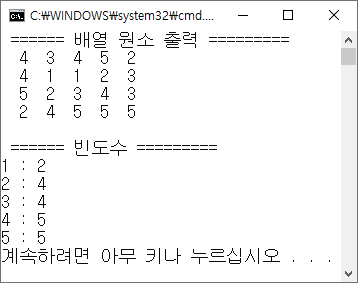
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 020\_1\_C\_7주 | 학번 : | 20175105 | 이름 : | 곽영주 |

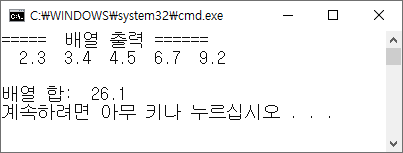
* **강의 내용**

1. 배열 활용 - 빈도수 계산 & 난수로 초기화



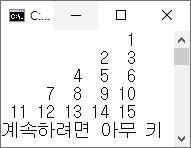
|  |
| --- |
| for (i = 0; i < SIZE; i++) {  r = rand() % 5 + 1;  ary[i] = r; // 배열에 난수 저장  ++count[r - 1]; // 빈도수  }  printf("===== 배열 원소 출력 =========\n");  for (i = 0; i < SIZE; i++) { // 배열 출력  printf("%-3d", ary[i]);  if ((i + 1) % 5 == 0)  puts("");  }  puts("");  printf("====== 빈도수 =========\n");  for (i = 0; i < 5; i++) { // 빈도수 출력  printf("%d : %d\n", i + 1, count[i]);  } |

1. 함수에서 일차원 배열 전달

****

|  |
| --- |
| int size = sizeof(data) / sizeof(data[0]);  printf("===== 배열 출력 ======\n");  arrayPrint(data, size);  printf("배열의 합: %3.1f", sum(data, size));  double sum(double d[], int size) {  int i;  double hap = 0;  for (i = 0; i < size; i++) {  hap += d[i];  }  return hap;  }  void arrayPrint(double d[], int size) {  int i;    for (i = 0; i < size; i++) {  printf("%5.1f", d[i]);  }  puts("");  } |

1. **함수에서 이차원 배열 전달 – 직각 삼각형**

****

|  |
| --- |
| initDim(dim);  write(dim);  void initDim(int dim[][col]) {  int i, j, k = 1;  for (i = 0; i < row; i++) {  for (j = col - (i + 1); j < col; j++) {  dim[i][j] = k++;  }  }  }  void write(int dim[][col]) {  int i, j;  for (i = 0; i < row; i++) {  for (j = 0; j < col; j++) {  dim[i][j] == 0 ? printf("%3c",' ') : printf("%3d",dim[i][j]);  }  puts("");  }  } |

* **내용 점검**

1. 비어있는 부분을 채워 넣으시오.
2. 이차원 배열은 테이블 형태의 구조를 나타낼 수 있으므로 ( 행 ) 과 ( 열 )의 구조로 표현할 수 있다.
3. 이차원 배열선언 int num[3][4];으로 할당되는 배열원소 수는 ( 12 ) 개 이다.
4. 문자형 배열에서 배열 크기가 초기값 원소 수보다 크면 지정하지 않은 원소의 초기 값은 자동으로 문자형은( ‘\0’ )인 널 문자가 자동으로 채워진다.
5. 연산자 sizeof를 이용한 식 ( sizeof(배열이름) / sizeof(배열원소) )의 결과는 배열크기이다.
6. 이차원 배열 선언 초기값 지정에서 첫 번째 대괄호 내부의 ( 행의 크기 )는 명시하지 않을 수 있다.
7. 이차원 배열 int x[2][3]의 행의 수는 연산자 sizeof를 이용한 식 ( sizeof(x) / sizeof(x[0]) )로 계산할 수 있다.
8. 다음 설명에 대한 배열 선언 문장을 작성하시오.
9. 크기가 12인 int형 배열 digit

int digit[12];

