|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **C2\_실습과제\_2주** | **이름 :** | **이갑성** | **학번 :** | **20135151** |

* **강의내용**

1. **난수 생성**

#include <stdlib.h> //rand(), srand()를 위한 헤더 파일 포함

#include <time.h> //time()을 위한 헤더 파일 포함

#define MAX 100

int main(d) {

int i, number;

srand((long) time(NULL));

printf("1 ~ %5d 사이의 난수 5개 :\n", MAX);

for (i = 0; i < 5; i++) {

number = rand()%MAX + 1;

printf("%5d ", number);

}

}

**//실수형 난수생성 : 0 <= rand()/RAND\_MAX <= 1**

|  |
| --- |
| [실행결과] |

1. **재귀함수**

int factorial(int);

int main(){

int i;

for ( i = 1; i <= 10; i++ )

printf( "%2d! = %d\n", i, factorial( i ) );

}

int factorial(int number) {

if (number <= 1) return 1;

else return (number \* factorial(number - 1));

}

|  |
| --- |
| [실행결과] |

1. 시간지연

#include <time.h>

#include <stdio.h>

#include <windows.h>

int main(void){

int i;

printf("start!\n");

for(i=0;i<5;i++)

{

Sleep(2000); //2초 후에 \*출력

printf("%c", '\*');

}

printf("\nend\n");

return 0;

}

|  |
| --- |
| [실행결과] |

1. 커서 위치 제어

#include <stdio.h>

#include <windows.h>

void gotoxy(int x, int y);

int main(void){

gotoxy(2,4); printf("Hello");

gotoxy(40, 20); printf("Hello\n");

return 0;

}

void gotoxy(int x, int y){

COORD Pos = {x - 1, y - 1};

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), Pos);

}

|  |
| --- |
| [실행결과] |

* 프로그램 과제

1. **다음을 참고로 피보나츠 수를 구하는 함수를 재귀함수로 작성**하여 처음부터 입력 받은 수 번째의 피보나츠 수에 대하여 각각의 함수 결과를 모두 출력하는 프로그램 작성.

- 피보나츠 수: Fn = Fn-1 + Fn-2, F1 = 1, F0 = 0;

- int fibo(int);

#include <stdio.h>

int fibo(int);

int main(void){

int count, input;

printf("n 번째의 피보나츠 수 (n입력) : ");

scanf("%d", &input);

printf("피보나츠수를 처음부터 %d번째 까지출력.\n\n", input);

for (count = 0; count <= input; count++)

printf("%5d번째 피보나츠수 = %6d\n", count, fibo(count));

}

int fibo(int count)

{

int fn;

…………

return fn;

}

|  |
| --- |
| [프로그램소스]  #include <stdio.h>  int fibo(int);  int main(void) {  int count, input;  printf("n 번째의 피보나츠 수 (n입력) : ");  scanf("%d", &input);  printf("피보나츠수를 처음부터 %d번째 까지출력.\n\n", input);  for (count = 0; count <= input; count++)  printf("%5d번째 피보나츠수 = %6d\n", count, fibo(count));  }  int fibo(int count)  {  int fn;    if (count == 0)  return 0;  else if (count == 1)  return 1;  else  {  fn = fibo(count - 1) + fibo(count - 2);  }    return fn;  } |
| [실행결과] |

1. 3초간격으로 문자를 화면에 출력하는 프로그램을 다음과 같이 작성하시오.

* int coord(char ) : 매개변수가 ‘x’이면 1~80 사이의 난수를 생성하여 반환하고 ‘y’이면 1~40사이의 난수를 생성하여 반환
* void write(int ) : 매개변수로 받은 횟수만큼 문자를 화면에 출력. 단, 출력위치는 gotoxy() 함수를 사용하며 좌표값은 coord()함수를 사용한다. 난수를 사용하여 매번 다른 문자를 출력하au 한 문자 출력 후 화면을 clear하도록 한다.

int main(void){

//출력회수 입력 – 본인작성

//write() 함수를 호출하여 문자 출력 – 본인 작성

return 0;

}

// void write(int) 함수 정의 – 본인작성

// int coord(char) 함수 정의 – 본인작성

void gotoxy(int x, int y){

COORD Pos = {x - 1, y - 1};

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), Pos);

}

|  |
| --- |
| [프로그램소스]  #include <stdio.h>  #include <stdlib.h>  #include <time.h>  #include <windows.h>  void write(int);  int coord(char);  void gotoxy(int, int);s  int main(void) {  int num;  srand((long)time(NULL));  printf("출력횟수를 입력하세요: ");  scanf("%d", &num);  write(num);  return 0;  }  void gotoxy(int x, int y) {  COORD Pos = { x - 1, y - 1 };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), Pos);  }  void write(int num)  {  char ch;  char selectCh;  int xPosition;  int yPosition;  int i;  for (i = 0; i < num; i++)  {    ch = rand() % 26 + 97;      xPosition = coord('x');  yPosition = coord('y');    gotoxy(xPosition, yPosition);  printf("%c", ch);  Sleep(3000);    gotoxy(xPosition, yPosition);    printf(" ");  }  }  int coord(char select)  {  int pos;    if (select == 'x')  pos = rand() % 80 + 1;  else if (select == 'y')  pos = rand() % 40 + 1;    return pos;  } |
| [실행결과] |

1. 하나의 문자를 정해진 방향으로 움직이며 출력하는 프로그램을 작성하시오. 단, 선택된 메뉴는 함수를 정의하여 처리하도록 한다. 반환값은 없으며 출력하고자 하는 문자를 입력 받아 형식매개변수로 전달한다.

