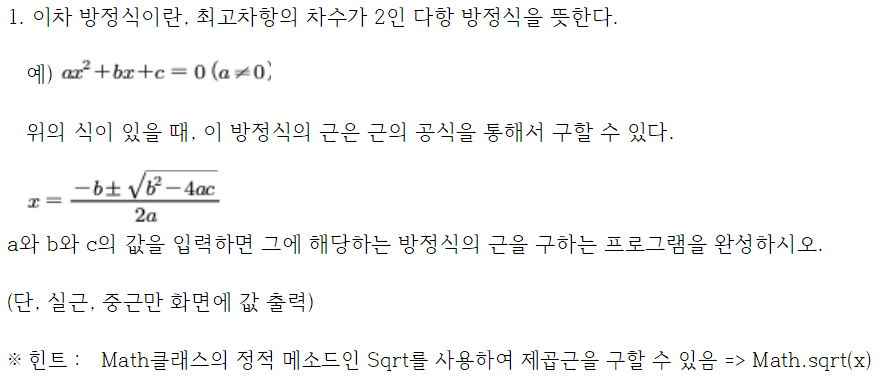
**윈도우즈 프로그래밍**

**컴퓨터공학과**

**20204062 이인규**

****

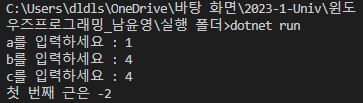
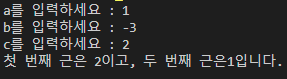
**소스 코드**

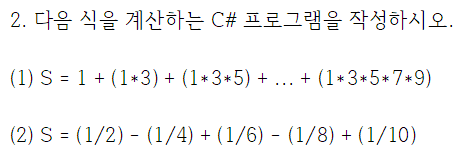
|  |
| --- |
| **int a, b, c;**  **Console.Write("a를 입력하세요 : ");**  **a = Int32.Parse(Console.ReadLine());**  **Console.Write("b를 입력하세요 : ");**  **b = Int32.Parse(Console.ReadLine());**  **Console.Write("c를 입력하세요 : ");**  **c = Int32.Parse(Console.ReadLine());**  **double answer1 = (-b + Math.Sqrt(b\*b - 4\*a\*c)) / 2\*a;**  **double answer2 = (-b - Math.Sqrt(b\*b - 4\*a\*c)) / 2\*a;**  **if ((Math.Sqrt(b\*b - 4\*a\*c)) > 0){**  **Console.WriteLine("첫 번째 근은 " + answer1 + "이고, 두 번째 근은" + answer2 + "입니다.");**  **}**  **else if((Math.Sqrt(b\*b - 4\*a\*c)) < 0){**  **Console.WriteLine("허근입니다.");**  **}**  **else{**  **Console.WriteLine("첫 번째 근은 " + answer1);**  **}** |

