

Q1

CPH JUDGE: RESULTS

style.css

1_1.html

2.html

20jan.html U

20_01.c

TC 1

Passed 29ms

Input: 2

Expected Output: 10

Received Output: 10

TC 2

Passed 16ms

Input: 8

Expected Output: 1000

Received Output: 1000

TC 3

Passed 15ms

Input: 15

Expected Output: 1111

Received Output: 1111

COA > Assignment 1 > ass_1_1.c > dec_bin(int)

```
1  #include <stdio.h>
2  //1
3  void dec_bin(int a)
4  {
5      int b[100];
6      int i = 0;
7      while (a >= 0)
8      {
9          if (a == 0)
10         {
11             b[i] = 0;
12             break;
13         }
14         else if (a == 1)
15         {
16             b[i] = 1;
17             break;
18         }
19         else{
20             b[i]=(a%2);
21         }
22         i++;
23         a=(a/2);
24     }
25     for (int j = i; j >=0 ; j--)
26     {
27         printf("%d",b[j]);
28     }
29 };
30 int main()
31 {
32     int a;
33     scanf("%d",&a);
34     dec_bin(a);
35     return 0;
36 }
```

Q2

CPH JUDGE: RESULTS

1(Biodata).html style.css 1_1.html 2.html 20jan.html U

TC 1
Passed 15ms

Input: 2
1 0
Expected Output: 2
Received Output: 2

TC 2
Passed 15ms

Input: 1
1
Expected Output: 1
Received Output: 1

TC 3
Passed 18ms

Input: 3
1 0 0
Expected Output: 4
Received Output: 4

COA > C ass_1_2.c > main()
1 #include<stdio.h>
2 #include<math.h>
3
4 int bin_dec(int a[],int n){
5
6 int num=0;
7 for (int i = n-1; i >=0; i--)
8 {
9 num+=(pow(2,i)*a[i]);
10 }
11 return num;
12 }
13
14 int main(){
15
16 int n;
17 scanf("%d",&n);
18 int a[n];
19 for (int i = n-1; i >=0; i--)
20 {
21 scanf("%d",&a[i]);
22 }
23
24 printf("%d",bin_dec(a,n));
25
26
27
28 return 0;
29 }

CPH JUDGE: RESULTS odata).html style.css 1_1.html 2.html 20jan.html U 20_01.c 99.c C

Local: **ass_1_3** 2 / 2 passed

TC 1 **Passed** 15ms

Input: 5

Expected Output: Octal: 5
Hexadecimal: 5

Received Output: Octal: 5
Hexadecimal: 5

TC 2 **Passed** 17ms

Input: 20

Expected Output: Octal: 24
Hexadecimal: 14

Received Output: Octal: 24
Hexadecimal: 14

+ New Testcase

☐ Set ONLINE_JUDGE

Run All + New

Stop Delete

```

COA > C ass_1_3.c > dec_to_octal(int)
1  // Function to convert decimal to octal
2
3  void dec_to_octal(int n)
4  {
5      int oct[100], i = 0;
6
7      if (n == 0)
8      {
9          printf("0");
10         return;
11     }
12
13     while (n > 0)
14     {
15         oct[i++] = n % 8;
16         n /= 8;
17     }
18
19     for (int j = i - 1; j >= 0; j--)
20         printf("%d", oct[j]);
21 }
22
23 void dec_to_hex(int n)
24 {
25     char hex[100];
26     int i = 0;
27
28     if (n == 0)
29     {
30         printf("0");
31         return;
32     }
33
34     while (n > 0)
35     {
36         int rem = n % 16;
37         if (rem < 10)
38             hex[i++] = rem + '0';
39         else
40             hex[i++] = rem - 10 + 'A';
41         n /= 16;
42     }

