|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2020\_1\_C\_6주 | 학번 : | 20175105 | 이름 : | 곽영주 |

* **내용 점검**

1. 다음 서술 내용이 맞으면 0, 틀리면 x 하시오.
2. C 프로그램이란 여러 함수의 집합으로 구성되는 프로그램이다. ( O )
3. 라이브러리 함수는 간단히 라이브러리(library) 또는 표준 함수(standard function)라고도 부른다. ( O )
4. 함수몸체(function body)는 (…)와 같이 소괄호로 시작하여 소괄호로 종료된다. ( X )
5. 함수몸체에서 변수선언 문장은 어디에든 사용할 수 있다. ( X )
6. 하나의 응용 프로그램은 하나의 main() 함수와 여러 개의 다른 함수로 구성되며 필요에 따라 여러 소스 파일로 나누어 프로그래밍할 수 있다. ( O )
7. return 문장은 함수에서 반환값을 전달하는 목적과 함께 함수의 작업 종료를 알리는 문장이다. ( O )
8. 함수원형 구문에서 매개변수의 변수이름은 생략할 수 있다. ( O )
9. 특정한 나열 순서나 규칙을 가지지 않는 연속적인 임의의 수를 난수(random number)라 한다. ( O )
10. 함수 rand()의 함수원형은 헤더파일 stdarg.h에 정의되어 있다. ( X )
11. 1에서 n까지의 난수를 발생시키려면 함수 rand()를 이용하여 수식 rand() % n을 이용한다. ( X )
12. 다음에서 비어있는 부분을 적당히 채우시오.
13. 프로그램에서 ( 함수 )는 원하는 특정한 작업을 수행하도록 설계된 독립된 프로그램 단위이다.
14. 함수는 ( 라이브러리 )와 사용자 정의 함수로 구분할 수 있다.
15. 함수 중심의 프로그래밍 방식을 ( 절차적 프로그래밍 ) 방식이라 한다.
16. ( 함수 원형 )은 함수선언으로 변수선언과 같이 함수를 호출하기 전에 반드시 선언되어야 한다
17. 함수구현에서 자기 자신 함수의 호출을 이용하는 함수를 ( 재귀함수 )라 한다.
18. 함수 rand()의 함수원형은 헤더파일 ( <stdlib.h> )에 정의되어 있다.
19. 함수정의에서 매개변수는 필요한 경우 자료형과 변수명의 목록으로 나타내며 필요 없으면 키워드 ( void )를 기술한다.
20. 함수 ( time(NULL) )은 1970년 1월 1일 이후 현재까지 경과된 시간을 초 단위로 반환하는 함수이다.
21. 재귀방법으로 구현한 함수는 대부분 ( 반복 구문 )으로 구현이 가능하다.
22. 다음 함수의 함수 원형을 기술하시오.
23. 함수 valid()는 첫 번째 매개변수는 int이고 두 번째 매개변수는 double이며 반환 값은 없음

void valid(int, double);

1. 함수 absolute()는 실수 하나를 받아들여 절대값을 반환하는 함수

double absolute(double);

1. 실수의 인자 x와 y 2개를 입력 받아 x의 y 번 제곱을 구하는 jegop() 함수, 반환값 없음

void jegop(double x, double y);

1. 형식매개변수로 받은 한 개의 문자에 대한 코드값을 반환하는 code() 함수

int code(char ch);

1. 1에서 100까지 중에서 소수를 출력하는 함수(display)로, 인자도 없고 반환 값도 없음

void display();

1. 다음 함수 원형에서 문법 오류를 찾아 수정 하시오.
2. int test(void, void);

int test(void);

1. int max(double x, y);

int max(double x, double y);

1. int min(int a; int b);

int min(int a, int b);

1. void sort((double) a, (double) b);

void sort(double a, double b);

1. double power(x, y);

double power(double x, double y)

1. 다음 함수정의에서 문법 오류를 찾아 수정 하시오.
2. int area(int r)

{

return (int)(2 \* 3.14 \* r);

}

1. double sum3(double a, double b, double c)

{

return a + b + c;

}

1. void print()

{

printf("C 언어의출력test 입니다.");

}

1. int mult(float a, b)

(

return (int)(a\*b);

)

1. int multifly(int x, int y)

{

return x \* y;

}

1. int fact(int a)

{

return a \* fact(a-1);

}

1. int fun(char c);

{

return (int)('a' + c);

}

1. double square(double number)

{

return (number\*number);

}

1. double sum(int a, double b)

{

return (a+b);

}

* **프로그래밍 과제**

1. 두 개의 임의 정수를 m, n을 입력 받아 다음 함수를 작성하여 mn의 결과를 출력하는 프로그램을 작성하시오.

