

|  |
| --- |
| **R E P O R T** |

|  |  |
| --- | --- |
| 제 목 | 온도변환, 369족보, 별 도형, 숫자야구 |
| 과 목 | 컴퓨터 개론 및 실습 |
| 날짜(년/월/일) | 2017/05/11 |
| 소속 학과 | 컴퓨터 전자 시스템 공학부 |
| 학번 | 201702234 |
| 이름 | 유동혁 |

|  |  |
| --- | --- |
| 본 보고서의 내용 중 다른 문서(자료)를 인용한 것이 있습니까?  예 ( ) 아니오 ( V ) | |
| 위에서 ‘예’로 답한 경우, 인용한 다른 문서는 무엇인지 아래에 명시해 주세요.(여러 개의 경우 주요 자료 2개 까지) | |
| - 저자 1 : |  |
| * 제목 1 : |  |
| * 저자 2 : |  |
| * 제목 2 : |  |

1. **섭씨와 화씨를 변환시켜주는 프로그램**
   1. **Solution**

- 화씨 = 섭씨\*1.8 + 32

- 섭씨 = (화씨-32)/1.8

- 함수의 반환 값이 소수이므로 float형 사용

* 1. **Source code**

/\*

\* hwaci\_subci.c

\*

\* Created on: 2017. 5. 10.

\* Author: 유동혁

\*/

#include<stdio.h>

//화씨와 섭씨를 변환시켜주는 프로그램

float hwa\_to\_sub(float gion\_hwa);

float sub\_to\_hwa(float gion\_sub);

//함수 값 계산 후 반환 값이 소수이기 때문에 int형이 아닌 float형을 사용

int main()

{

float gion\_sub=28, result\_sub;

float gion\_hwa=80, result\_hwa;

result\_sub = sub\_to\_hwa(gion\_sub);

result\_hwa = hwa\_to\_sub(gion\_hwa);

//변환 결과는 각 함수의 반환 값과 같다.

printf("섭씨 %f도는 화씨 %f도와 같습니다.\n", gion\_sub, result\_sub);

printf("화씨 %f도는 섭씨 %f도와 같습니다.\n", gion\_hwa, result\_hwa);

return 0;

}

float sub\_to\_hwa(float gion\_sub)

{

return gion\_sub \* 1.8 + 32;

}

//섭씨를 화씨로 변환하는 함수 반환값은 화씨

float hwa\_to\_sub(float gion\_hwa)

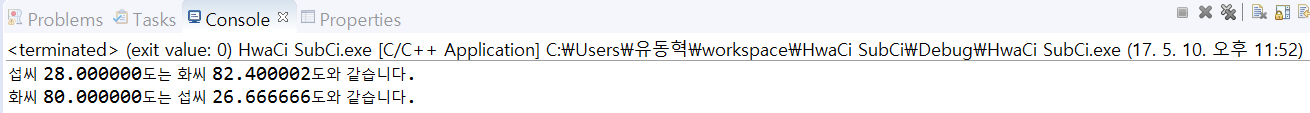
{

return (gion\_hwa - 32) / 1.8;

}

//화씨를 섭씨로 변환하는 함수 반환값은 섭씨

* 1. **Result (Snapshot)**



* 1. **Analysis & Discussion**

- 함수의 반환형에 따라 반환하는 값의 형태(소수여부 등)이 변한다.

-

-

-

1. **369게임의 족보를 출력하는 프로그램(1~400)**
   1. **Solution**

- if-else문으로 별을 출력하는 부분과 숫자를 출력하는 부분을 분리하여, 별과 숫자가 동시에 출력되지 않도록 한다.

- if(369중 하나라도 들어가 있으면) 369의 개수만큼 별을 출력한다

- else원래 숫자를 출력한다.

* 1. **Source code**

/\*

\* 369jokbo.c

\*

\* Created on: 2017. 5. 9.

\* Author: 유동혁

\*/

#include <stdio.h>

//369게임의 족보를 출력하는 프로그램

int poham369(int num);

int main()

{

int n/\*세로줄 갯수\*/, i/\*가로 글자수\*/, k/\*별 출력 갯수\*/, num=1;

for(n=1; n<=40; n++)

{

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

{

if(poham369(num) != 0)

//count가 0이아니면= 369가 하나라도 포함되면

{

for(k=1; k<=poham369(num); k++)

printf("\*");

printf("\t");

}

else//369가 하나도 들어가지 않으면

printf("%d\t", num);//그냥 숫자 출력

num++;

}

printf("\n");

}

return 0;

}

//369를 포함하는지 알아보는 함수

int poham369(int num)

{

int t/\*num으로부터 분리\*/, count=0/\*369갯수\*/;

for(t=num; t>0; t/=10)

{

if(t%10==3 || t%10==6 || t%10==9)

