REPORT #1

|  |
| --- |
| **제 출 일** : 2017년 월 일 |
| **과 목 명** : 컴퓨터 프로그래밍 및 실습 |
| **담당교수 : 김정국 교수님** |
| **학 과** : 컴퓨터 전자 시스템 공학부 |
| **학 번** : 201702234 |
| **성 명 : 유동혁** |

**HANKUK UNIVERSITY OF FOREIGN STUDIES**

****

**1 . 포인터 연산 예제**

* 1. **문제 정의**

Int, char, double등 각 변수형에 따라 저장되는 포인터의 위치 변화가 다르다

**1.2. 문제 해결 방법**

char형은 1byte, int형은 4byte, double형은 8byte이므로 각 포인터에서 +1 했을 시 증가하는 값이 다를것이다.

**1.3. 소스 코드**

|  |
| --- |
| /\*  \* pointer.cpp  \*  \* Created on: 2017. 9. 11.  \* Author: Yoo DongHyuk  \*/  #include<iostream>  using namespace std;  int main()  {  char \*pc;  int \*pi;  double \*pd;  pc = (char \*)10000;  pi = (int \*)10000;  pd = (double \*)10000;  cout << "증가 전 pc = " << (void \*)pc << " pi = " << pi << " pd = " << pd << endl;  pc++;  pi++;  pd++;  cout << "증가 후 pc = " << (void \*)pc << " pi = " << pi << " pd = " << pd << endl;  return 0;  } |

**1.4 결과 화면**

|  |
| --- |
|  |

**2 . 사이즈가 10인 배열을 입력 받아 총합과 평균을 구함**

**2.1 문제 정의**

Cin을 이용해 사이즈가 10인 배열을 입력 받고, 각 값의 총합과 평균을 구한다

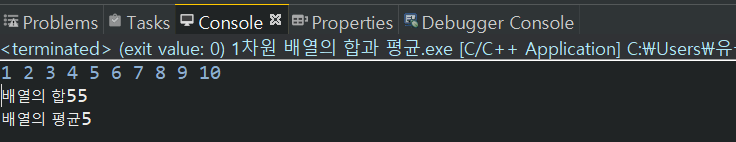
**2.2. 문제 해결 방법**

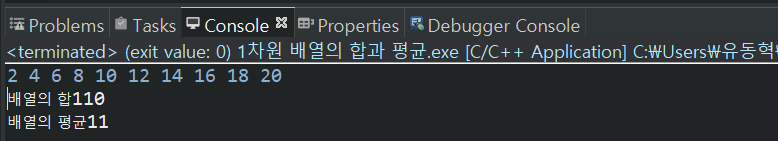
Cin을 이용해 사이즈가 10인 배열을 입력 받고, sum 변수에 for문을 이용해서 총 합을 저장한 후, sum값과 sum/10값을 출력한다.

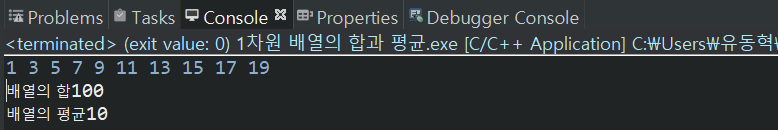
**2.3. 소스 코드**

|  |
| --- |
| /\*  \* 1chawon.cpp  \*  \* Created on: 2017. 9. 11.  \* Author: Yoo DongHyuk  \*/  #include<iostream>  using namespace std;  int main()  {  int hop[10] = {0};  int i, sum=0;  for(i=0; i<10; i++)  cin >> hop[i];//배열 값 입력  for(i=0; i<10; i++)  sum += hop[i];//배열 총 합  cout << "배열의 합" << sum << endl;  cout << "배열의 평균" << sum/10 << endl;  return 0;  } |

**2.4 결과 화면**







**3 . 3x3, 5x5, 7x7사이즈의 행렬들을 반시계방향으로 90도 회전한 행렬을 출력**

**3.1 문제 정의**

각 사이즈의 행렬들을 반시계방향으로 90도 회전한 행렬을 출력

**3.2. 문제 해결 방법**

0,0 0,1 0,2 0,2 1,2 2,2

1,0 1,1 1,2 🡺 0,1 1,1 2,1

2,0 2,1 2,2 0,0 1,0 2,0

반시계방향으로 90도 회전한 행렬은 위와 같으므로, 이중 for문을 이용하여 적용한다.

**3.3. 소스 코드**

/\*

\* hangryeol hoaejeon.cpp

\*

\* Created on: 2017. 9. 11.

