print-apc.txt

```
// File: tuition-c/APC-S-001.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-S-001.c

#include<stdio.h>
int main() {
    int a = 9, b = 4, c;
    c = a + b;
    printf("A + B = %d\n", c);
    c = a / b;
    printf("A / B = %d\n", c);
    c = a % b;
    printf("A %% B = %d\n", c);
    return 0;
}

/* --- End of tuition-c/APC-S-001.c --- */

// File: tuition-c/APC-S-002.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-S-002.c

#include<stdio.h>
int main() {
    int a = 5, b = 5, c = 10;
    printf("a = b = %d\n", a == b);
    printf("a > b = %d\n", a > b);
    printf("a < b = %d\n", a < b);
    return 0;
}

/* --- End of tuition-c/APC-S-002.c --- */

// File: tuition-c/APC-S-003.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-S-003.c

#include <stdio.h>
int main() {
    int a = 5, b = 5, c = 10, result;

    result = (a == b) && (c > b);
    printf("Result is %d\n", result);

    result = (a == b) && (c < b);
    printf("Result is %d\n", result);

    result = (a != b) || (c < b);
    printf("Result is %d\n", result);

    result = (a != b) || (c < b);
    printf("Result is %d\n", result);

    result = !(a != b);
    printf("Result is %d\n", result);

    result = !(a == b);
    printf("Result is %d\n", result);

    return 0;
```

```
}

/* --- End of tuition-c/APC-S-003.c --- */

// File: tuition-c/APC-SPS-001.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-001.c

// WAP to perform arithmatic operation on integer data

#include<stdio.h>
int main() {
    int a, b, sum, sub, multi, div, mod;
    printf("Enter 1st number : ");
    scanf("%d", &a);
    printf("Enter 2nd number : ");
    scanf("%d", &b);
    sum = a + b;
    sub = a - b;
    multi = a * b;
    div = a / b;
    mod = a % b;
    printf("\nSum = %d", sum);
    printf("\nSubtraction = %d", sub);
    printf("\nMultiplication = %d", multi);
    printf("\nDivision = %d", div);
    printf("\nModulas = %d", div);
    return 0;
}

/* --- End of tuition-c/APC-SPS-001.c --- */

// File: tuition-c/APC-SPS-002.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-002.c

/* WAP to swap two integers. Display both numbers
before and after swap */

#include<stdio.h>
int main() {
    int a = 10, b = 20, temp;
    printf("Before swap A : %d, B : %d", a, b);
    temp = a;
    a = b;
    b = temp;
    printf("\nAfter swap A : %d, B : %d", a, b);
    return 0;
}

/* --- End of tuition-c/APC-SPS-002.c --- */

// File: tuition-c/APC-SPS-003.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-003.c

/* Bitwise AND '&' */

#include<stdio.h>
int main() {
    unsigned int a = 4, b = 5, c = 6;
    unsigned int x, y;
    x = a & b;
```

```
    y = b & c;
    printf("x = %u y = %u", x, y);
    return 0;
}

/* --- End of tuition-c/APC-SPS-003.c --- */

// File: tuition-c/APC-SPS-004.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-004.c

#include<stdio.h>
int main() {
    int x = 25, y = 19, z;
    z = x - y;
    z = z & x ;
    printf("Z = %d", z);
    return 0;
}

/* --- End of tuition-c/APC-SPS-004.c --- */

// File: tuition-c/APC-SPS-005.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-005.c

/* Bitwise OR '|' */

#include<stdio.h>
int main() {
    int x = 12, y = 14, z = 10, res;
    x++;
    z++;
    x = x + y + z;
    res = x | y;
    z = res | z;
    printf("x = %d y = %d z = %d res = %d", x, y, z, res);
    return 0;
}

/* --- End of tuition-c/APC-SPS-005.c --- */

// File: tuition-c/APC-SPS-006.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-006.c

/* Bitwise NOT '~' */

#include<stdio.h>
int main() {
    int x = 12, y = 15, z = 21;
    int res, res1, res2;
    res = x > 10;
    res1 = ~res;
    res2 = ~x;
    printf("REs = %d, Res1 = %d, Res2 = %d", res, res1, res2);
    return 0;
}

/* --- End of tuition-c/APC-SPS-006.c --- */

// File: tuition-c/APC-SPS-007.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-007.c
```

```c
/* WAP to check a number is even or odd using bitwise operator */

#include<stdio.h>
int main() {
    int x, res = 1;
    printf("Enter a number : ");
    scanf("%d", &x);
    res = res & x;
    if (res == 0) {
        printf("\nInput %d is a even number.", x);
    }
    else {
        printf("\nInput %d is a odd number.", x);
    }
    return 0;
}

/* --- End of tuition-c/APC-SPS-007.c --- */

// File: tuition-c/APC-SPS-008.c
// URL: https://github.com/notamitgamer/bsc/blob/main/tuition-c/APC-SPS-008.c

/* WAP to calculate area of circle by accepting radius as input */
/* Author : Amit Dutta, Date : 15th September, 2025 */

#include<stdio.h>
#include<math.h>
int main() {
    double r, area;
    printf("Enter the radius of circle : ");
    scanf("%lf", &r);
    area = M_PI * r * r;
    printf("\nArea : %lf", area);
    return 0;
}

/* --- End of tuition-c/APC-SPS-008.c --- */
```