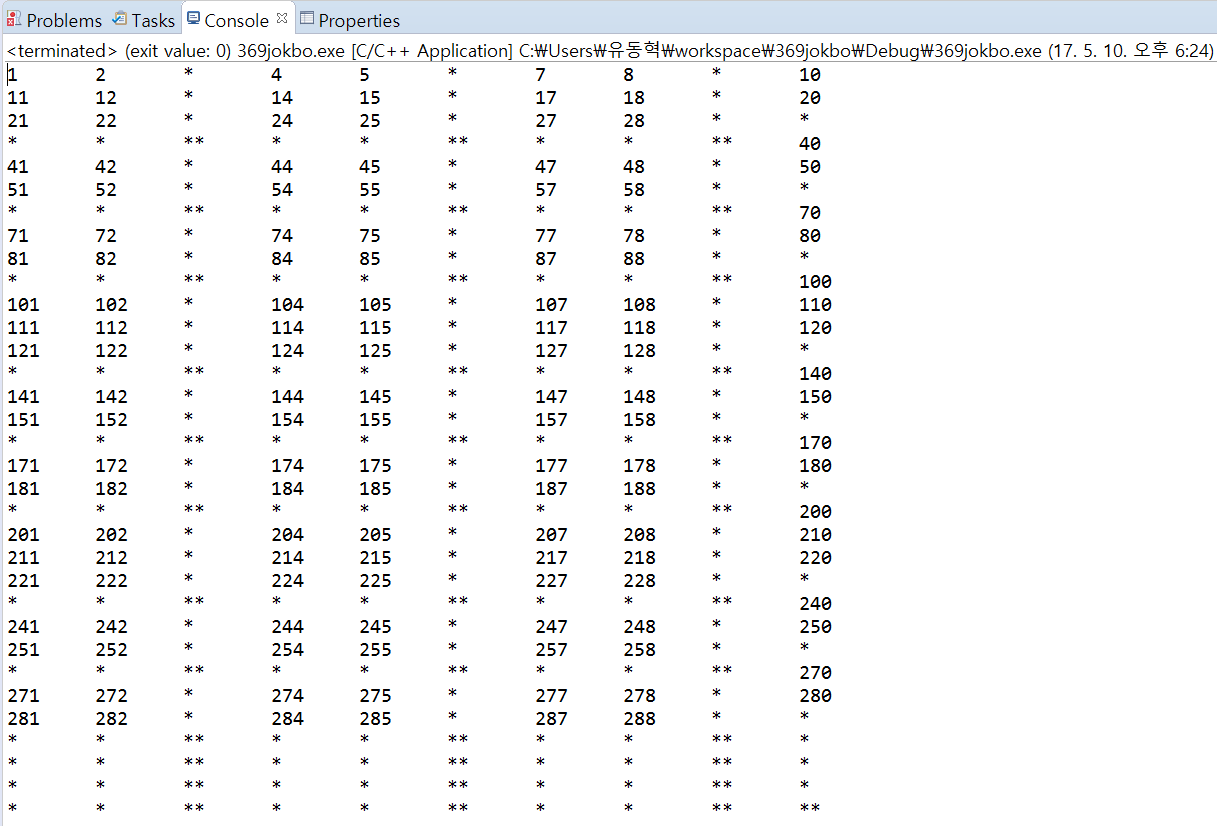
count++;//만약 369중 하나가 포함되면 \*갯수를 하나 증가

}

return count;

}

* 1. **Result (Snapshot)**



* 1. **Analysis & Discussion**

- 369의 존재 여부를 나의 방법(10으로 나눈 나머지가 369인지)말고 다른 방법으로도 파악할 수 있다고 한다. 그 방법은 무엇일까?

-

1. **사용자에게서 크기를 입력 받은 대로 도형을 출력하는 프로그램**
   1. **Solution**

- 함수가 반환 하는 값이 문자열이므로, 함수에char\*반환형을 사용한다

- 메뉴를 구성하기 위해 조건문을 사용한다.

- 원하는 부분(전부 다시 시작, 메뉴 다시 선택, 종료여부 다시 선택)부터 다시 시작하기 위해 goto를 사용한다.

* 1. **Source code**

/\*

\* byeoldohyeong.c

\*

\* Created on: 2017. 5. 9.

\* Author: 유동혁

\*/

#include <stdio.h>

//별로 메뉴에서 선택한 도형을 그려주는 프로그램

char\* square(int a, int b, int size);

char\* lowerleft(int a, int b, int size);

char\* upperleft(int a, int b, int size);

char\* lowerright(int a, int b, int size);

char\* upperright(int a, int b, int size);

char\* diagonal(int a, int b, int size);

char\* backdiagonal(int a, int b, int size);

char\* crossdiagonal(int a, int b, int size);

//반환값이 문자열이기 때문에 char\*사용

int main()

{

setvbuf(stdout, NULL, \_IONBF, 0);

char\* result;

int a, b, size, menu\_sel, end\_sel;

all\_retry://처음부터 다시하는 포인트

printf("\nEnter the size of a square : ");

scanf("%d", &size);

menu\_retry://메뉴를 다시 선택하는 포인트

printf("1: square\n"

"2: lower left\n"

"3: upper left\n"

"4: lower right\n"

"5: upper right\n"

"6: diagonal\n"

"7: back diagonal\n"

"8: cross diagonal\n"

"\n"

"Select a menu : ");

scanf("%d", &menu\_sel);

//메뉴 출력 후 메뉴 선택

switch(menu\_sel)

{

case 1:

result = square(a, b, size);