* 메뉴항목
* 1번 : 수평방향으로 출력
* 2번 : 수직방향으로 출력
* 3번 : 대각선방향으로 출력

힌트) 움직이며 출력하기

for(i=1;i<40;i+=2){

gotoxy(i,5);

printf("%c”, ‘@’);

Sleep(150);

gotoxy(i,5);

printf(" "); // 대상지우기

}

|  |
| --- |
| [프로그램소스]  #include <stdio.h>  #include <windows.h>  void gotoxy(int, int);  void garo(char);  void sero(char);  void cross(char);  void main()  {  int num;  int ch;  printf("=======메뉴======= \n");  printf("1번: 수평방향으로 출력 \n");  printf("2번: 수직방향으로 출력 \n");  printf("3번: 대각선방향으로 출력 \n");    printf("메뉴 입력: ");  scanf("%d", &num);  if (num == 1)  {  printf("수평방향으로 출력을 선택하셨습니다. \n 문자를 입력하세요: ");  ch = getche();  printf("\n");  Sleep(2000);  system("cls");  garo(ch);  }  else if (num == 2)  {  printf("수직방향으로 출력을 선택하셨습니다. \n 문자를 입력하세요: ");  ch = getche();  printf("\n");  Sleep(2000);  system("cls");  sero(ch);  }  else if (num == 3)  {  printf("대각선방향으로 출력을 선택하셨습니다. \n 문자를 입력하세요: ");  ch = getche();  printf("\n");    Sleep(2000);  system("cls");  cross(ch);    }  }  void garo(char ch)  {  int i;    for (i = 1; i < 40; i++)  {  gotoxy(i, 10);  printf("%c", ch);  Sleep(1000);  gotoxy(i, 10);  printf(" ");  }  }  void sero(char ch)  {  int i;  for (i = 1; i < 40; i++)  {  gotoxy(10, i);  printf("%c", ch);  Sleep(1000);  gotoxy(10, i);  printf(" ");  }  }  void cross(char ch)  {  int i;  for (i = 1; i < 40; i++)  {  gotoxy(i, i);  printf("%c", ch);  Sleep(1000);  gotoxy(i, i);  printf(" ");  }  }  void gotoxy(int x, int y) {  COORD Pos = { x - 1, y - 1 };  SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), Pos);  } |
| [실행결과] |

1. 화면에 출력된 두 개의 정수데이터에 대한 사칙연산 답을 입력 받아 맞춘 횟수를 출력하는 프로그램을 작성하시오. 단, 10회 반복하도록 한다

출력예시] 40 + 38

정답은? 78

결과 : 1번 맞았습니다.

* 조건 1: 출력되는 정수데이터는 10~99. 난수 사용
* 조건 2: 2초 후 화면에 출력된 데이터 삭제
* 조건 3: 사칙연산은 새 데이터를 출력할 때 마다 다르게 할 것
* 조건 4: 연산은 다음과 같은 함수를 정의하여 처리하도록 한다. 매개변수로 받은 데이터 연산결과 반환

int add(int , int ) //+연산

int sub(int, int) //-연산

int mul(int, int) //\* 연산

int div(int, int) // / 연산

|  |
| --- |
| [프로그램소스]  #include <stdio.h>  #include <stdlib.h>  #include <time.h>  #include <windows.h>  int add(int, int);  int sub(int, int);  int mul(int, int);  int nanugi(int, int);  void main()  {  int result;  int answer = 0;  int cnt = 0;  int n1, n2;  int cal;  int i;  srand((long)time(NULL));  printf("게임을 시작합니다.\n");  system("cls");  for (i = 0; i < 10; i++)  {  n1 = rand() % 90 + 10;  n2 = rand() % 90 + 10;  cal = rand() % 4 + 1;  if (cal == 1)  {  printf("%d + %d = ? \n", n1, n2);  result = add(n1, n2);  Sleep(3000);  system("cls");  }  else if (cal == 2)  {  printf("%d - %d = ? \n", n1, n2);  result = sub(n1, n2);  Sleep(3000);  system("cls");  }  else if (cal == 3)  {  printf("%d \* %d = ? \n", n1, n2);  result = mul(n1, n2);  Sleep(3000);  system("cls");  }  else if (cal == 4)  {  printf("%d / %d = ? \n", n1, n2);  result = nanugi(n1, n2);  Sleep(3000);  system("cls");  }  printf("정답은? \n");  scanf("%d", &answer);  if (answer == result)  {  cnt++;  printf("정답! \n%d 번 맞았습니다. \n", cnt);  Sleep(3000);  system("cls");  }  else  {  printf("틀렸습니다.\n");  Sleep(3000);  system("cls");  }  }  }  int add(int a, int b)  {  return a + b;  }  int sub(int a, int b)  {  return a - b;  }  int mul(int a, int b)  {  return a \* b;  }  int nanugi(int a, int b)  {  return a / b;  } |
| [실행결과] |