```

CPH JUDGE: RESULTS odata).html style.css 1_1.html 2.html 20jan.html U 20_01.c C

Local: **ass_1_3** 2 / 2 passed

TC 1 **Passed** 15ms

Input: 5

Expected Output: Octal: 5
Hexadecimal: 5

Received Output: Octal: 5
Hexadecimal: 5

TC 2 **Passed** 17ms

Input: 20

Expected Output: Octal: 24
Hexadecimal: 14

Received Output: Octal: 24
Hexadecimal: 14

+ New Testcase

☐ Set ONLINE_JUDGE

Run All + New

Stop Delete

```

COA > C ass_1_3.c > main()
22
23 void dec_to_hex(int n)
24 {
25     char hex[100];
26     int i = 0;
27
28     if (n == 0)
29     {
30         printf("0");
31         return;
32     }
33
34     while (n > 0)
35     {
36         int rem = n % 16;
37         if (rem < 10)
38             hex[i++] = rem + '0';
39         else
40             hex[i++] = rem - 10 + 'A';
41         n /= 16;
42     }
43
44     for (int j = i - 1; j >= 0; j--)
45         printf("%c", hex[j]);
46 }
47
48 int main()
49 {
50     int n;
51     scanf("%d", &n);
52
53     printf("Octal: ");
54     dec_to_octal(n);
55
56     printf("\nHexadecimal: ");
57     dec_to_hex(n);
58
59     return 0;
60 }
61

```

Q4

Local:
ass_1_4

2 / 2 passed

TC 1

Passed 18ms

Input:

100001

Expected Output:

100 : 4 001 : 1
Octal: 41

Received Output:

100 : 4 001 : 1
Octal: 41

TC 2

Passed 15ms

Input:

110101

Expected Output:

110 : 6 101 : 5
Octal: 65

Received Output:

110 : 6 101 : 5
Octal: 65

+ New Testcase

Set ONLINE_JUDGE

Run All

+ New

COA > Assignment 1 > ass_1_4.c > main()

```
1 #include <stdio.h>
2 #include <string.h>
3
4 int main()
5 {
6     char bin[100];
7     int len, i, value;
8
9     scanf("%s", bin);
10    len = strlen(bin);
11
12    int pad = (3 - (len % 3)) % 3;
13
14    for (i = 0; i < len; i++)
15    {
16        printf("%c", bin[i]);
17        if ((i + pad + 1) % 3 == 0)
18        {
19
20            value = (bin[i - 2] - '0') * 4 +
21                    (bin[i - 1] - '0') * 2 +
22                    (bin[i] - '0') * 1;
23            printf(" : %d ", value);
24        }
25    }
26
27    printf("\nOctal: ");
28
29
30
31    int start = (len + pad) % 3;
32    for (i = 0; i < len; i += 3)
33    {
34        int idx = i - pad;
35        if (idx < 0) continue;
36
37        value = (bin[idx] - '0') * 4 +
38                (bin[idx + 1] - '0') * 2 +
39                (bin[idx + 2] - '0');
40        printf("%d", value);
41    }
42
43    return 0;
44 }
```

Q5

Local:
ass_1_5

2 / 2 passed

TC 1

Passed 17ms

Input: 10101111

Expected Output: AF

Received Output: AF

TC 2

Passed 17ms

Input: 10001010

Expected Output: 8A

Received Output: 8A

+ New Testcase

Set ONLINE_JUDGE

Support

Run All

+ New

Stop

Info

Delete

COA > Assignment 1 > C ass_1_5.c > main()

```
3 int main(){
4     char bin[100];
5     int len, i, value;
6
7     scanf("%s", bin);
8     len = strlen(bin);
9
10    int pad = (4 - (len % 4)) % 4;
11    for (i = 0; i < len; i++)
12    {
13        if ((i + pad + 1) % 4 == 0)
14        {
15            value = ( (bin[i - 3] - '0') * 8 +
16                    (bin[i - 2] - '0') * 4 +
17                    (bin[i - 1] - '0') * 2 +
18                    (bin[i] - '0') * 1 );
19
20            if (value > 9){
21                if (value == 10)
22                {
23                    printf("A");
24                }
25                else if (value == 11)
26                {
27                    printf("B");
28                }
29                else if (value == 12)
30                {
31                    printf("C");
32                }
33                else if (value == 13)
34                {
35                    printf("D");
36                }
37                else if (value == 14)
38                {
39                    printf("E");
40                }
41                else if (value == 15)
42                {
43                    printf("F");
44                }
45            }
46            else printf("%d", value);
47        }
48    }
49    return 0;
```

