CS101 Mid-Sem Practice

Autumn 2024-25

Abhijat Bharadwaj, Devavrat Patni and Shubham Raj

Question 1: Rightly exponential

Identify the problems in the following code.

```
int calculate-exp(double x){
    double sum = term = 1;
    for (int i = 1; i < 100; ++i) {
        term *= x / i;
        sum += term;
    }
    return term;
}
```

Solution 1: Rightly exponential

1. Function's name contains illegal character "-"
2. Function's type should be double
3. Variable term is used before initialization

```
double calculate_exp  (double x){
    double sum = 1, term = 1;
    for (int i = 1; i < 100; ++i) {
        term *= x / i;
        sum += term;
    }
    return sum;
}
```

Alternative Solution:

```
double term;
double sum = term = 1;
```

functions

for loops

Question 2: Floating precision

What is the output of the following program?

a) 0 0
b) 1 0
c) 1 1
d) 0 1

```
#include<simplecpp>

main_program{
    float a = 0.1;
    float b = 0.2;
    float c = a+b;

    cout << (c == 0.3)<< " "
         <<(c == 0.3F);
}
```

Solution 2: Floating precision

What is the output of the following program?

a) 0 0
b) 1 0
c) 1 1
d) 0 1

```
#include<simplecpp>

main_program{
    float a = 0.1;
    float b = 0.2;
    float c = a+b;

    cout << (c == 0.3)<< " "
         <<(c == 0.3F);
}
```

Floating point precision error

(c == 0.3) is false, which typecasts to 0

By default 0.3 is double. F forces it to float

floats

doubles

Question 3: Pascal’s triangle

Complete the following code to print ‘n’ rows of Pascal’s triangle.

For n = 6

```
1.                 1
2.               1   1
3.             1   2   1
4.           1   3   3   1
5.         1   4   6   4   1
6.       1   5   10   10   5   1
```

```
void pascalTriangle(int n){
    int c = 1;
    for(int i = 0; i < n; i++) {
        for(int s = 1; s <= n-i; s++)
            __(i)__;
        for(int j = 0; j <= i; j++) {
            if (__(ii)__)c = 1;
            else c = c *(i-j+1)/j;
            cout << c << "   ";
        }
       __(iii)__;
    }
}
```

Solution 3: Pascal's triangle

Divide pattern printing problem into three subproblems:
- calculating starting whitespace
- calculating individual element value
- calculating internal whitespace

Solve them using nested loops!

```
void pascalTriangle(int n){
    int c = 1;
    for(int i = 0; i < n; i++) {
        for(int s = 1; s <= n-i; s++)
            cout << "  ";
        for(int j = 0; j <= i; j++) {
            if (i == 0 || j == 0)c = 1;
            else c = c *(i-j+1)/j;
            cout << c << "   ";
        }
        cout << endl;
    }
}
```

|| represents logical OR
&& represents logical AND

Switch to new line
Alternatively: `cout << '\n'`

loops

conditions

There is one redundant statement in this code, can you find it?

Question 4: Moving the Base

Guess the output again

1

```
int main(){
    cout << myFunc(200,4)
    <<" "<< myFunc(10,2)
    <<" "<< myFunc(30,5);
    return 0;
}
```

2

```
int myFunc(int n, int b) {
    if (n == 0) return 0;
    int m = 0, pv = 1, num = n;
    while (num > 0) {
        int r = num % b;
        m += r * pv;
        pv*= 10;
        num /= b;
    }
    return m;
}
```

Solution 4: Moving the Base

Output: 3020 1010 110

This function converts any integer ‘n’ from decimal (base10) to an integer in base ‘b’.