//위 라인<result ~ size);>에서 어떤Warnings가 발생하는데, 실행에는 문제없음. Analysis & Discussion참조

printf("%s\n", result);

break;

case 2:

result = lowerleft(a, b, size);

printf("%s\n", result);

break;

case 3:

result = upperleft(a, b, size);

printf("%s\n", result);

break;

case 4:

result = lowerright(a, b, size);

printf("%s\n", result);

break;

case 5:

result = upperright(a, b, size);

printf("%s\n", result);

break;

case 6:

result = diagonal(a, b, size);

printf("%s\n", result);

break;

case 7:

result = backdiagonal(a, b, size);

printf("%s\n", result);

break;

case 8:

result = crossdiagonal(a, b, size);

printf("%s\n", result);

break;

default ://1~8을 벗어난 숫자 선택 시

printf("Don't select other numbers\n");

goto menu\_retry;//메뉴 다시선택

break;

}

end\_retry://프로그램 종료여부 다시 선택 포인트

printf("1: continue\n"

"2: exit\n"

"Select a menu : ");

scanf("%d", &end\_sel);

if(end\_sel == 1)//1이면

goto all\_retry;

else if(end\_sel == 2)//1이 아니고 2면

return 0;

else//1,2 모두 아니면

{

printf("Don't select other numbers");

goto end\_retry;//종료여부 다시선택

}

}

char\* square(int a, int b, int size)

//정사각형 모양

{

for(a=1; a<=size; a++)

//세로방향

{

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

//가로방향

printf("\*");

printf("\n");

}

return 0;

}

char\* lowerleft(int a, int b, int size)

//왼쪽 정방향계단

{

for(a=1; a<=size; a++)

//세로방향

{

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

//가로방향

printf("\*");

printf("\n");

}

return 0;

}

char\* upperleft(int a, int b, int size)

//왼쪽 역방향계단

{

for(a=size; a>=1; a--)

//세로방향

{

for(b=a; b>=1; b--)

//가로방향

printf("\*");

printf("\n");

}

return 0;

}

char\* lowerright(int a, int b, int size)

//오른쪽 정방향계단

{

for(a=1; a<=size; a++)

//세로방향

{

for(b=1; b<=size-a; b++)

//공백출력

printf(" ");

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

//'\*'출력

printf("\*");

printf("\n");

}

return 0;

}

char\* upperright(int a, int b, int size)

//오른쪽 역방향계단

{

for(a=size; a>=1; a--)

//세로방향

{

for(b=size-a; b>=1; b--)

//공백출력

printf(" ");

for(b=a; b>=1; b--)

//'\*'출력

printf("\*");

printf("\n");

}

return 0;

}

char\* diagonal(int a, int b, int size)

//'\'모양

{

for(a=1; a<=size; a++)

//세로방향

{

for(b=1; b<=a-1; b++)

//공백출력

printf(" ");

printf("\*\n");

}

return 0;

}

char\* backdiagonal(int a, int b, int size)

//'/'모양

{

for(a=size; a>=1; a--)

//세로방향

{

for(b=a-1; b>=1; b--)

//공백출력

printf(" ");

printf("\*\n");

}

return 0;

}

char\* crossdiagonal(int a, int b, int size)

//'X'모양

{

switch(size%2)

//size가 홀수면 x모양 한가운데에 \*하나가 들어가야하고, 짝수면 x모양 한가운데가 \*네개로 구성됨

{

case 0://짝수일때

for(a=1; a<=size/2; a++)

//세로방향\_위쪽

{

for(b=1; b<=a-1; b++)

printf(" ");

printf("\*");

for(b=1; b<=size-2\*a; b++)

printf(" ");

printf("\*\n");

}

for(a=size/2; a>=1; a--)

//세로방향\_아래쪽

{

for(b=1; b<=a-1; b++)

printf(" ");

printf("\*");

for(b=1; b<=size-2\*a; b++)

printf(" ");

printf("\*\n");

}

break;

case 1://홀수일때

for(a=1; a<=size/2; a++)

//세로방향\_위쪽

{

for(b=1; b<=a-1; b++)

printf(" ");

printf("\*");

for(b=1; b<=size-2\*a; b++)

printf(" ");

printf("\*\n");

}

for(b=1; b<=size/2; b++)//한가운데에 \*출력

printf(" ");

printf("\*\n");

for(a=size/2; a>=1; a--)

//세로방향\_아래쪽

{

for(b=1; b<=a-1; b++)

printf(" ");

printf("\*");

for(b=1; b<=size-2\*a; b++)

printf(" ");

printf("\*\n");