**소스 코드 설명**

**a, b, c 값을 Int 형으로 입력 받는다. 그 후 근의 공식에 따라서 계산을 한 후 근이 2개일때, 1개일때, 허근일때의 경우에 따라 출력을 다르게 한다.**

**실행 결과**

****

****

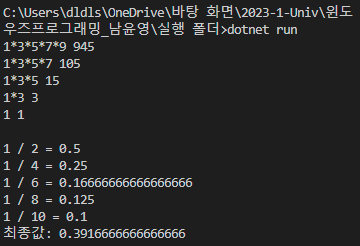
**소스 코드**

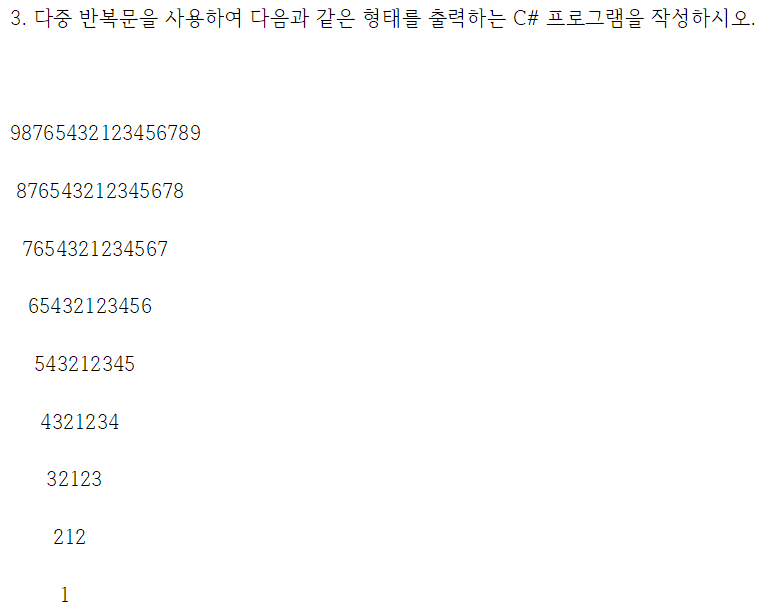
|  |
| --- |
| **for(int i=9; i>=1; i--){**  **int num1 = 0;**  **int temp = 1;**  **for(int j=1; j<=i; j++){**  **temp \*= j;**  **Console.Write(j);**    **if(j != i){**  **Console.Write("\*");**  **}**  **j = j + 1;**  **}**  **num1 += temp;**  **Console.WriteLine(" " + num1);**  **i = i - 1;**  **}**  **Console.WriteLine();**  **double num2 = 0;**  **for(int i=1; i<=5; i++){**  **double output = 1.0 / (double)(i\*2);**  **Console.WriteLine("1 / " + i\*2 + " = " + output);**  **if(i % 2 == 1){**  **num2 = num2 + output;**  **}**  **else{**  **num2 = num2 - output;**  **}**  **}**  **Console.WriteLine("최종값: " + num2);** |

**소스 코드 설명**

**2중 for문을 통해서 숫자와 \* 기호를 반복해서 출력한다. 그 후 마지막일때는 \* 기호 대신 공백 한 칸 이우 모두 합친 결과 값을 출력한다.**

**실행 결과**

****

****

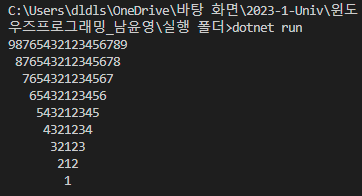
**소스 코드**

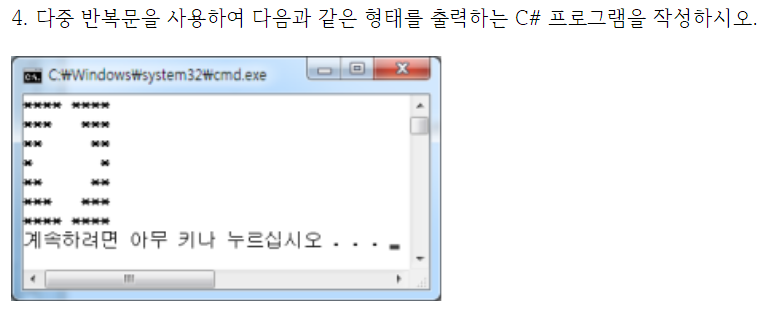
|  |
| --- |
| **for(int num=9; num>=1; num--){**  **if(num != 9){**  **for(int i=0; i<9-num; i++){**  **Console.Write(" ");**  **}**  **}**  **for(int i=num; i>=1; i--){**  **Console.Write(i);**  **}**  **for(int i=2; i<=num; i++){**  **Console.Write(i);**  **}**  **Console.WriteLine();**  **}** |