$3020_4 = 200_{10}$
$1010_2 = 10_{10}$
$110_5 = 30_{10}$

loops

functions

```
int myFunc(int n, int b) {
    if (n == 0) return 0;
    int m = 0, pv = 1, num = n;
    while (num > 0) {
        int r = num % b;
        m += r * pv;
        pv*= 10;
        num /= b;
    }
    return m;
}
```

Question 5: Operation Theatre

Which of the following, when replacing the blank in the code given, will result in 18?

a) a + b / c + d
b) d + a | (c / b)
c) a | (c / b) + d
d) a + float(b) / c + d

```
int a = 10;
int b = 3;
int c = 7;
int d = 8;

int k = __(i)__;
cout<<k;
```

Solution 5: Operation Theatre

Which of the following, when replacing the blank in the code given, will result in 18?

a) a + b / c + d
b) d + a | (c / b)
c) a | (c / b) + d
d) a + float(b) / c + d

c) will result in 10

```
int a = 10;
int b = 3;
int c = 7;
int d = 8;

int k = __(i)__;
cout<<k;
```

Arithmetic operators have higher priority over bitwise operators

operators

Operator Precedence in C++ (Highest to Lowest)

++ – –	Increment/Decrement	Arithmetic
! ~ – +	Logical/Bitwise NOT, Unary plus/minus	
* / %	Multiplication, Division, Modulus	
+ –	Addition, Subtraction	
<< >>	Bitwise shift	Bitwise Shift
< <= > >=	Relational operators	Relations
== !=	Equality operators	
&	Bitwise AND	Bitwise Operators
^	Bitwise XOR	
|	Bitwise OR	
&&	Logical AND	Logical
||	Logical OR	
?:	Ternary conditional	
= += -= *= /= %= &= ^= |= <<= >>=	Assignment and compound assignment	Assignment

Question 6: Subtle Reference

Guess the output of the program.

a) x = 10
   ref = 20
b) x = 30
   ref = 20
c) x = 20
   ref = 30
d) x = 10
   ref = 30

```
#include<simplecpp>

main_program{
  int x = 10;
  int& ref = x;
  ref = 20;
  cout << "x = " << x << endl ;
  x = 30;
  cout << "ref = " << ref << endl;
}
```

Solution 6: Subtle Reference

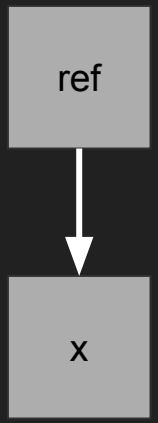
Guess the output of the program.

a) x = 10
   ref = 20

b) x = 30
   ref = 20

c) x = 20
   ref = 30

d) x = 10
   ref = 30




```
#include<simplecpp>

main_program{
  int x = 10;
  int& ref = x;
  ref = 20;
  cout << "x = " << x << endl ;
  x = 30;
  cout << "ref = " << ref << endl;
}
```

references

Question 7: See Plus Plus

```
#include<simplecpp>
main_program{
   int a = 5;
   cout << a++ ;
}
```

```
#include<simplecpp>
main_program{
   int a = 5;
   cout << ++a ;
}
```

```
#include<simplecpp>
main_program{
   int a = 5;
   a++;
   cout << a;
}
```

```
#include<simplecpp>
main_program{
   int a = 5;
   ++a;
   cout << a;
}
```

Predict the output?

a)	5	6	6	6
b)	6	5	5	5
c)	5	6	5	6
d)	6	6	6	6

Answer 7

#include<simplecpp>
main_program{
int a = 5;
cout << a++ ;
}
	#include<simplecpp>
	main_program{
	int a = 5;
	cout << ++a ;
	}
		#include<simplecpp>
		main_program{
		int a = 5;
		a++;
		cout << a;
		}
			#include<simplecpp>
			main_program{
			int a = 5;
			++a;
			cout << a;
			}

Predict the output?

a)	5	6	6	6
b)	6	5	5	5
c)	5	6	5	6
d)	6	6	6	6

Solution 7

```
#include<simplecpp>
main_program{
   int a = 5;
   cout << a++ ;
}
```

cout will get a = 5 and then ‘a’ will increase

```
#include<simplecpp>
main_program{
   int a = 5;
   cout << ++a ;
}
```

‘a’ will increase to 6 and then cout will get value of ‘a’

```
#include<simplecpp>
main_program{
   int a = 5;
   a++;
   cout << a;
}
```

‘a’ increases to 6

end of statement

cout gets value of ‘a’

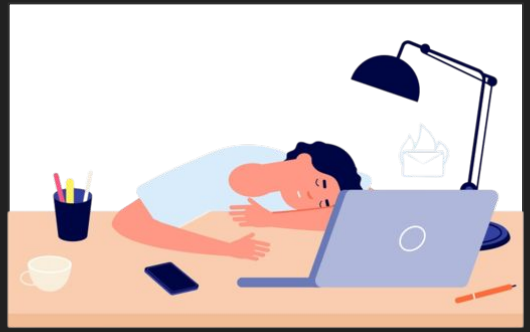
```
#include<simplecpp>
main_program{
   int a = 5;
   ++a;
   cout << a;
}
```

‘a’ increases to 6

end of statement

cout gets value of ‘a’

Increment operators



Doubt break!