Q6

CPH JUDGE: RESULTS

ass_1_5.cass_1_6.cass_1_7.cass_1_8.cass_1_9.cass_1_10.cass_1_11.c

Local:
ass_1_6

1 / 1 passed

TC 1
Passed 32ms

Input:
3
1 5 0

Expected Output:
positional power
method:- 104
iterative
multiplication
method:- 104

Received Output:
positional power
method:- 104
iterative
multiplication
method:- 104

+ New Testcase

Set ONLINE_JUDGE

Support

Feedback

Bugs

COA > Assignment 1 > ass_1_6.c > ...

```
1 #include<stdio.h>
2 int oct_dec_method1(int a[],int n){//positional power method
3
4     int num=0;
5     for (int i = n-1; i >=0; i--)
6     {
7         num+=(pow(8,i)*a[i]);
8     }
9     return num;
10 }
11
12 int oct_dec_method2(int a[],int n){//iterative multiplication method
13     int decimal=0;
14     for (int i = n-1; i >=0; i--) {
15         decimal = decimal * 8 + a[i];
16     }
17     return decimal;
18 }
19
20 int main(){
21
22     int n;
23     scanf("%d", &n);
24     int a[n];
25     for (int i = n-1; i >=0; i--)
26     {
27         scanf("%d",&a[i]);
28     }
29     printf("positional power method:- %d \n",oct_dec_method1(a,n));
30     printf("iterative multiplication method:- %d",oct_dec_method2(a,n));
31
32
33     return 0;
34 }
```

Q7

CPH JUDGE: RESULTS

ass_1_5.cass_1_6.cass_1_7.cXass_1_8.cass_1_9

Local:
ass_1_70 / 1 passed

TC 1

Input: 9ACopy

Expected Output: Copy
Binary equivalent:
10011010

+ New Testcase

Set ONLINE_JUDGE

Support

FeedbackBugs

COA > Assignment 1 > ass_1_7.c > main()

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char hex[20];
6      int i;
7
8      char binary[] = {
9          "0000", "0001", "0010", "0011",
10         "0100", "0101", "0110", "0111",
11         "1000", "1001", "1010", "1011",
12         "1100", "1101", "1110", "1111"
13     };
14
15     scanf("%s", hex);
16
17     printf("Binary equivalent: ");
18
19     for (i = 0; hex[i] != '\0'; i++) {
20         if (hex[i] >= '0' && hex[i] <= '9') {
21             printf("%s", binary[hex[i] - '0']);
22         } else if (hex[i] >= 'A' && hex[i] <= 'F') {
23             printf("%s", binary[hex[i] - 'A' + 10]);
24         } else if (hex[i] >= 'a' && hex[i] <= 'f') {
25             printf("%s", binary[hex[i] - 'a' + 10]);
26         } else {
27             printf("\nInvalid hexadecimal digit!");
28             return 1;
29         }
30     }
31
32     return 0;
33 }
```

Q8

CPH JUDGE: RESULTS

ass_1_5.cass_1_6.cass_1_7.cass_1_8.cXass_1_9.cass

Local:
ass_1_81 / 1 passed

TC 1
Passed41ms

Input: Copy
1101

Expected Output: Copy
1's Complement =
0010
2's Complement =
0011

Received Output: Copy
1's Complement =
0010
2's Complement =
0011

+ New Testcase

Set ONLINE_JUDGE

Support

Feedback

Bugs

COA > Assignment 1 > C ass_1_8.c > main()

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char binary[50], ones[50], twos[50];
6      int i, carry = 1;
7
8      scanf("%s", binary);
9
10     for (i = 0; binary[i] != '\0'; i++) {
11         if (binary[i] == '0')
12             ones[i] = '1';
13         else
14             ones[i] = '0';
15     }
16     ones[i] = '\0';
17
18     strcpy(twos, ones);
19
20     for (i = strlen(twos) - 1; i >= 0; i--) {
21         if (twos[i] == '1' && carry == 1) {
22             twos[i] = '0';
23         } else if (twos[i] == '0' && carry == 1) {
24             twos[i] = '1';
25             carry = 0;
26         }
27     }
28
29     printf("1's Complement = %s\n", ones);
30     printf("2's Complement = %s\n", twos);
31
32     return 0;
33 }
34
```


