <expr> * (<expr> * (<expr> | +B)$ | <expr> → <term> |
| $ <id> = <expr> * (<expr> * (<expr> + | B)$ | Swift + |
| $ <id> = <expr> * (<expr> * (<expr> + B | )$ | Shift B |
| $ <id> = <expr> * (<expr> * (<expr> + <id> | )$ | <id> → B |

| Stack | Input String | Action |
|---|---|---|
| $ <id>= <expr> *(<expr>* (<expr>+ <factor> | )$ | <term> → <id> |
| $ <id>= <expr>* (<expr>* (<expr>+ <term> | )$ | <term> → <factor> |
| $ <id> = <expr>* (<expr> *(<expr>+ <expr> | )$ | <expr> → <term> |
| $ <id> = <expr>* (<expr> *(<expr>+<expr>) | $ | shift )<br>Accept |

## Ans: to the Ques: NO: 03

### (a)

i) sub 1  x ~~decl~~ declared from sub 1 will be referenced

ii) sub 2  x declared from Sub 1 will be. referenced

iii) sub 3  x declared from. main will be. referenced

### (b)

i) Sub 1  x declared from sub 1 will be referenced

ii) sub 2  x declared from Sub 1 will be referenced

iii) Sub 3  x declared from Sub 1 will be referenced.

## Ans: to the Ques: NO: 04

Output: 20

Explanation: In this program the int main() function & in it the printf() function the logic is $c = x + g(x)$. x is a global variable & the value is 10.

So, $c = 10 + g()$

Now, in the g(x) function x has been declared as 20. But It has a return function f(). And f() function returns to global variables x. So the value of g() is 10.

So, $c = 10 + 10 = 20$