1. 배열 month를 선언하면서 초기 값으로 각 월의 날수를 저장, 단 2월은 28로 저장

int month[] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

1. 3행 4열의 double형 이차원 배열 twoF

double twoF[3][4];

1. 2행 3열의 int형 이차원 배열 twoD을 선언하면서 초기 값으로 13, 56, 69, 36, 72, 6을 저장

int twoD[2][3] = {13, 56, 69, 36, 72, 6};

1. 5개의 int형 배열 dim을 선언하면서 초기 값을 모두 0으로 저장

int dim[5] = 0;

1. double 형 배열 real을 선언하면서 값 2.13, 5.71, 2.87, 7.89를 초기화하는 문장

double real[] = {2.13, 5.71, 2.87, 7.89};

1. 1차원 int 배열 ary를 원소 개수 12로 선언하는 문장과 세 번째 원소에 10을 저장하는 문장

int ary[12];

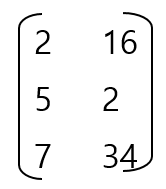
ary[2] = 10;

1. int 형 이차원 배열 two[3][4]을 선언하는 문장과 메모리 순서로 6번째에 20을 저장하는 문장

int two[3][4];

int two[2][5] = 20;

1. int 형 이차원 배열 matrixB에서 다음을 행렬을 초기화하는 문장

 int matrix[2][3] = { {2, 16}, {5, 2}, {7, 34} };

1. 다음 문장에서 문법 오류 및 실행 오류를 찾아 수정하시오.
2. int temperature[];

int temperature[10];

1. int input(3);

int input[3];

1. int[3] data;

int data[3];

1. double values[3,4];

double values[3][4];

1. double[3] eval[4]; //2차원 배열 선언

double eval[3][4];

1. double price[3][4] = {{1.1, 2.3, 3.4, 5.6, 2.1}};

double price[3][4] = {1.1, 2.3, 3.4, 5.6, 2.1};

1. double time[2][] = {{6.1, 7.2}, {3.4, 5.6}, {3.9}};

double time[][2] = {{6.1, 7.2}, {3.4, 5.6}, {3.9}};

* **프로그래밍 과제**

1. 다음과 같이 일차원 배열의 동등함을 검사하는 함수를 작성하여 결과를 알아보는 프로그램을 작성하시오.

* int isequalarray(int a[], int b[], int n /\* 배열 원소 수 \*/)
* 배열 a와 b의 배열 크기가 모두 n이며 순차적으로 원소 값이 모두 같으면 1을 반환, 아니면 0을 반환하는 함수

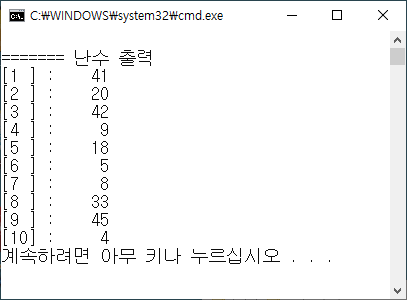
int a[] = { 4, 7, 9, 3, 6 };

int b[] = { 4, 8, 2, 3, 6 };

int c[] = { 4, 7, 9, 3, 6 };

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  int isqualarray(int a[], int b[], int n);  void arrayPrint(int arr[]);  int main() {  int a[] = { 4, 7, 9, 3, 6 };  int b[] = { 4, 8, 2, 3, 6 };  int c[] = { 4, 7, 9, 3, 6 };  int n = sizeof(a) / sizeof(a[0]);  printf("배열 출력\n");  printf("a = ");  arrayPrint(a,n);  printf("b = ");  arrayPrint(b,n);  printf("c = ");  arrayPrint(c,n);  printf("=== 두 개의 배열이 같으면 1 다르면 0 ===\n");  printf("a와 b 비교 : %d\n", isqualarray(a, b, n));  printf("a와 c 비교 : %d\n", isqualarray(a, c, n));  printf("b와 c 비교 : %d\n", isqualarray(b, c, n));  return 0;  }  int isqualarray(int a[], int b[], int n) {  int i, k = 0;  for (i = 0; i < n; i++) {  if (a[i] != b[i]) { // 배열 원소가 다르면  k = 0;  break;  }  else // 배열 원소가 같으면  k = 1;  }  return k;  }  void arrayPrint(int arr[], int n) {  int i;  for (i = 0; i < n; i++) {  printf("%-3d", arr[i]);  }  puts("");  } |
| **[실행 결과]** |

