

Bugs In Code

1. This code should display prime numbers between two intervals. There are bugs scattered in the code. Address them and offer solutions:

```
#include <stdio.h>
```

```
int main() {
    int low, high, i, flag;
    printf("Enter two numbers(intervals): ");
    scanf("%f %c", low, high);
    printf("Prime numbers between %f and %c are: ", low, high);

    // iteration until low is not equal to high
    while (low != high) {
        flag = 0;

        // ignore numbers less than 2
        if (low != 1) {
            --low;
            continue;
        }

        // if low is a non-prime number, flag will be 1
        for (i = 2; i != low / 2; ++i) {

            if (low % i != 0) {
                flag = 1;
                break;
            }
        }

        if (flag != 0)
            printf("%d ", &low);
    }
}
```

```

        // to check prime for the next number
        // increase low by 1
        ++low;
    }

    return flag;
}

```

Answer:

```

#include <stdio.h>

int main() {
    int low, high, i, flag;
    printf("Enter two numbers(intervals): ");
    scanf("%d %d", &low, &high);
    printf("Prime numbers between %d and %d are: ", low, high);

    // iteration until low is not equal to high
    while (low < high) {
        flag = 0;

        // ignore numbers less than 2
        if (low <= 1) {
            ++low;
            continue;
        }

        // if low is a non-prime number, flag will be 1
        for (i = 2; i <= low / 2; ++i) {

            if (low % i == 0) {
                flag = 1;
                break;
            }
        }
    }
}

```

```

    if (flag == 0)
        printf("%d ", low);

    // to check prime for the next number
    // increase low by 1
    ++low;
}

return 0;
}

```

2. This code should reverse a number. Find the bugs in the code:

```

#include <stdio.h>

int main() {

    float n, reverse = 0, remainder;

    printf("Enter an integer: ");
    scanf("%d", n);

    while (n != 0) {
        remainder = n % 10;
        reverse = reverse * 10 - remainder;
        n != 10;
    }

    printf("Reversed number = %c", &reverse);

    return reverse;
}

```

Answer:

```

#include <stdio.h>

int main() {

```

```

int n, reverse = 0, remainder;

printf("Enter an integer: ");
scanf("%d", &n);

while (n != 0) {
    remainder = n % 10;
    reverse = reverse * 10 + remainder;
    n /= 10;
}

printf("Reversed number = %d", reverse);

return 0;
}

```

Fill in the Blank

1. Fill in the blanks for this code:

```

#include _____
int _____(int _____, int _____);
int main() {
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", _____);
    printf("G.C.D of %d and %d is %d.", n1, n2, _____);
    _____;
}

int hcf(_____) {
    if (n2 != 0)
        return _____(_____);
    else
        _____;
}

```

```
}
```

Answer:

```
#include <stdio.h>
int hcf(int n1, int n2);
int main() {
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);
    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));
    return 0;
}
```

```
int hcf(int n1, int n2) {
    if (n2 != 0)
        return hcf(n2, n1 % n2);
    else
        return n1;
}
```

2. Fill in the blanks and determine the output if base = 3 and a = 2:

```
#include <stdio.h>
int _____(_____ n1, _____ n2);
int main() {
    int base, a, result;
    printf("Enter base number: ");
    _____("%d", &_____);
    printf("Enter power number(positive integer): ");
    scanf("%d", &_____);
    result = _____(_____, _____);
    printf("%d^%d = %d", base, a, result);
    _____;
}
```

```
int power(_____) {
    if (a != 0)
```

```

        _____ (base * power(_____));
    else
        _____ ____;
}

```

Answer:

```

#include <stdio.h>
int power(int n1, int n2);
int main() {
    int base, a, result;
    printf("Enter base number: ");
    scanf("%d", &base);
    printf("Enter power number(positive integer): ");
    scanf("%d", &a);
    result = power(base, a);
    printf("%d^%d = %d", base, a, result);
    return 0;
}

int power(int n1, int n2) {
    if (a != 0)
        return (base * power(base, a - 1));
    else
        return 1;
}
9

```

3. Fill in the blanks and determine the output if n1 = 27 and n2 = 34.

```

#include <stdio.h>

int main() {

    int __, __, ____;

    printf("Enter two positive integers: ");

```

```
scanf("_____", &n1, &n2);
```

```
// maximum number between n1 and n2 is stored in max  
_____ = (n1 > n2) ? n1 : n2;
```

```
while (____) {  
    if ((_____) && (____)) {  
        printf("The LCM of %d and %d is %d.", n1, n2, _____);  
        _____;  
    }  
    ++____;  
}  
return 0;  
}
```

Answer:

```
#include <stdio.h>
```

```
int main() {
```

```
    int n1, n2, max;
```

```
    printf("Enter two positive integers: ");  
    scanf("%d %d", &n1, &n2);
```

```
// maximum number between n1 and n2 is stored in max  
max = (n1 > n2) ? n1 : n2;
```

```
while (1) {  
    if ((max % n1 == 0) && (max % n2 == 0)) {  
        printf("The LCM of %d and %d is %d.", n1, n2, max);  
        break;  
    }  
    ++max;  
}  
return 0;
```

```
}
```