Q9

CPH JUDGE: RESULTS

ass_1_5.cass_1_6.cass_1_7.cass_1_8.cass_1_9.c

Local:
ass_1_92 / 2 passed

TC 1

Passed23ms

Input:11000011

Expected Output:Binary Sum = 1111

Received Output:Binary Sum = 1111

TC 2

Passed18ms

Input:11111010

Expected Output:Binary Sum = 11001

Received Output:Binary Sum = 11001

+ New Testcase

☐ Set ONLINE_JUDGE

Run All

+ New

Stop

Delete

COA > Assignment 1 > ass_1_9.c > main()

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char bin1[50], bin2[50], result[51];
6      int i, j, k, carry = 0;
7      int sum;
8
9      scanf("%s", bin1);
10     scanf("%s", bin2);
11
12     i = strlen(bin1) - 1;
13     j = strlen(bin2) - 1;
14     k = 0;
15     while (i >= 0 || j >= 0) {
16         sum = carry;
17
18         if (i >= 0)
19             sum += bin1[i--] - '0';
20         if (j >= 0)
21             sum += bin2[j--] - '0';
22
23         result[k++] = (sum % 2) + '0';
24         carry = sum / 2;
25     }
26
27     if (carry)
28         result[k++] = '1';
29
30     result[k] = '\0';
31
32     for (i = 0, j = k - 1; i < j; i++, j--) {
33         char temp = result[i];
34         result[i] = result[j];
35         result[j] = temp;
36     }
37
38     printf("Binary Sum = %s\n", result);
39
40     return 0;
41 }
```

Q10

```
COA > Assignment 1 > C ass_1_10.c > main()
1  #include <stdio.h>
2
3  int binaryToDecimal(long int binary) {
4      int decimal = 0, base = 1, digit;
5      while (binary > 0) {
6          digit = binary % 10;
7          decimal += digit * base;
8          base *= 2;
9          binary /= 10;
10     }
11     return decimal;
12 }
13
14 long int decimalToBinary(int decimal) {
15     long int binary = 0;
16     int rem, base = 1;
17     while (decimal > 0) {
18         rem = decimal % 2;
19         binary += rem * base;
20         base *= 10;
21         decimal /= 2;
22     }
23     return binary;
24 }
25
26 int main() {
27     long int bin1, bin2;
28     int dec1, dec2, result;
29     int choice;
30
31     printf("Enter Binary numbers:");
32     scanf("%ld", &bin1);
33     scanf("%ld", &bin2);
34
35     dec1 = binaryToDecimal(bin1);
36     dec2 = binaryToDecimal(bin2);
37
38     printf("MENU\n");
39     printf("1. Addition\n");
40     printf("2. Multiplication\n");
41     printf("3. Division\n");
42     printf("Enter your choice: \n");
43     scanf("%d", &choice);
44
45     switch (choice) {
46         case 1:
47             result = dec1 + dec2;
```

```
COA > Assignment 1 > C ass_1_10.c > main()
26 int main() {
44     // result = dec1 + dec2;
45     switch (choice) {
46         case 1:
47             result = dec1 + dec2;
48             printf("Binary Sum = %ld\n", decimalToBinary(result));
49             break;
50
51         case 2:
52             result = dec1 * dec2;
53             printf("Binary Product = %ld\n", decimalToBinary(result));
54             break;
55
56         case 3:
57             if (dec2 == 0) {
58                 printf("Division by zero not possible\n");
59             } else {
60                 result = dec1 / dec2;
61                 printf("Binary Quotient = %ld\n", decimalToBinary(result));
62             }
63             break;
64
65         default:
66             printf("Invalid choice\n");
67     }
68
69     return 0;
70 }
71
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Enter Binary numbers:110
101
MENU
1. Addition
2. Multiplication
3. Division
Enter your choice:
2
Binary Product = 11110
```

Q11