}

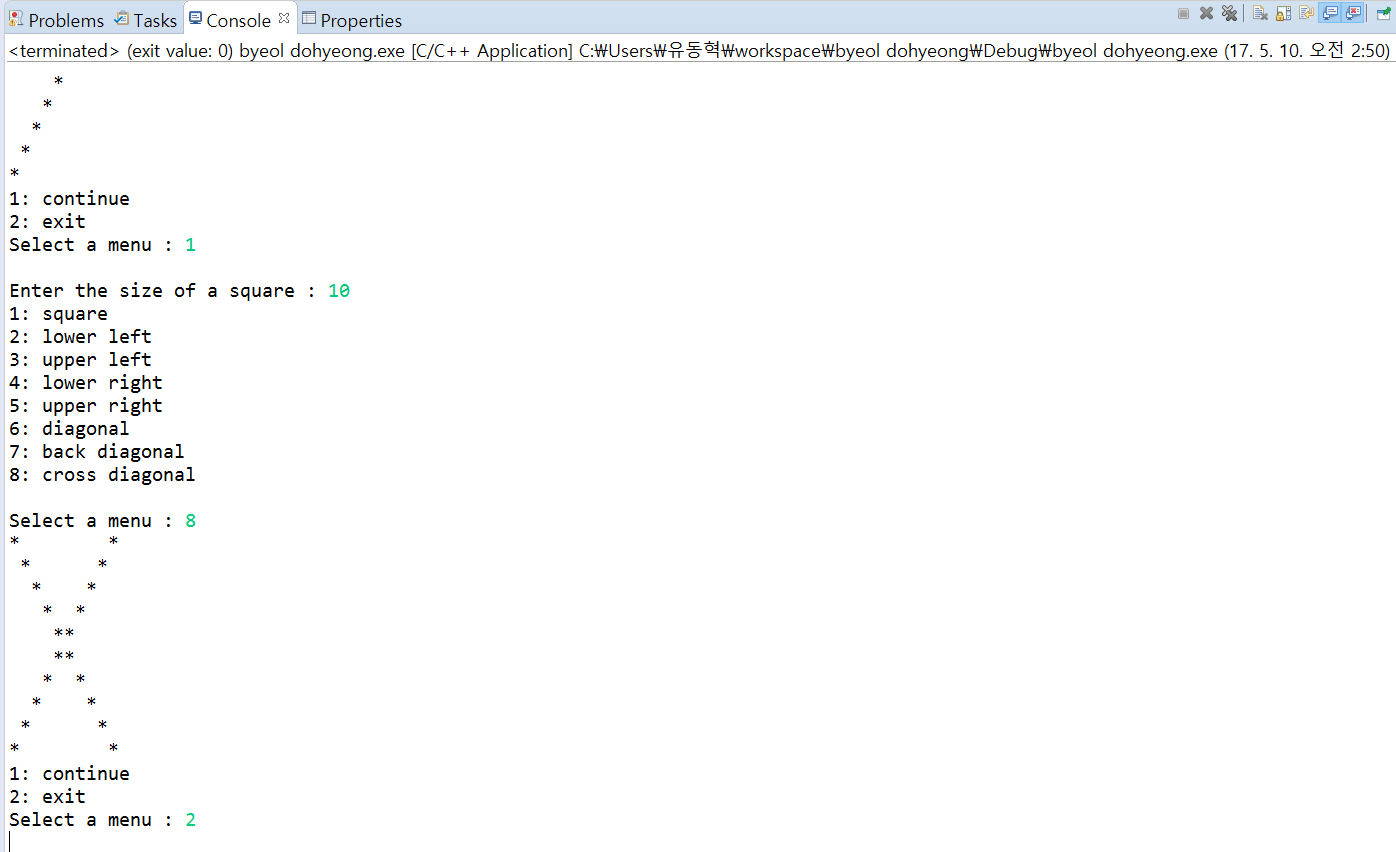
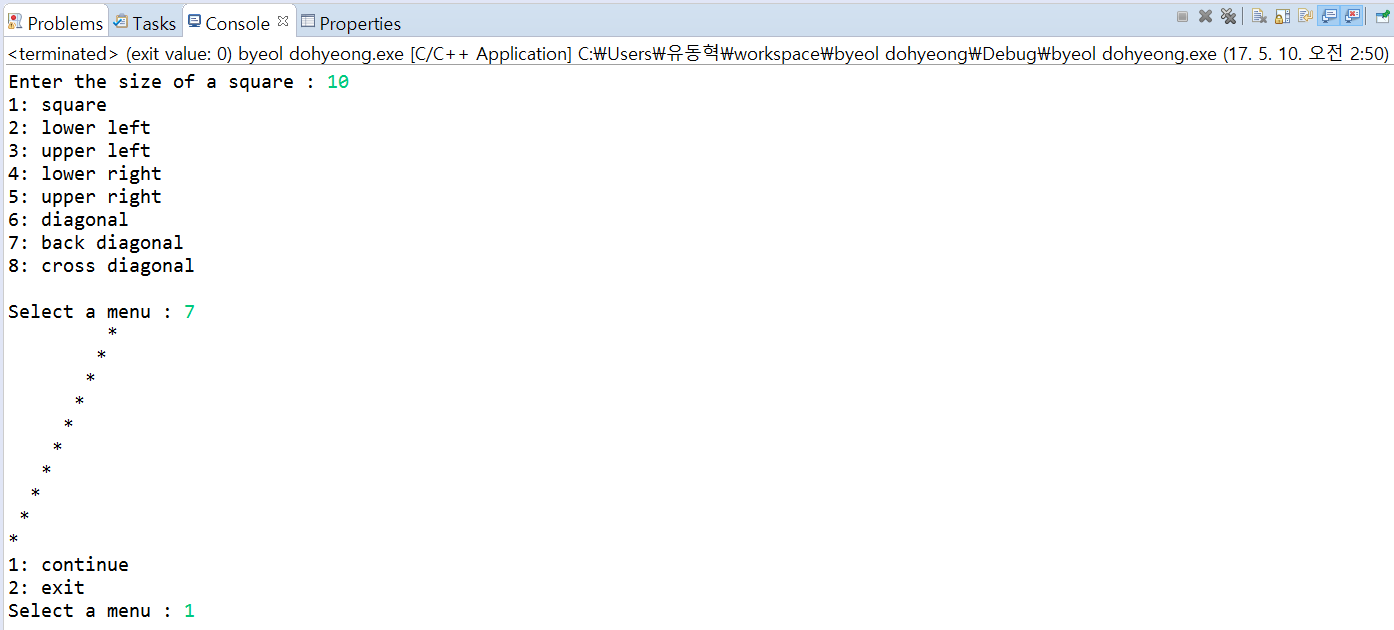