1. 1부터 50사이의 난수 10개를 생성하여 일차원 배열에 저장한다. 단 배열에 동일한 값은 중복저장 할 수 없다.



int main(void) {

int rnd[size], i, j;

srand((long)time(NULL)); //시드값 배정

initRnd(rnd); //1~50사이의 난수로 배열 초기화

printf("\n%-18s\n", "======= 난수 출력");

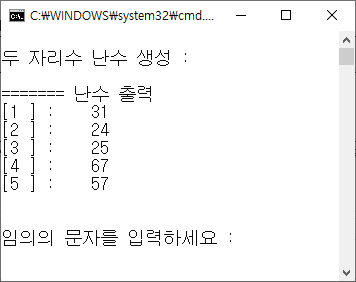
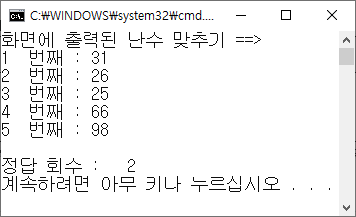
print(rnd); //배열 원소 출력

return 0;

}

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  #include <stdlib.h>  #include <time.h>  # define size 10  void initRnd(int arr[]);  void print(int arr[]);  int main() {  int rnd[size];  srand((long)time(NULL));  initRnd(rnd);  printf("\n % -18s\n", "======= 난수 출력");  print(rnd);  return 0;  }  void initRnd(int arr[]) {  int i, j, r;  for (i = 0; i < size; i++) {  arr[i] = rand() % 50 + 1; // 난수  for (j = 0; j < i; j++) { // 중복 검사  if (arr[i] == arr[j]) {  i--;  break;  }  }  }  }  void print(int arr[]) {  int i;  for (i = 0; i < size; i++) {  printf("[%-2d] : %5d\n", i + 1, arr[i]);  }  } |
| **[실행 결과]** |

1. 배열 pass에는 2자리수의 정수 난수를 5개 생성하여 배열에 저장한 후 다음과 같이 실행되는 프로그램을 작성하시오.

int main(void) {

int tot;

int weight[size];

srand((long)time(NULL)); //시드값 배정

printf("\n두 자리수 난수 생성 :\n");

init(weight);

printf("\n%-18s\n", "======= 난수 출력");

print(weight);

system("pause");

system("cls");

printf("화면에 출력된 난수 맞추기 ==>\n");

tot = count(weight);

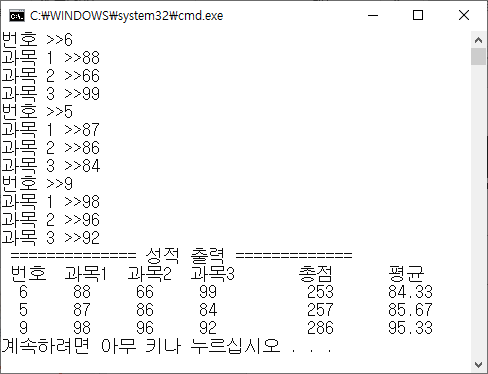
printf("\n정답 회수 : %3d\n", tot);

return 0;