\* Author: Yoo DongHyuk

\*/

#include<iostream>

using namespace std;

int main()

{

int three[3][3];//3x3사이즈 배열

int five[5][5];//5x5 사이즈 배열

int seven[7][7];//7x7 사이즈 배열

int i, x, y;

i=0;

for(x=0; x<3; x++)

{

for(y=0; y<3; y++)

{

three[x][y] = ++i;

cout << three[x][y] << " ";

}

cout << endl;

}//3x3 회전 전

for(y=2; y>=0; y--)

{

for(x=0; x<3; x++)

cout << three[x][y] << " ";

cout << endl;

}//3x3 회전 후

cout << "-----------" << endl;

i=0;

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

{

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

{

five[x][y] = ++i;

cout << five[x][y] << " ";

}

cout << endl;

}//5x5 회전 전

for(y=4; y>=0; y--)

{

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

cout << five[x][y] << " ";

cout << endl;

}//5x5 회전 후

cout << "-----------" << endl;

i=0;

for(x=0; x<7; x++)

{

for(y=0; y<7; y++)

{

seven[x][y] = ++i;

cout << seven[x][y] << " ";

}

cout << endl;

}//7x7 회전 전

for(y=6; y>=0; y--)

{

for(x=0; x<7; x++)

cout << seven[x][y] << " ";

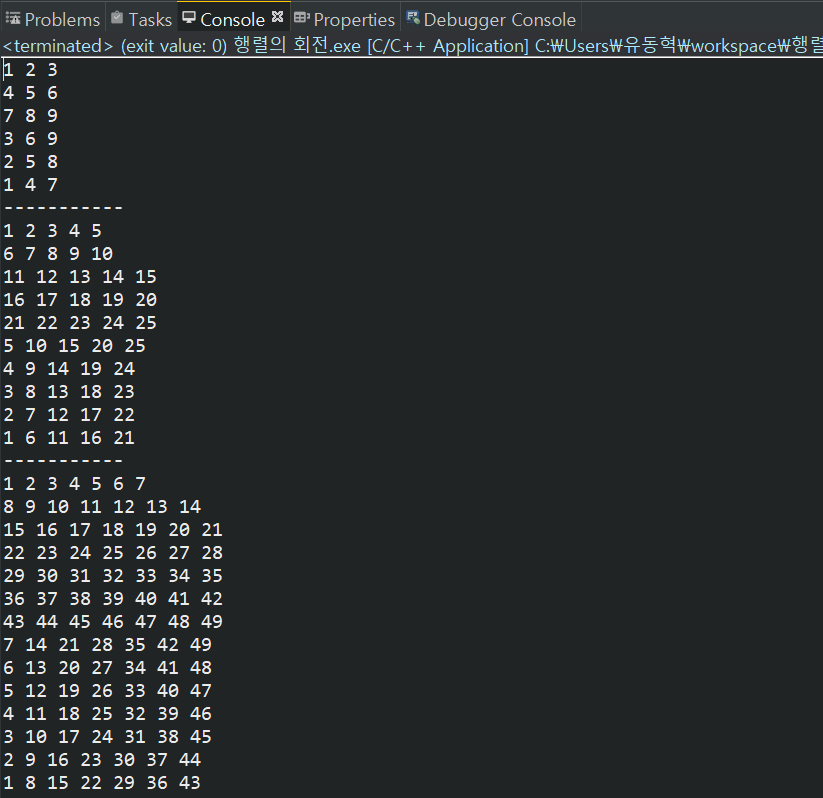
cout << endl;

}//7x7 회전 후

return 0;

}

**3.4 결과 화면**



**4 . a^b 계산하고 오름차순 정렬**

**4.1 문제 정의**

A^b를 계산해서 크기가 16인 배열에 저장한 후 중복되는 값은 정리하고 오름차순으로 정리한다.

**4.2. 문제 해결 방법**

for문으로 a^b를 계산해 배열에 일괄적으로 넣는다

배열에 있는 값들을 비교해 왼쪽 값 보다 오른쪽 값이 더 작으면 서로 자리를 바꿔준다.

자리를 바꾸는 도중 중복되는 값이 생기면 (배열의 숫자 수) – 1 (=a)을 해준 다음, 오른쪽에 있던 값을 int의 최대값으로 바꾼다 🡺정리가 끝났을 때 가장 오른쪽으로 가게 된다.

오름차순으로 정렬이 끝나면 a만큼 배열의 값들을 출력한다. 🡺이 때 int의 최대값으로 바꿔준 숫자들은 출력 되지 않는다.

**4.3. 소스 코드**

/\*

\* auibseoung.cpp

\*

\* Created on: 2017. 9. 11.

\* Author: Yoo DongHyuk

\*/

#include<iostream>

#include<limits.h>//int의 최대값을 이용하기 위해서

using namespace std;

int main()

{

int a, b, i=0, x, y, save, count;

int gyaesan[] = {1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1};

for(a=2; a<=5; a++)//a^b 계산, 저장

{

for(b=2; b<=5; b++)

{

for(x=1; x<=b; x++)

gyaesan[i] \*= a;

i++;//for문이 끝났을 때 i=16(배열은 0~15)

}

}

count = i;//count = 16

for(x=0; x<i; x++)//정렬

{

for(y=x+1; y<i; y++)

{

if(gyaesan[y] < gyaesan[x])//왼쪽값 보다 오른쪽 값이 더 작으면

{

save = gyaesan[x];

gyaesan[x] = gyaesan[y];

gyaesan[y] = save;//서로 자리를 바꿔준다

}

else if(gyaesan[y] == gyaesan[x])

{

gyaesan[y] = INT\_MAX;//중복값은 int의 최대값으로 바꿔서 가장 오른쪽으로 가게함

count--;//count에서 1을 뺌으로써 가장 오른쪽 값은 출력되지않게함

}

}

}

cout << "count = " << count << endl;

for(x=0; x<count; x++)//정렬 후 출력

cout << gyaesan[x] << " ";

return 0;

}

**4.4 결과 화면**