**소스 코드 설명**

**2중 포문을 만들어서 줄, 숫자 출력 총 2개의 loop를 반복한다. 또한 숫자가 1일까지 출력하는 부분과 그 뒷 부분을 나누어서 for문으로 코드를 작성하였다.**

**실행 결과**

****

****

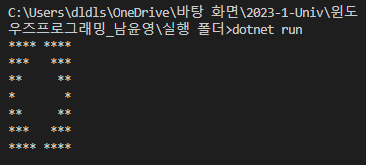
**소스 코드**

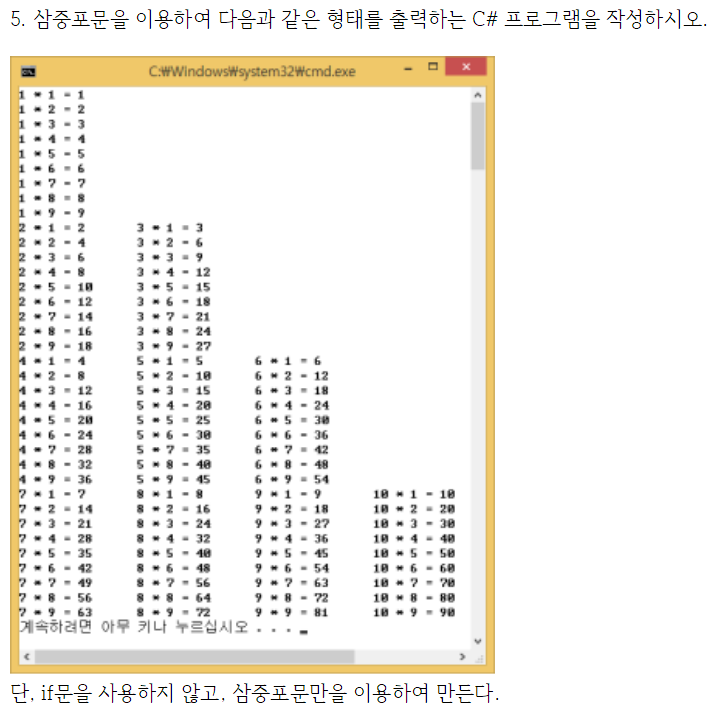
|  |
| --- |
| **int outcount = 4;**  **int delta = -1;**  **for(int i=0; i<7; i++){**  **for(int j=0; j<outcount; j++){**  **Console.Write("\*");**  **}**  **for(int j=0; j<9-2\*outcount; j++){**  **Console.Write(" ");**  **}**  **for(int j=0; j<outcount; j++){**  **Console.Write("\*");**  **}**  **if(outcount == 1){**  **delta = 1;**  **}**  **outcount += delta;**  **Console.WriteLine();**  **}** |

**소스 코드 설명**

**출력해야할 \*을 먼저 outcount 변수에 저장한다. 그 후 루프가 돌때마다 \*의 개수를 줄여주며, 그 줄어든 \*을 공백으로 출력해준다. 만약 outcount가 1이되면, 음수였던 delta 값을 양수로 바꾸어주면서, 아래의 그림을 완성시킨다.**

**실행 결과**

****

****

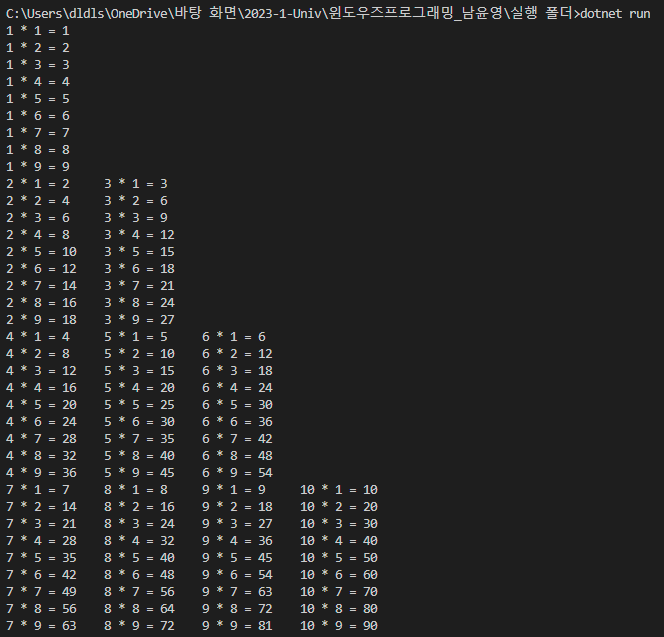
**소스 코드**

|  |
| --- |
| **int num = 1;**  **for(int i=0; i<=3; i++){**  **for(int q=1; q<=9; q++){**  **for(int p=num; p<=num+i; p++){**  **Console.Write(p + " \* " + q + " = " + (p \* q).ToString().PadRight(6, ' '));**  **}**  **Console.WriteLine();**  **}**  **num = num + i + 1;**  **}** |

**소스 코드 설명**

**3중 포문을 이용하여, 코드를 작성한다. 한 줄에 출력할 곱셈의 개수, 1~9까지 곱한 수, 출력할 줄 수 총 3개의 loop를 반복시킨다. 매 반복 횟수마다 한 줄에 출력할 곱셈의 수를 늘려가면서 계단 형식으로 출력하게 한다.**

**실행 결과**

****