Question 8: See Minus Minus too

```
#include<simplecpp>
main_program {
    int a = 5, b = 10, c = 15;
    int result = (a++) + (++b) - (--c) + (b--) + (c++) - (--a) + (++b) - (c--);
    cout << "a = " << a << endl;
    cout << "b = " << b << endl;
    cout << "c = " << c << endl;
    cout << "result = " << result << endl;
}
```

Predict the output?

a = 5 b = 10 c = 14 result = 17	a = 5 b = 10 c = 13 result = 17	a = 5 b = 11 c = 14 result = 18	a = 4 b = 11 c = 13 result = 19

Answer 8

```
#include<simplecpp>
main_program {
    int a = 5, b = 10, c = 15;
    int result = (a++) + (++b) - (--c) + (b--) + (c++) - (--a) + (++b) - (c--);
    cout << "a = " << a << endl;
    cout << "b = " << b << endl;
    cout << "c = " << c << endl;
    cout << "result = " << result << endl;
}
```

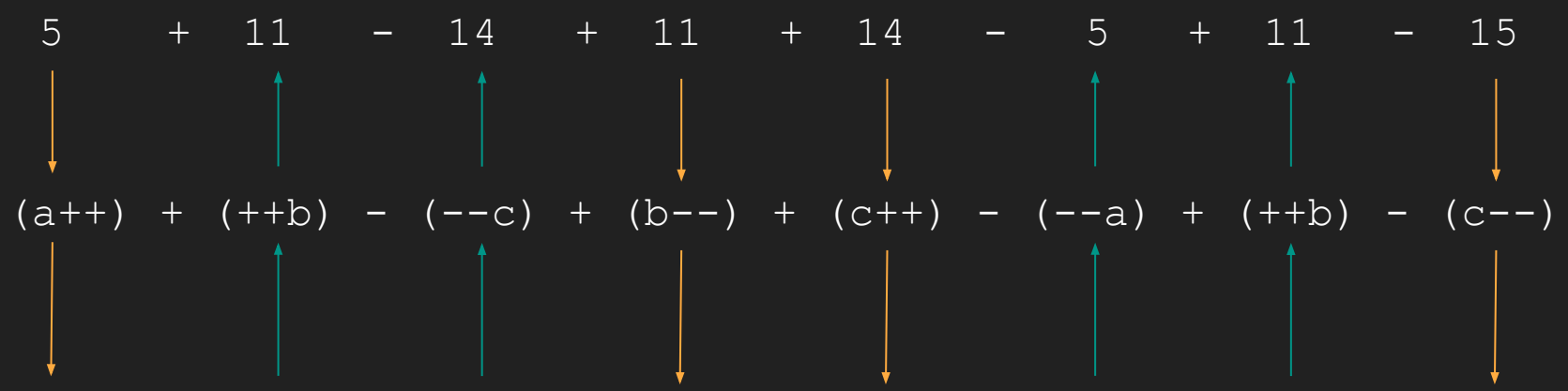
Predict the output?

a = 5 b = 10 c = 14 result = 17	a = 5 b = 10 c = 13 result = 17	a = 5 b = 11 c = 14 result = 18	a = 4 b = 11 c = 13 result = 19

a = 5
b = 10
c = 15

Pre Increment → Increase First → Then Use

Post Increment → Use Value → Then Increment



a = 6 b = 10 c = 15	a = 6 b = 11 c = 15	a = 6 b = 11 c = 14	a = 6 b = 10 c = 14	a = 6 b = 10 c = 15	a = 5 b = 10 c = 15	a = 5 b = 11 c = 15	a = 5 b = 11 c = 14