}

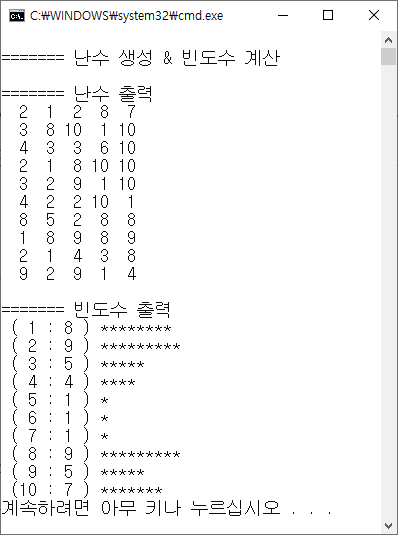
|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  #include <stdlib.h>  #include <time.h>  #define size 5  void init(int arr[]);  void print(int arr[]);  int count(int arr[]);  int main() {  int tot;  int weight[size];  srand((long)time(NULL));  printf("\n두 자리수 난수 생성 : \n");  init(weight); // 초기화  printf("\n%-18s\n", "======= 난수 출력");  print(weight); // 배열 출력  system("pause");  system("cls");  printf("화면에 출력된 난수 맞추기 ==>\n");  tot = count(weight); // 정답 횟수  printf("\n정답 회수 : %3d\n", tot);  return 0;  }  void init(int arr[]) {  int i;  for (i = 0; i < size; i++) {  arr[i] = rand() % 90 + 10; // 배열에 난수 저장  }  }  void print(int arr[]) {  int i;  for (i = 0; i < size; i++) {  printf("[%-2d] : %5d\n", i + 1, arr[i]);  }  }  int count(int arr[]) {  int i, n = 0, cnt = 0;  for (i = 0; i < size; i++) {  printf("%-2d 번째 : ", i + 1);  scanf\_s(" %d", &n); // 표준 입력  if (arr[i] == n) // 정답이 맞으면 count  cnt++;  }  return cnt;  } |
| **[실행 결과]** |

1. 3명의 성적을 다음과 같이 처리하여 출력하는 프로그램을 작성하시오. – 이차원 배열 사용



|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  void print(int arr[][4], int rsize, int csize);  void hapANDavg(int arr[][4], int rsize, int csize);  int main() {  int score[][4] = { {6, 88, 66, 99}, {5, 87, 86, 84}, {9, 98, 96, 92} };    int rsize = sizeof(score) / sizeof(score[0]);  int csize = sizeof(score[0]) / sizeof(score[0][0]);  print(score, rsize, csize);  printf("============ 성적 출력 ============\n");  printf(" 번호\t과목1\t과목2\t과목3\t\t총점\t평균\n");  hapANDavg(score, rsize, csize);  }  void print(int arr[][4], int rsize, int csize) {  int i, j;  for (i = 0; i < rsize; i++) {  for (j = 0; j < csize; j++) {  if (j == 0)  printf("번호 >> %d\n", arr[i][j]); // 번호 출력  else  printf("과목 %d >> %d\n", j, arr[i][j]); // 과목 점수 출력  }  puts("");  }  }  void hapANDavg(int arr[][4], int rsize, int csize) {  int i, j;  int hap ;  double avg;  for (i = 0; i < rsize; i++) {  hap = 0;  avg = 0;  for (j = 0; j < csize; j++) {  if (j == 0)  printf("%3d\t", arr[i][j]); // 번호  else {  printf("%3d\t", arr[i][j]); // 과목 점수  hap += arr[i][j]; // 합  }  }  avg = (double)hap / (csize - 1); // 평균  printf("\t%3d\t%5.2f\n", hap, avg);  }  } |
| **[실행 결과]** |

1. 1~10 난수 50개를 생성시켜 빈도수를 계산한 후 다음과 같이 출력하는 프로그램을 작성하시오



#include <stdio.h>

#include <stdlib.h> //rand(), srand()

#include <time.h> //time()

void init(int[], int, int[]);

void print(int[], int);

void graph(int[], int);

int main(void) {

int ran[50] = { 0 }; //난수 저장

int count[10] = { 0 }; //빈도수 저장

srand((long)time(NULL)); //시드값 배정

printf("\n%-18s\n", "======= 난수 생성 & 빈도수 계산");

init(ran, sizeof(ran)/sizeof(int), count);

printf("\n%-18s\n", "======= 난수 출력");

print(ran, sizeof(ran) / sizeof(int));

printf("\n%-18s\n", "======= 빈도수 출력");

graph(count, sizeof(count) / sizeof(int));

return 0;

