**윈도우즈 프로그래밍**

**컴퓨터공학과**

**20204062 이인규**

**과제 #1**

**C#으로 프로그래밍 하시오.**

1. **다음을 계산하여 출력하시오.**
2. |b^2 - 4ac|       // ^는 제곱

|  |
| --- |
| double cal\_a(double a, double b, double c){      double dump1 = Math.Pow(b,2);      double dump2 = 4 \* a \* c;      double sum = dump1 - dump2;      return Math.Abs(sum);  }  Console.Write("1번의 a문제 입력: ");  a = double.Parse(Console.ReadLine());  b = double.Parse(Console.ReadLine());  c = double.Parse(Console.ReadLine());  Console.WriteLine(cal\_a(a, b, c)); |
| **중간 계산 과정을 dump1, dump2 변수에 저장.**  **계산을 sum 변수에 모두 합친 후, Math.Abs() 함수를 통해 절대값을 반환합니다.** |

**실행 결과**

텍스트이(가) 표시된 사진

자동 생성된 설명

**b.** log10(x+y)^3   // 밑지수 10, ^3은 세제곱

|  |
| --- |
| double cal\_b(double x, double y){      double dump1 = Math.Log10(x+y);      double dump2 = Math.Pow(dump1,3);      return dump2;  }  Console.Write("1번의 b문제 입력: ");  a = double.Parse(Console.ReadLine());  b = double.Parse(Console.ReadLine());  Console.WriteLine(cal\_b(a, b)); |
| **로그의 중간 과정을 dump1에 저장 후 제곱의 결과는 dump2에 저장.**  **그 후 dump2 반환함** |

**실행 결과**

텍스트이(가) 표시된 사진

자동 생성된 설명

**c.**  root((x1-x2)^2+(y1-y2)^2) //root 는 루트

|  |
| --- |
| double cal\_c(double x1, double x2, double y1, double y2){      double dump1 = (x1-x2) \* (x1-x2);      double dump2 = (y1-y2) \* (y1-y2);      double sum = Math.Sqrt(dump1 + dump2);      return sum;  }  Console.Write("1번의 c문제 입력: ");  a = double.Parse(Console.ReadLine());  b = double.Parse(Console.ReadLine());  c = double.Parse(Console.ReadLine());  d = double.Parse(Console.ReadLine());  Console.WriteLine(cal\_c(a, b, c, d)); |
| **각 제곱의 결과 값을 dump1, dump2 에 저장 후 더한 후 root를 씌운 값을 sum에 저장, 그 후 sum 값을 반환한다.** |

**실행 결과**

텍스트이(가) 표시된 사진

자동 생성된 설명

1. **원금과 이율, 기간을 입력받아 복리법으로 원리 합계를 구하는 프로그램**

**S = 원금 \* ( 1 + 이율 )^기간**

|  |
| --- |
| double cal\_2(double principal, double inrate, double period){      double dump1 = principal;      double dump2 = (1+inrate);      dump2 = Math.Pow(dump2, period);      double sum = dump1 \* dump2;      return sum;  }  Console.Write("2번 문제 입력: ");  a = double.Parse(Console.ReadLine());  b = double.Parse(Console.ReadLine());  c = double.Parse(Console.ReadLine());  Console.WriteLine(cal\_2(a, b, c)); |
| **(1+이율)^기간을 먼저 구한 후 dump2에 저장, 그 후 원금과 곱해준 값을 sum을 반환한다.** |

**실행 결과**

텍스트이(가) 표시된 사진

자동 생성된 설명

1. **반지름 (r)을 읽어 원의 면적을 구하시오.**

**S=4\*파이\*r^2**

|  |
| --- |
| double cal\_3(double r){      double dump1 = 4.0 \* 3.141592;      double dump2 = r \* r;      double sum = dump1 \* dump2;      return sum;  }  Console.Write("3번 문제 입력: ");  a = double.Parse(Console.ReadLine());  Console.WriteLine(cal\_3(a)); |
| **PI 값은 임의로 3.141592로 지정 후 계산 함.** |

**실행 결과**



1. **연도를 읽어 윤년인지 판별하는 프로그램**

|  |
| --- |
| void cal\_4(int year){      Console.Write(year);      if (year % 400 == 0){          Console.WriteLine("는 윤년입니다.");      }      else if(year % 100 == 0){          Console.WriteLine("는 윤년이 아닙니다.");      }      else if(year % 4 == 0){          Console.WriteLine("는 윤년입니다.");      }      else{          Console.WriteLine("는 윤년이 아닙니다.");      }  }  Console.Write("4번 문제 입력: ");  a = double.Parse(Console.ReadLine());  cal\_4((int)a); |
| **조건문을 사용해서 각 케이스 마다 윤년인 것과 윤년이 아닌 것을 분류한다.** |

**실행 결과**



1. **10개의 정수를 읽어서 제일 큰 수를 출력하는 프로그램을**

|  |
| --- |
| int cal\_5(int[] array){      int maximum = array[0];      for(int i=1; i<10; i++){          if(maximum < array[i]){              maximum = array[i];          }      }      return maximum;  }  Console.Write("5번 문제 입력: ");  for(int i=0; i<10; i++){      array[i] = int.Parse(Console.ReadLine());  }  Console.Write("제일 큰 수는: ");  Console.WriteLine(cal\_5(array)); |
| **10개의 정수를 배열 array로 입력 받아서, 가장 큰 수를 찾아 반환한다.** |

**실행 결과**