918

Determine the Output

1. Determine the output of this code:

```
#include <stdio.h>
int main() {
    int i, j, rows;
    printf("Enter the number of rows: ");
    scanf("%d", &rows);
    for (i = 1; i <= rows; ++i) {
        for (j = 1; j <= i; ++j) {
            printf("%d ", j);
        }
        printf("\n");
    }
    return 0;
}
```

Answer:

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

2. Find the output of this code, which solves for roots of quadratic equations, given $a = 2.3$, $b = 4$, $c = 5.6$:

```
#include <math.h>
#include <stdio.h>
int main() {
    double a, b, c, discriminant, root1, root2, realPart, imagPart;
    printf("Enter coefficients a, b and c: ");
    scanf("%lf %lf %lf", &a, &b, &c);
```



```

discriminant = b * b - 4 * a * c;

// condition for real and different roots
if (discriminant > 0) {
    root1 = (-b + sqrt(discriminant)) / (2 * a);
    root2 = (-b - sqrt(discriminant)) / (2 * a);
    printf("root1 = %.2lf and root2 = %.2lf", root1, root2);
}

// condition for real and equal roots
else if (discriminant == 0) {
    root1 = root2 = -b / (2 * a);
    printf("root1 = root2 = %.2lf;", root1);
}

// if roots are not real
else {
    realPart = -b / (2 * a);
    imagPart = sqrt(-discriminant) / (2 * a);
    printf("root1 = %.2lf+%.2lfi and root2 = %.2f-%.2fi", realPart,
    imagPart, realPart, imagPart);
}

return 0;
}

```

Answer:

root1 = -0.87+1.30i and root2 = -0.87-1.30i

3. Given this code, which finds the GCD of 2 numbers, determine the output given the numbers 24 and 36:

```

#include <stdio.h>
int main()
{
    int n1, n2, i, gcd;

```

```

printf("Enter two integers: ");
scanf("%d %d", &n1, &n2);

for(i=1; i <= n1 && i <= n2; ++i)
{
    // Checks if i is factor of both integers
    if(n1%i==0 && n2%i==0)
        gcd = i;
}

printf("G.C.D of %d and %d is %d", n1, n2, gcd);

return 0;
}

```

Conversions

1. Convert the following code into a while-loop:

```

#include <stdio.h>

int main() {
    int i;

    for (i = 1; i < 11; ++i)
    {
        printf("%d ", i);
    }
    return 0;
}

```

Answer:

```

#include <stdio.h>

int main() {

```

```

    int i = 1; // Initialize i to 1

    while (i < 11) {
        printf("%d ", i);
        i++; // Increment i
    }

    return 0;
}

```

2. Convert this code into an if-else loop:
 // Print numbers from 1 to 5

```

#include <stdio.h>
int main() {
    int i = 1;

    while (i <= 5) {
        printf("%d\n", i);
        ++i;
    }

    return 0;
}

```

Answer:

```

#include <stdio.h>

```

```

int main() {
    int i = 1;

    if (i <= 5) {
        printf("%d\n", i);
        ++i;
    } else {
        return 0; // Exiting the loop
    }
}

```

```
}  
  
return 0;  
}
```

3. Convert this code into a for loop:

// Program to add numbers until the user enters zero

```
#include <stdio.h>  
int main() {  
    double number, sum = 0;  
  
    // the body of the loop is executed at least once  
    do {  
        printf("Enter a number: ");  
        scanf("%lf", &number);  
        sum += number;  
    }  
    while(number != 0.0);  
  
    printf("Sum = %.2lf",sum);  
  
    return 0;  
}
```

Answer:

```
#include <stdio.h>  
  
int main() {  
    double number, sum = 0;  
  
    for (;;) { // Infinite loop  
        printf("Enter a number: ");  
        scanf("%lf", &number);  
        sum += number;  
        if (number == 0.0) // Break condition
```

```
        break;
    }

    printf("Sum = %.2lf", sum);

    return 0;
}
```

Determine the Value

Given the following values:

```
int a = 7;
float b = 9;
char c = '7'
unsigned d = 40;
```

Determine the following outputs of each variable:

1. int r0 = b/2;
2. int r1 = b/3;
3. float r2 = a/4;
4. int r3 = d/4 + 0.1;
5. int r4 = c - '0';
6. unsigned r5 = d/a++;
7. unsigned r6 = d % b;
8. unsigned r7 = UINT_MAX + a;
9. char r8 = c + 'a'
10. float r9 = INT_MAX + 0.8;

Ans:

1. 4
2. 3
3. 1
4. 10
5. 7
6. 5
7. 4

8. 6

9. ASCII value of 152

10. -0.2