Question 9: Reference Riddle - I

```
#include <simplecpp>

void modify(int &x, int y) {
    x += y;
    y = x - y;
    x = x - y;
}

void calculate(int a, int &b, int &c) {
    a += 5;
    b = a * 2;
    modify(b, c);
    c += a;
}

main_program {
    int p = 3, q = 4, r = 5;

    calculate(p, q, r);

    cout << "p = " << p << endl;
    cout << "q = " << q << endl;
    cout << "r = " << r << endl;
}
```

Predict the output

a)	p = 3 q = 17 r = 13
b)	p = 3 q = 5 r = 13
c)	p = 8 q = 17 r = 13
d)	p = 8 q = 5 r = 13

Answer

```
#include <simplecpp>

void modify(int &x, int y) {
    x += y;
    y = x - y;
    x = x - y;
}

void calculate(int a, int &b, int &c) {
    a += 5;
    b = a * 2;
    modify(b, c);
    c += a;
}

main_program {
    int p = 3, q = 4, r = 5;

    calculate(p, q, r);

    cout << "p = " << p << endl;
    cout << "q = " << q << endl;
    cout << "r = " << r << endl;
}
```

Predict the output

a)	p = 3 q = 17 r = 13
b)	p = 3 q = 5 r = 13
c)	p = 8 q = 17 r = 13
d)	p = 8 q = 5 r = 13

Solution

```
#include <simplecpp>

void modify(int &x, int y) {
    x += y;
    y = x - y;
    x = x - y;
}

void calculate(int a, int &b, int &c) {
    a += 5;
    b = a * 2;
    modify(b, c);
    c += a;
}

main_program {
    int p = 3, q = 4, r = 5;

    calculate(p, q, r);

    cout << "p = " << p << endl;
    cout << "q = " << q << endl;
    cout << "r = " << r << endl;
}
```

x = 16 + 5 = 21

y = 21 - 5 = 16

x = 21 - 15 = 5

a = 3 + 5 = 8

b = 8 * 2 = 16

c = 5 + 8 = 13

x = 16 y = 5

a = 3 b = 4 c = 5

b = 16 (passed by ref, value will change)
c = 5 (passed by value, value will not change)

p = 3 (passed by value, value will not change)
q = 5 (passed by ref, value will change)
r = 13 (passed by ref, value will change)

Question 10: Reference Riddle - II

```
#include <simplecpp>

int bitwiseOperations(int x, int y) {
    y = y >> 5;

    int a = x & y;

    int b = x ^ y;

    x <<= 2;

    int c = y | (x << 1);

    return (a + b + c + x + y);
}

main_program {
    int x = 6;
    char y = 'a';
    int result = bitwiseOperations(x, y);
    cout << "Result: " << result << endl;
    return 0;
}
```

Predict the output

a)	Result: 179
b)	Result: 84
c)	Result: 154
d)	Result: 85

Answer

```
#include <simplecpp>

int bitwiseOperations(int x, int y) {
    y = y >> 5;

    int a = x & y;

    int b = x ^ y;

    x <<= 2;

    int c = y | (x << 1);

    return (a + b + c + x + y);
}

main_program {
    int x = 6;
    char y = 'a';
    int result = bitwiseOperations(x, y);
    cout << "Result: " << result << endl;
    return 0;
}
```

Predict the output

a)	Result: 179
b)	Result: 84
c)	Result: 154
d)	Result: 85

Solution

```
#include <simplecpp>

int bitwiseOperations(int x, int y) {

    y = y >> 5;

    int a = x & y;

    int b = x ^ y;

    x <<= 2;

    int c = y | (x << 1);

    return (a + b + c + x + y);
}

main_program {
    int x = 6;
    char y = 'a';
    int result = bitwiseOperations(x, y);
    cout << "Result: " << result << endl;
    return 0;
}
```

x = 6
y = 97

y = (97 >> 5)
0110 0001
_____
0000 0011
y = 3

a = (6 & 3)
0000 0110
0000 0011
_____
00000010
a = 2

b = (6 ^ 3)
0000 0110
0000 0011
_____
0000 0101
b = 5

x = x << 2
x = (6 << 2)
0000 0110
_____
0001 1000
x = 24

c = (3 | (6 << 1))
0000 0011
0011 0000
_____
0011 0011
c = 51

Q11 Overload… and more references

```
void a2b(long &a, long &b){
    a += b;
    a += b;
    cout << "reference " << a << endl;
}
void a2b(int a, int b){
    a += b;
    a += b;
    cout << "value " << a << endl;
}

int main(){
    int x = 3, y = 9;
    long a = 5, b = 10, c = 20;
    a2b(a, b);
    a2b(x, y);
    a2b(c, c);
    return 0;
}
```