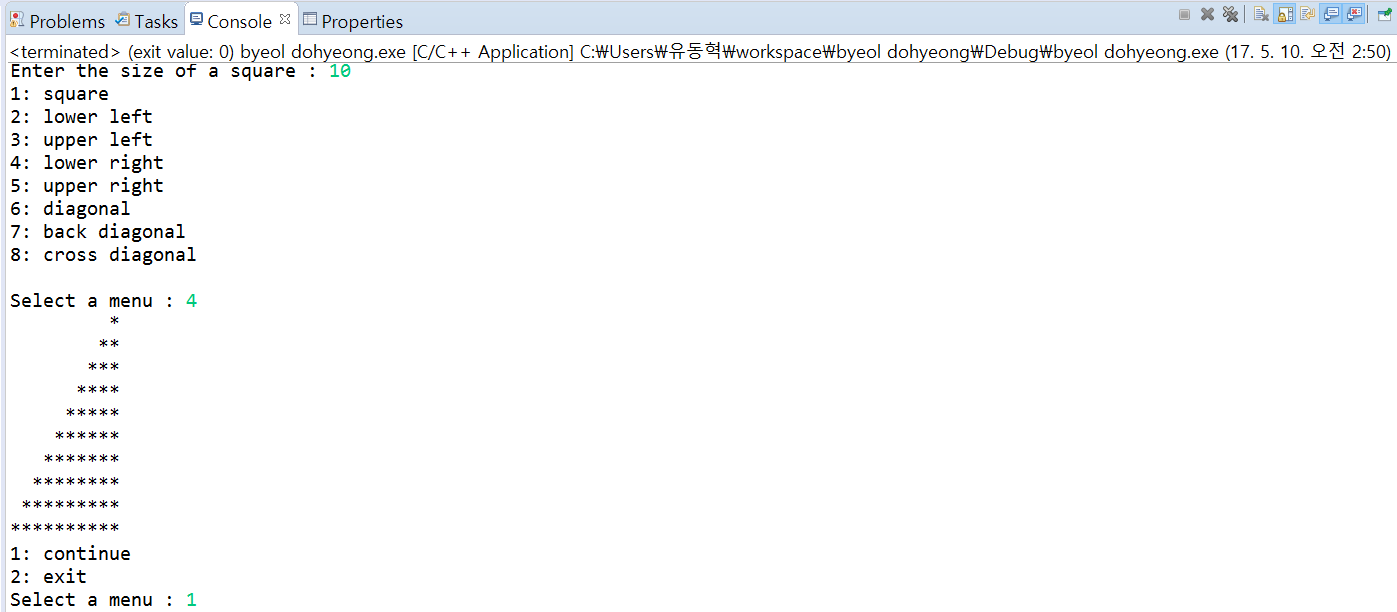
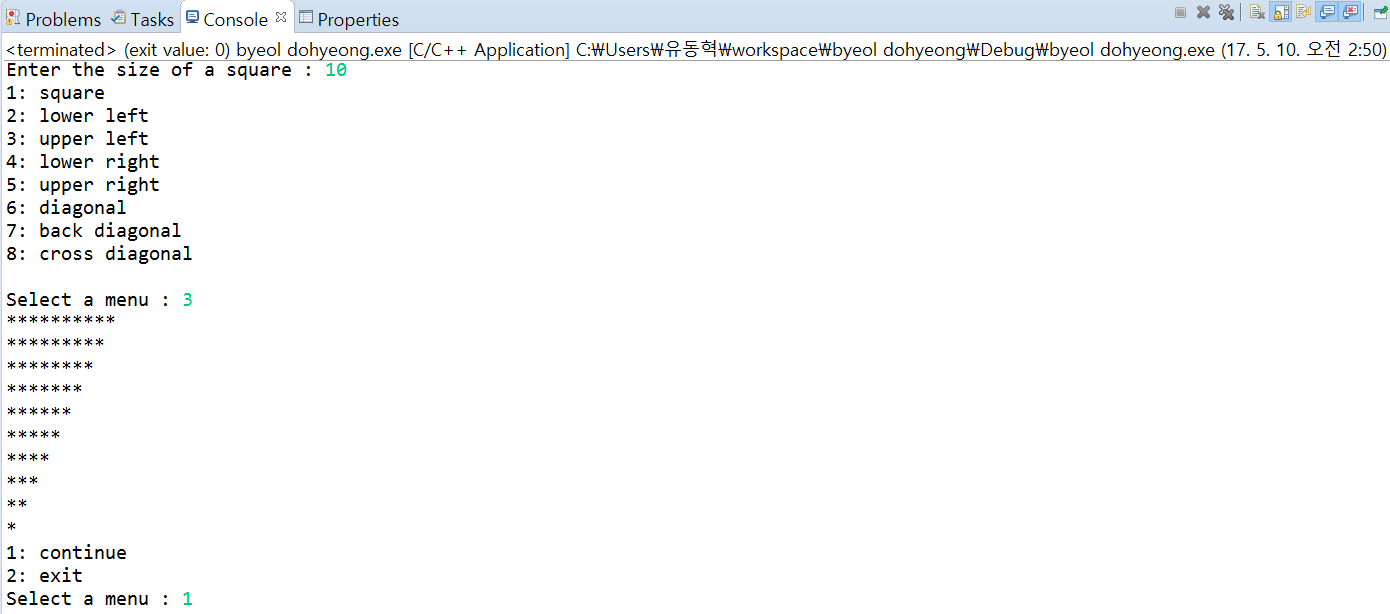