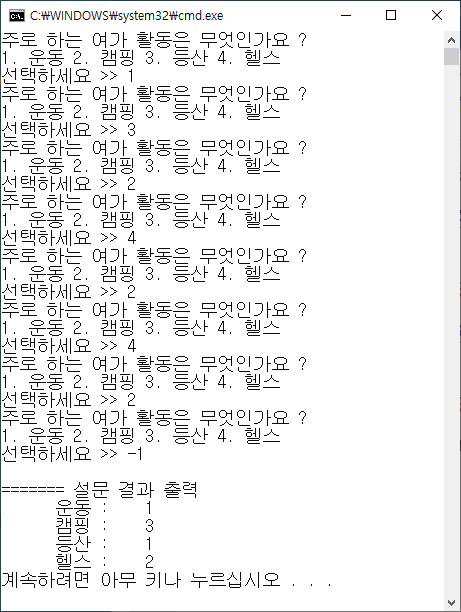
}

|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  #include <stdlib.h>  #include <time.h>  void init(int[], int, int[]);  void print(int[], int);  void graph(int[], int);  int main() {  int ran[50] = { 0 };  int count[10] = { 0 };  srand((long)time(NULL));  printf("\n%-18s\n", "======= 난수 생성 & 빈도수 계산");  init(ran, sizeof(ran) / sizeof(int), count);  printf("\n%-18s\n", "======= 난수 출력");  print(ran, sizeof(ran) / sizeof(int));  printf("\n%-18s\n", "======= 빈도수 출력");  graph(count, sizeof(count) / sizeof(int));  return 0;  }  void init(int arr[], int size, int cnt[]) {  int i, r;  for (i = 0; i < size; i++) {  r = rand() % 10 + 1;  arr[i] = r; // 배열에 난수 저장  ++cnt[r - 1]; // 빈도수 계산  }  }  void print(int arr[], int size) {  int i;  for (i = 0; i < size; i++) {  printf("%3d", arr[i]);  if ((i + 1) % 5 == 0) // 줄바꿈  puts("");  }  }  void graph(int cnt[], int size) {  int i, j;  for (i = 0; i < size; i++) {  printf("( %2d :%2d )", i + 1, cnt[i]); // 빈도수 출력  for (j = 0; j < cnt[i]; j++) { //별찍기  printf(" %c", '\*');  }  puts("");  }  } |
| **[실행 결과]** |

1. 다음과 같이 여과 활동 설문 결과를 출력하는 프로그램을 작성하시오.

int question(); //질문에 대한 답을 반환하는 함수

void print(int[], int); //설문 결과를 저장하고 있는 배열을 매개변수로 받아 제시된 결과처럼 출력하는 함수



|  |
| --- |
| **[프로그램 소스]**  #include <stdio.h>  int question();  void print(int[], int);  int main() {  int count[4] = { 0 };  int num = 0;  while(1){  printf("주로 하는 여가 활동은 무엇인가요?\n");  num = question();  if (num == -1) // 반복문 종료  break;  ++count[num - 1]; // 설문 결과 계산  }  printf("====== 설문 결과 출력\n");  print(count, sizeof(count) / sizeof(int));  }  int question() {  int num = 0;  printf("1. 운동 2. 캠핑 3. 등산 4. 헬스\n");  printf("선택하세요 >> ");  scanf\_s("%d", &num);  return num;  }  void print(int cnt[], int size) {  int i;  for (i = 0; i < size; i++) {  switch (i) { // 각각 설문 결과  case 0 :  printf("\t운동 : %5d\n", cnt[i]);  break;  case 1 :  printf("\t캠핑 : %5d\n", cnt[i]);  break;  case 2 :  printf("\t등산 : %5d\n", cnt[i]);  break;  case 3 :  printf("\t헬스 : %5d\n", cnt[i]);  break;  }  }  } |
| **[실행 결과]** |