\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Report: hw5

Author: F74046022 陳冠仁 <jeremy851004@gmail.com>

Class: 乙班

Description:

How do you finish this homework?

將先前的猜測數據(猜測、H、X)存入陣列

並以此過濾下次猜測

What did you learned from this homework?

遞迴陣列的基本應用

divide and conquer的概念

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* 1A2B (COM PLAYER) \*

\* PURPOSE : \*

\* GET THE ANSWER WITH MINIMUM GUESSES. \*

\* SOLUTION : \*

\* STORE GUESSES AND RESULTS \*

\* USE THEM AS FILTERS FOR THE NEXT GUESS \*

\* FLAWS : \*

\* PROCESS TIME IS PROLONGED WITH EACH GUESS. \*

\* (MORE FILTERS -> HARGER TO GET THROUGH) \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int i, j, k;

//loop variables

int main(int argc, char \*argv[])

{

int n=atoi(argv[1]), p=atoi(argv[2]);

int tri=0, cnc=n\*10, z=0;

int h[cnc+1], x[cnc+1], gue[cnc+1][p], ans[p];

//n->range p->positions cnc->chances

//h[], x[], g[][]->storage for guess data

//ans[]->storage for answer

srand(time(0));

if(n<1 || p<1 || n<p) return 0;

//exit when inputs are invalid

gnr\_rnd(n, p, ans);

print\_ans(z, p, ans);

//generate answer

do{

tri++;

print\_try(tri);

guess(n, p, tri, gue, h, x);

cmp\_ans\_gue(p, tri, ans, gue, h, x);

}while(h[tri]!=p);

//main part of the game

printf("\n\n");

return 0;

}

int gnr\_rnd(int n, int p, int a[])

{

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

{

a[i]=rand()%n+1;

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

if(a[i]==a[j])

{

i--;

break;

}

}

}

//generate random, non-repetitive array

int print\_ans(int z, int p, int ans[])

{

if(z==0) printf("\n ANSWER :");

if(z==p) return;

printf("%3d", ans[z++]);

print\_ans(z, p, ans);

}

int print\_try(int tri)

{

printf("\n%2d", tri);

switch(tri%10)

{

case 1: printf("st"); break;

case 2: printf("nd"); break;

case 3: printf("rd"); break;

default: printf("th"); break;

}

printf(" guess :");

}

int guess(int n, int p, int tri, int gue[][p], int h[], int x[])

{

int dif, H, X;

do{

dif=0;

gnr\_rnd(n, p, gue[tri]);

//random guess

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

{

H=0; X=0;

for(j=0; j<p; j++)

{

if(gue[tri][j]==gue[i][j])

H++;

for(k=0; k<p; k++)

if(gue[tri][j]==gue[i][k] && j!=k)

X++;

}

if(H!=h[i] || X!=x[i])

{

dif++;

break;

}

}

//use previous guess, h, x to filter

}while(dif!=0);

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

printf("%3d", gue[tri][i]);

}

//generate guess

int cmp\_ans\_gue(int p, int tri, int ans[], int gue[][p], int h[], int x[])

{

h[tri]=0; x[tri]=0;

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

{

if(gue[tri][i]==ans[i])

h[tri]++;

for(j=0; j<p; j++)

if(gue[tri][i]==ans[j] && i!=j)

x[tri]++;

}

printf("\t%3dH%3dX", h[tri], x[tri]);

}

//calculate and store h, x

Compilation:

gcc –o hw5 hw5.c

Execution:

./hw5 (N) (P)

Output:

ANSWER : 5 4 3 8 6

1st guess : 1 5 9 8 7 1H 1X

2nd guess : 3 6 9 4 1 0H 3X

3rd guess : 9 2 3 8 4 2H 1X

4th guess : 8 5 3 6 4 1H 4X

5th guess : 5 3 6 8 4 2H 3X

6th guess : 5 4 3 8 6 5H 0X

/tmp/ccnGAJI3.o: In function `main':

hw5.c:(.text+0x338): undefined reference to `gnr\_rnd'

/tmp/ccnGAJI3.o: In function `guess':

hw5.c:(.text+0x6b2): undefined reference to `gnr\_rnd'

collect2: error: ld returned 1 exit status

hw5.c: In function ‘main’:

hw5.c:47:2: error: expected ‘;’ before ‘printf’

printf("\n\n");

^

hw5.c: In function ‘gnr\_rnd’:

hw5.c:131:1: error: expected declaration or statement at end of input

}

^