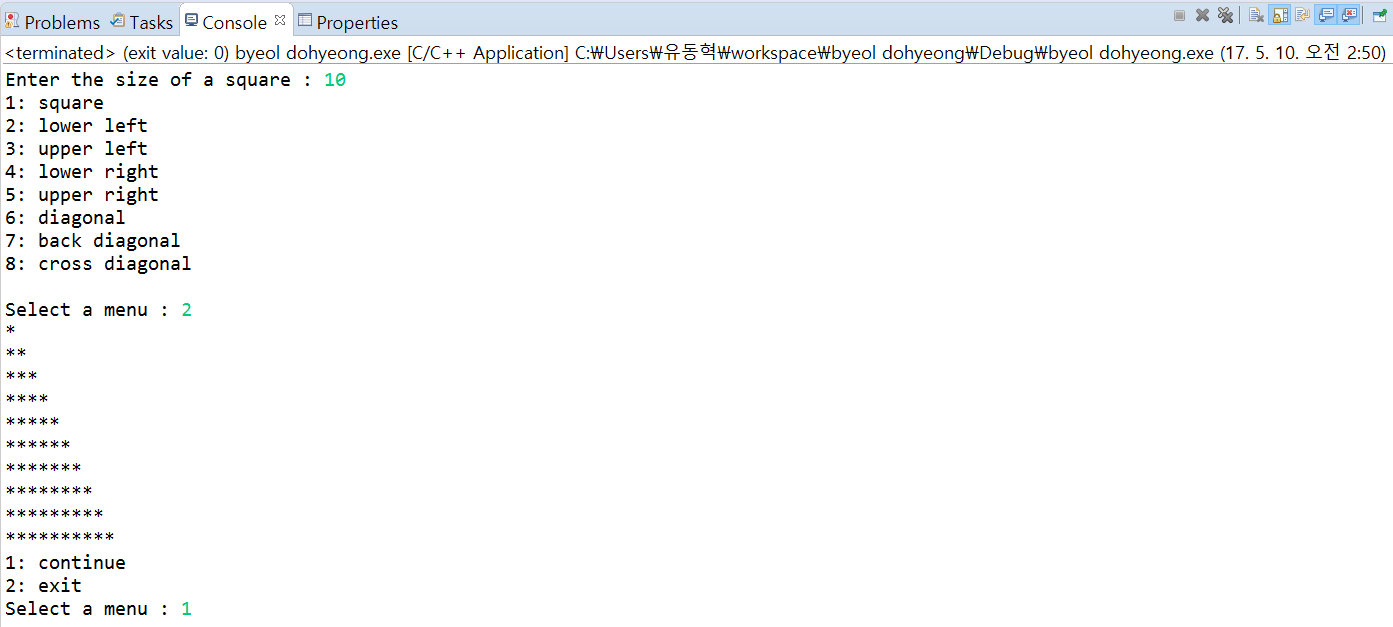
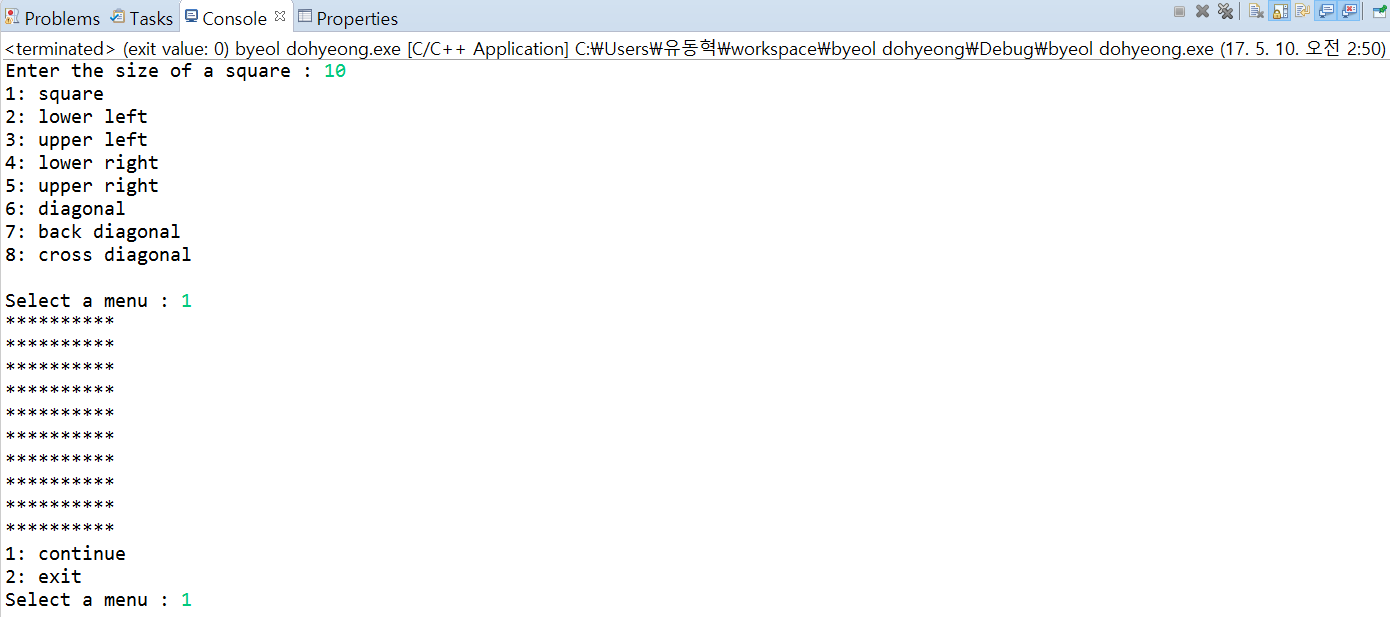
break;