```
COA > Assignment 1 > C ass_1_11.c > ...
1  #include <stdio.h>
2  #include <math.h>
3
4  void decimalToBinary(int num, int bits, int binary[]) {
5      int i;
6      for (i = bits - 1; i >= 0; i--) {
7          binary[i] = num % 2;
8          num = num / 2;
9      }
10 }
11
12 void onesComplement(int binary[], int bits) {
13     for (int i = 0; i < bits; i++) {
14         binary[i] = (binary[i] == 0) ? 1 : 0;
15     }
16 }
17
18 void twosComplement(int binary[], int bits) {
19     int carry = 1;
20     for (int i = bits - 1; i >= 0; i--) {
21         int sum = binary[i] + carry;
22         binary[i] = sum % 2;
23         carry = sum / 2;
24     }
25 }
26
27 void displayBinary(int binary[], int bits) {
28     for (int i = 0; i < bits; i++) {
29         printf("%d", binary[i]);
30     }
31     printf("\n");
32 }
33
34 int main() {
35     int number, bits;
36     int binary[32];
37
38     printf("Enter decimal number: ");
39     scanf("%d", &number);
40
41     printf("Enter word length (bits): ");
42     scanf("%d", &bits);
43
44     int absNum = abs(number);
45     decimalToBinary(absNum, bits, binary);
46 }
```

```
COA > Assignment 1 > C ass_1_11.c > ...
27 void displayBinary(int binary[], int bits) {
28     for (int i = 0; i < bits; i++) {
31         printf("%d", binary[i]);
32     }
33 }
34
35 int main() {
36     int number, bits;
37     int binary[32];
38
39     printf("Enter decimal number: ");
40     scanf("%d", &number);
41
42     printf("Enter word length (bits): ");
43     scanf("%d", &bits);
44
45     int absNum = abs(number);
46     decimalToBinary(absNum, bits, binary);
47
48     if (number < 0) {
49         onesComplement(binary, bits);
50         twosComplement(binary, bits);
51         printf("2's Complement Representation: ");
52     } else {
53         printf("Binary Representation: ");
54     }
55
56     displayBinary(binary, bits);
57
58     int min = -pow(2, bits - 1);
59     int max = pow(2, bits - 1) - 1;
60
61     printf("Range for %d-bit word: %d to %d\n", bits, min, max);
62
63     return 0;
64 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Enter decimal number: 4
Enter word length (bits): 4
Binary Representation: 0100
Range for 4-bit word: -8 to 7
PS C:\Users\Dell\OneDrive\Desktop\code\COA\Assignment 1>
```

Q12

```

COA > Assignment 1 > C ass_1_12.c > main()
1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char a[50], b[50], sum[50];
6      int i, carry = 0, n;
7      int valid = 0;
8
9      printf("Enter binary number: ");
10     scanf("%s", a);
11     scanf("%s", b);
12
13     n = strlen(a);
14
15     for (i = n - 1; i >= 0; i--) {
16         int total = (a[i] - '0') + (b[i] - '0') + carry;
17         sum[i] = (total % 2) + '0';
18         carry = total / 2;
19     }
20
21     if (carry == 1) {
22         carry = 1;
23         for (i = n - 1; i >= 0; i--) {
24             int total = (sum[i] - '0') + carry;
25             sum[i] = (total % 2) + '0';
26             carry = total / 2;
27         }
28     }
29     sum[n] = '\0';
30     for (int i = 0; i < strlen(sum); i++) {
31         if (sum[i] == '0') {
32             valid = 1;
33             break;
34         }
35     }
36
37     printf(" \n  Final Result (1's Complement Arithmetic): %s \n", sum);
38
39     if (!valid)
40         printf("Validation: Invalid result (Negative Zero)\n");
41     else
42         printf("Validation: Valid result\n");
43
44     return 0;
45 }
46

```

```

36
37     printf(" \n  Final Result (1's Complement Arithmetic): %s \n", sum);
38
39     if (!valid)
40         printf("Validation: Invalid result (Negative Zero)\n");
41     else
42         printf("Validation: Valid result\n");
43
44     return 0;
45 }
46
47
48

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Validation: Valid result

PS C:\Users\Dell\OneDrive\Desktop\code\COA\Assignment 1> cd "c:\Users\Dell\OneDrive\Desktop\code\COA\Assignment 1"