Predict the output

Q11 Overload… and more references

```
void a2b(long &a, long &b){
    a += b;
    a += b;
    cout << "reference " << a << endl;
}
void a2b(int a, int b){
    a += b;
    a += b;
    cout << "value " << a << endl;
}

int main(){
    int x = 3, y = 9;
    long a = 5, b = 10, c = 20;
    a2b(a, b);
    a2b(x, y);
    a2b(c, c);
    return 0;
}
```

Predict the output

reference 25
value 21
reference 80

Q11 Overload… and more references

```
void a2b(long &a, long &b){
    a += b;
    a += b;
    cout << "reference " << a << endl;
}
void a2b(int a, int b){
    a += b;
    a += b;
    cout << "value " << a << endl;
}

int main(){
    int x = 3, y = 5;
    long a = 5, b = 10, c = 20;
    a2b(a, b);
    a2b(x, y);
    a2b(c, c);
    return 0;
}
```

a=25, b=10 (pass by ref)

Predict the output

reference 25
value 21
reference 80

Q11 Overload… and more references

```
void a2b(long &a, long &b){
    a += b;
    a += b;
    cout << "reference " << a << endl;
}
void a2b(int a, int b){
    a += b;
    a += b;
    cout << "value " << a << endl;
}

int main(){
    int x = 3, y = 9;
    long a = 5, b = 10, c = 20;
    a2b(a, b);
    a2b(x, y);
    a2b(c, c);
    return 0;
}
```

a=25, b=10 (pass by ref)

x=3, y=9

Predict the output

reference 25
value 21
reference 80

Q11 Overload… and more references

```
void a2b(long &a, long &b){
    a += b;
    a += b;
    cout << "reference " << a << endl;
}
void a2b(int a, int b){
    a += b;
    a += b;
    cout << "value " << a << endl;
}

int main(){
    int x = 3, y = 9;
    long a = 5, b = 10, c = 20;
    a2b(a, b);
    a2b(x, y);
    a2b(c, c);
    return 0;
}
```

c updated!!

a=25, b=10 (pass by ref)

x=3, y=9

Predict the output

reference 25
value 21
reference 80

Q12 Structs and functions

```
struct Vector3D {
double x; double y; double z;
};

_________ crossProduct (_________, _________) {
Vector3D result;

_________
return result;
}

int main() {
Vector3D vector1 = {1.0, 2.0, 3.0};
Vector3D vector2 = {4.0, 5.0, 6.0};
Vector3D result = crossProduct(vector1, vector2);
}
```

Complete the function

Q12 Structs and functions

```
struct Vector3D {
double x; double y; double z;
};

Vector3D crossProduct (_____, _____) {
Vector3D result;

_____
return result;
}

int main() {
Vector3D vector1 = {1.0, 2.0, 3.0};
Vector3D vector2 = {4.0, 5.0, 6.0};
Vector3D result = crossProduct(vector1, vector2);
}
```

Complete the function

Q12 Structs and functions

```
struct Vector3D {
double x; double y; double z;
};

Vector3D crossProduct (Vector3D v1, Vector3D v2) {
Vector3D result;

_____
return result;
}

int main() {
Vector3D vector1 = {1.0, 2.0, 3.0};
Vector3D vector2 = {4.0, 5.0, 6.0};
Vector3D result = crossProduct(vector1, vector2);
}
```

Complete the function

Q12 Structs and functions

```
struct Vector3D {
double x; double y; double z;
};

Vector3D crossProduct (Vector3D v1, Vector3D v2) {
Vector3D result;
result.x = v1.y * v2.z - v1.z * v2.y;
result.y = v1.z * v2.x - v1.x * v2.z;
result.z = v1.x * v2.y - v1.y * v2.x;
return result;
}

int main() {
Vector3D vector1 = {1.0, 2.0, 3.0};
Vector3D vector2 = {4.0, 5.0, 6.0};
Vector3D result = crossProduct(vector1, vector2);
}
```

Complete the function