}

return 0;

}

* 1. **Result (Snapshot)**



* 1. **Analysis & Discussion**

-menu의 case1에서 첫번째 줄에서(square) 다음과 같은 에러가 발생하는데, 다른 도형함수와 코드작성을 똑같이 했음에도 불구하고 이 함수에서만 이런 에러가 발생한다. 프로그램 실행에는 문제가 없어서 그냥 넘어갔다.

-switch문과 if문을 둘다 사용해 본 결과 메뉴를 구성할 때는 switch문이 좀 더 간편하다.

1. **컴퓨터가 지정한 무작위한 숫자를 맞추는 숫자야구 프로그램**
   1. **Solution**

- 임의의 수를 얻기 위해 stdlib.h헤더파일과 rand(), 계속 같은 숫자만 나오는 걸 피하기 위해time.h헤더 파일과 srand(()time(NULL))을 사용한다.

- 사용자에게서 세 자리 수를 입력 받고 각 자리를 분리해서 비교하기위해 10으로 나눈 나머지 구하기--10으로 나누기를 반복한다.

- 임의의 숫자는 각 자리에 중복되는 숫자(ex. 999)가 있으면 안된다. 그런 숫자가 나왔을 때 다시 숫자를 정하게 한다.

- 스트라이크 카운트가 0일 때 마다 노\_스트라이크 카운트를 하나 추가하고, 3이 되면 힌트변수에 임의의 숫자를 저장해 몇 번째 자리의 숫자를 공개할 지 결정한다.

* 1. **Source code**

/\*

\* sutjayagu.c

\*

\* Created on: 2017. 5. 10.

\* Author: 유동혁

\*/

#include<stdio.h>

#include<time.h>

#include<stdlib.h>

//컴퓨터가 정한 임의의 숫자를 맞추는 숫자야구 프로그램

int strike\_count(int com1, int com2, int com3, int num1, int num2, int num3);

int ball\_count(int com1, int com2, int com3, int num1, int num2, int num3);

int main()

{

setvbuf(stdout, NULL, \_IONBF, 0);

int num, com; //사용자가 고른 숫자, 컴퓨터가 지정한 숫자

int com1, com2, com3;//세 자리 수를 비교하기 위해 숫자 3개로 분리

int num1, num2, num3;//상동

int hint;//hint에 1,2 또는 3을 저장해 몇번째 자리를 공개할지 결정

int strike, ball, no\_strike=0;//스트라이크, 볼 카운트 , 스트라이크가 세번안나오면 힌트출력

int count=0;//게임 회차

srand((unsigned) time(NULL));//rand에서 같은 숫자가 계속 나오는걸 방지

RE\_RAND://랜덤한 숫자가 조건을 충족시키지않을때 다시 결정하게 함

com = rand() %1000; // 임의의 수 지정

com3 = com % 10;

com /= 10;

com2 = com % 10;

com /=10;

com1 = com % 10;

//세 자리 수를 비교하기 위해 숫자 3개로 분리

while (strike<3)//스트라이크 == 3이 되면 종료

{

if (com1 != com2 && com1 != com3 && com2 != com3) //숫자가 겹치면 안된다.