Enter binary number: 1101

1011

Final Result (1's Complement Arithmetic): 1001

Validation: Valid result


Q13

COA > Assignment 1 >  ass_1_13.c >  main()

```
1  #include <stdio.h>
2  #include <string.h>
3
4  void binaryAdd(char a[], char b[], char result[], int *carryOut) {
5      int n = strlen(a);
6      int carry = 0;
7
8      for (int i = n - 1; i >= 0; i--) {
9          int sum = (a[i] - '0') + (b[i] - '0') + carry;
10         result[i] = (sum % 2) + '0';
11         carry = sum / 2;
12     }
13
14     *carryOut = carry;
15     result[n] = '\0';
16 }
17
18 int main() {
19     char a[20], b[20], result[20];
20     int carryOut;
21
22     printf("Enter binary number: ");
23     scanf("%s", a);
24     scanf("%s", b);
25
26     if (strlen(a) != strlen(b)) {
27         printf("Error: Binary numbers must be of same length.\n");
28         return 0;
29     }
30
31     binaryAdd(a, b, result, &carryOut);
32
33     printf("\nSigned 2's Complement Result: %s\n", result);
34     printf("Unsigned Carry Out: %d\n", carryOut);
35
36     if ((a[0] == b[0]) && (result[0] != a[0])) {
37         printf("Overflow Detected \n");
38     } else {
39         printf("No Overflow \n");
40     }
41
42     return 0;
43 }
44
```

```
25
26     if (strlen(a) != strlen(b)) {
27         printf("Error: Binary numbers must be of same length.\n");
28         return 0;
29     }
30
31     binaryAdd(a, b, result, &carryOut);
32
33     printf("\nSigned 2's Complement Result: %s\n", result);
34     printf("Unsigned Carry Out: %d\n", carryOut);
35
36     if ((a[0] == b[0]) && (result[0] != a[0])) {
37         printf("Overflow Detected \n");
38     } else {
39         printf("No Overflow \n");
40     }
41
42     return 0;
43 }
44
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

 cd "c:\Users\Dell\COA

Enter binary number: 010000
010000

Signed 2's Complement Result: 100000
Unsigned Carry Out: 0
Overflow Detected

Q14

```
COA > Assignment 1 > C ass_1_14.c > main()
1  #include <stdio.h>
2  #include <string.h>
3
4  void onesComplement(char b[]) {
5      for (int i = 0; i < strlen(b); i++) {
6          b[i] = (b[i] == '0') ? '1' : '0';
7      }
8  }
9
10 int binaryAdd(char a[], char b[], char result[]) {
11     int carry = 0;
12     int n = strlen(a);
13
14     for (int i = n - 1; i >= 0; i--) {
15         int sum = (a[i] - '0') + (b[i] - '0') + carry;
16         result[i] = (sum % 2) + '0';
17         carry = sum / 2;
18     }
19
20     result[n] = '\0';
21     return carry;
22 }
23
24 int main() {
25     char a[20], b[20], result[20];
26
27     printf("Enter minuend (A): ");
28     scanf("%s", a);
29
30     printf("Enter subtrahend (B): ");
31     scanf("%s", b);
32
33     if (strlen(a) != strlen(b)) {
34         printf("Error: Both numbers must have same length.\n");
35         return 0;
36     }
37
38     onesComplement(b);
39
40     int carry = binaryAdd(a, b, result);
41
42     if (carry == 1) {
43         for (int i = strlen(result) - 1; i >= 0; i--) {
44             if (result[i] == '0') {
45                 result[i] = '1';
46                 break;
47             } else {
48                 result[i] = '0';
49             }
50         }
51         printf("\nResult (Positive): %s\n", result);
52     } else {
53         onesComplement(result);
54         printf("\nResult (Negative): -%s\n", result);
55     }
56
57     int valid = 0;
58     for (int i = 0; i < strlen(result); i++) {
59         if (result[i] == '0') {
60             valid = 1;
61             break;
62         }
63     }
64
65     if (!valid)
66         printf("Validation: Invalid result (Negative Zero)\n");
67     else
68         printf("Validation: Valid result\n");
69
70     return 0;
71 }
72
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Dell\OneDrive\Desktop\code> cd "c:\Users\Dell\OneDrive\Desktop\code\COA\Assignment 1\" ; if
● Enter minuend (A): 101100
Enter subtrahend (B): 010011

Result (Positive): 011001
Validation: Valid result
○ PS C:\Users\Dell\OneDrive\Desktop\code\COA\Assignment 1> |
```