함수 intpow(int m, int n)의 결과는 mn

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  int pow(int pm, int pn);  int main() {  int m, n;  printf("밑 입력: ");  scanf\_s("%d", &m);  printf("지수 입력: ");  scanf\_s(" %d", &n);  printf("계산한 결과값: %d\n", pow(m, n));  return 0;  }  int pow(int pm, int pn) {  int i, result = 1;  for (i = 1; i <= pn; i++) {  result \*= pm;  }  return result;  } |
| **[실행 결과]** |

1. 다음을 참고로 임의의 수의 제곱을 구하는 함수 square()와 세제곱을 구하는 함수 cube()를 구현하여 임의의 수를 입력 받아 다섯 제곱을 구하는 프로그램을 작성하시오.

* cube() 구현 시 square()를 함수 호출하여 이용

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  int square(num);  int cube(num);  int main() {  int num, five;  printf("임의의 수 입력: ");  scanf\_s("%d", &num);  five = square(num) \* cube(num);  printf("%d의 다섯제곱: %d\n", num, five);  return 0;  }  int square(int n) {  int i, result = 1;  for (i = 1; i <= 2; i++) {  result \*= n;  }  return result;  }  int cube(int m) {  return square(m) \* m;  } |
| **[실행 결과]** |

1. 임의의 양의 정수 n 이 소수인지를 확인할 수 있는 함수를 구현하여, 임의의 수를 입력 받아 소수임을 확인하는 프로그램을 작성하시오

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  void primeNumber(int pn);  int main() {  int n;  printf("임의의 양의 정수 입력: ");  scanf\_s("%d", &n);  primeNumber(n);  return 0;  }  void primeNumber(int pn) {  int i;  if (pn <= 1) // 음수 or 1  printf("소수가 아니다.");  else if (pn == 2) // 2는 소수  printf("소수이다.");  else {    for (i = 2; i < pn; i++) {  if (pn % i == 0) { // 2부터 pn-1까지 나누어 떨어지면  printf("소수가 아니다.");  break;  }  else { // 1과 자기 자신만 나누어 떨어지면  printf("소수이다.");  break;  }  }  }  } |
| **[실행 결과]** |

1. 다음 식을 참고로 직각 삼각형에서 양변의 길이 a, b가 주어졌을 때 하나의 사선 길이 c를 구하는 함수를 만들고, 표준입력으로 받은 두 변의 길이 a, b를 이용하여 하나의 사선 길이 c를 구하는 프로그램을 작성하시오. 단, 사선의 길이를 계산하여 반환한다

* a2 + b2 = c2

c

a

b

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  #include <math.h>  double diagonal(double pa, double pb);  int main() {  double a, b;  printf("a = ");  scanf\_s("%lf", &a);  printf("b = ");  scanf\_s(" %lf", &b);  printf("사선의 길이 = %f", diagonal(a, b));  return 0;  }  double diagonal(double pa, double pb) {  double aa = 1, bb = 1;  int i;  for (i = 1; i <= 2; i++) {  aa \*= pa;  bb \*= pb;  }  return aa + bb;  } |
| **[실행 결과]** |

1. 함수 absolute()는 실수 하나를 받아들여 절대값을 반환하는 함수를 만들고 입력 받은 실수에 대하여 절대값을 출력하는 프로그램을 작성하시오

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  double absolute(double pn);  int main() {  double n;  printf("실수 입력: ");  scanf\_s("%lf", &n);  printf("절대값 = %f\n", absolute(n));  return 0;  }  double absolute(double pn) {  double ab = 0;  if (pn < 0) // 음수일 때  ab = -pn;  else if (pn > 0) // 양수일 때  ab = pn;  else // 0일 때  ab = 0.0;  return ab;  } |
| **[실행 결과]** |

1. 1부터 n까지의 합을 구하는 함수를 재귀 함수로 작성하여, 20을 실인자로 구현한 재귀 함수를 한번 호출하여 1부터 20까지 각각의 합이 출력되도록 프로그램을 작성하시오.

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  int jsum(int pn);  int main() {  int n = 20;  printf("== 1부터 %d까지의 각각의 합 ==\n", n);  jsum(n);  return 0;  }  int jsum(int pn) {  int sum = 0;  if (pn > 0) {  sum = pn + jsum(pn - 1);  printf("1부터 %2d까지 합 = %4d\n", pn, sum);  return sum;  }  else  return 0;  } |
| **[실행 결과]** |

1. 로또 번호를 생성하여 출력하는 프로그램을 작성하시오.

* 함수 random() 다음과 같은 조건으로 작성하여 처리

조건1) 난수는 1~n 까지 이며 6개를 생성하여 출력한다

조건2) n에 해당하는 값은 main() 함수에서 입력 받아 random() 함수 형식매개변수로 전달한다

조건3) random()함수 반환값은 없다

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  #include <stdlib.h>  #include <time.h>  void random(int pn);  int main() {  int n;  printf("난수의 MAX = ");  scanf\_s("%d", &n);  printf("\t== 6개 번호 ==\n");  random(n);  return 0;  }  void random(int pn) {  int i, num;  srand((long)time(NULL));  for (i = 0; i < 6; i++) {  num = rand() % pn + 1;  printf("%4d ", num);  }  puts("");  } |
| **[실행 결과]** |