{

count++;

//게임 회차 증가

strike = 0;

ball = 0;

//스트라이크, 볼 초기화

printf("숫자야구 - 세자리 수를 입력하세요 : ");

scanf("%d", &num);

num3 = num % 10;

num /= 10;

num2 = num % 10;

num /= 10;

num1 = num % 10;

//컴퓨터의 숫자와 비교하기위해 세 숫자로 분리

strike = strike\_count(com1, com2, com3, num1, num2, num3);

ball = ball\_count(com1, com2, com3, num1, num2, num3);

//스트라이크, 볼 판별 함수 사용

if(strike == 0)//0스트라이크면

no\_strike++;

else//0스트라이크가 아니면

no\_strike = 0;//힌트 발생 조건은 0스트라이크 연속 세번이므로 초기화

printf("--------------------------\n");

if(no\_strike == 3)//0스트라이크가 세번 연속되면

{

hint = rand()%3+1;//com1,2,3중 뭘 공개할지 정함

switch(hint)

{

case 1:

printf("Hint : %d\_\_ Count : %d\n", com1, count);

break;

case 2:

printf("Hint : \_%d\_ Count : %d\n", com2, count);

break;

case 3:

printf("Hint : \_\_%d Count : %d\n", com3, count);

break;

}

}

else//힌트 발생 조건을 만족 X

{

printf("Strike : %d Ball : %d Count : %d\n", strike, ball, count);

}

printf("--------------------------\n");

}

else//임의의 수가 조건을 만족하지 않을 때

goto RE\_RAND;//다시 정하는 곳으로 감

}

return 0;

}

//스트라이크를 정하는 함수

int strike\_count(int com1, int com2, int com3, int num1, int num2, int num3)

{

int strike = 0;

if(com1 == num1)

strike++;

if(com2 == num2)

strike++;

if(com3 == num3)

strike++;

return strike;

}

//볼을 정하는 함수

int ball\_count(int com1, int com2, int com3, int num1, int num2, int num3)

{

int ball=0;

if(com1 == num2)

ball++;

if(com1 == num3)

ball++;

if(com2 == num1)

ball++;

if(com2 == num3)

ball++;

if(com3 == num1)

ball++;

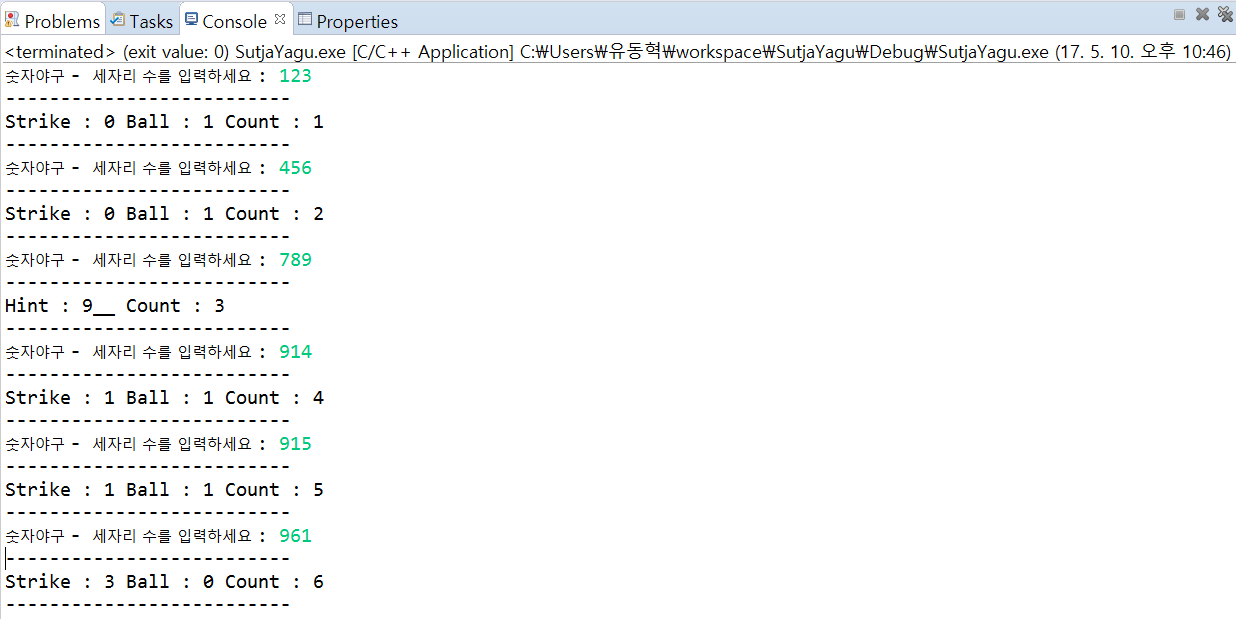
if(com3 == num2)

ball++;

return ball;

}

* 1. **Result (Snapshot)**



* 1. **Analysis & Discussion**

- 자릿수를 분리할 때 함수를 이용하려 했는데 한번에 여러 변수를 반환할 수가 없어서 함수를 이용하지 않았다.

- return값을 상황에 맞게 사용하는 부분이 어렵다. 어떤 때는 반환 값을 0으로 해야 하고, 어떤 때는 반환 값을 변수로 해야 한다.

-

-