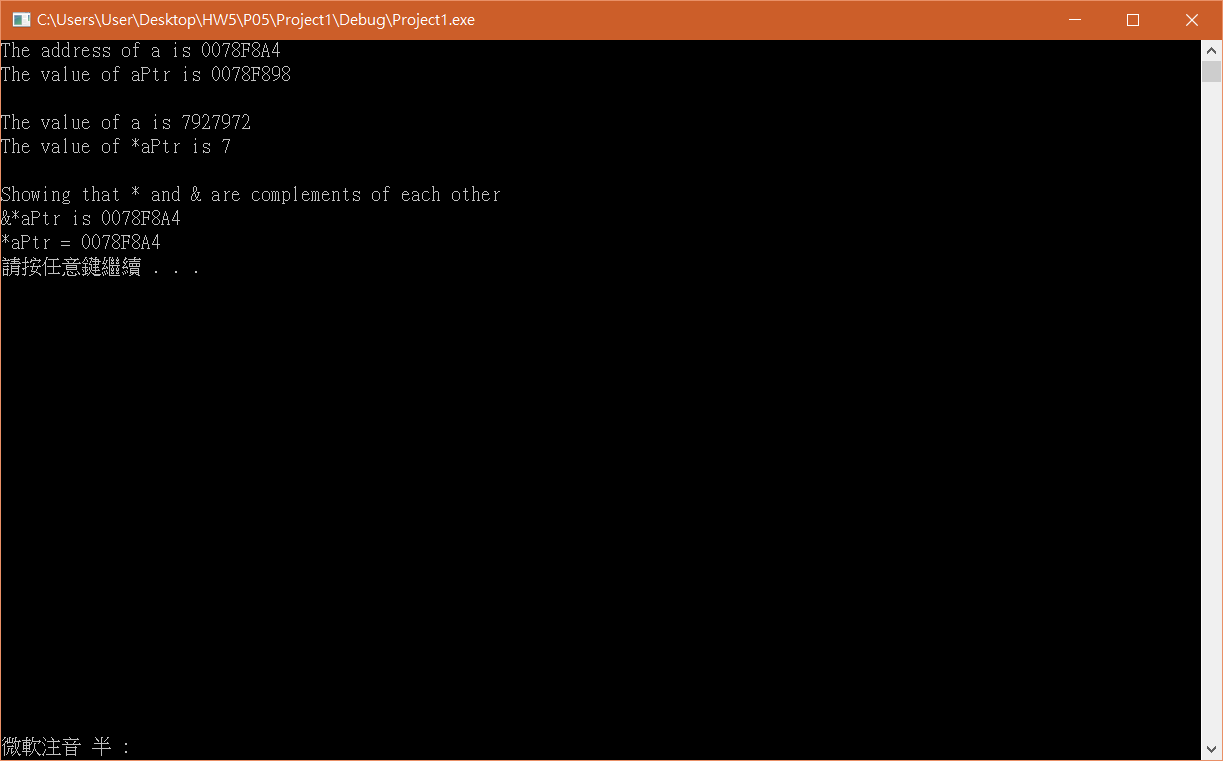
P05



#include <stdio.h>

#include <stdlib.h>

int main(void)

{

int a;

int \*aPtr;

a = 7;

aPtr = &a;

printf("The address of a is %p"

"\nThe value of aPtr is %p", &a, &aPtr);

printf("\n\nThe value of a is %d"

"\nThe value of \*aPtr is %d", &a, \*aPtr);

printf("\n\nShowing that \* and & are complements of "

"each other\n&\*aPtr is %p"

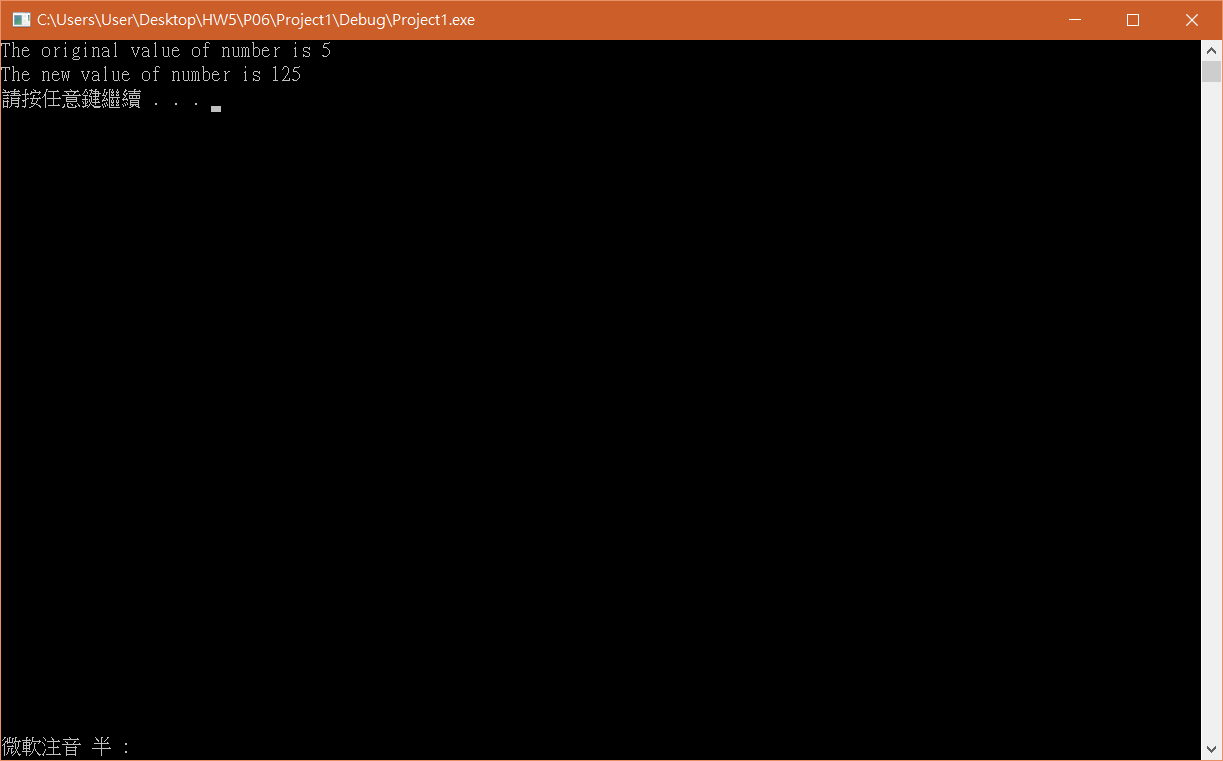
"\n\*aPtr = %p\n", &\*aPtr, \*&aPtr);

system("pause");

return 0;

}

P06



#include <stdio.h>

#include <stdlib.h>

int cubeByValue(int n);

int main(void)

{

int number = 5;

printf("The original value of number is %d", number);

number = cubeByValue(number);

printf("\nThe new value of number is %d\n", number);

system("pause");

return 0;

}

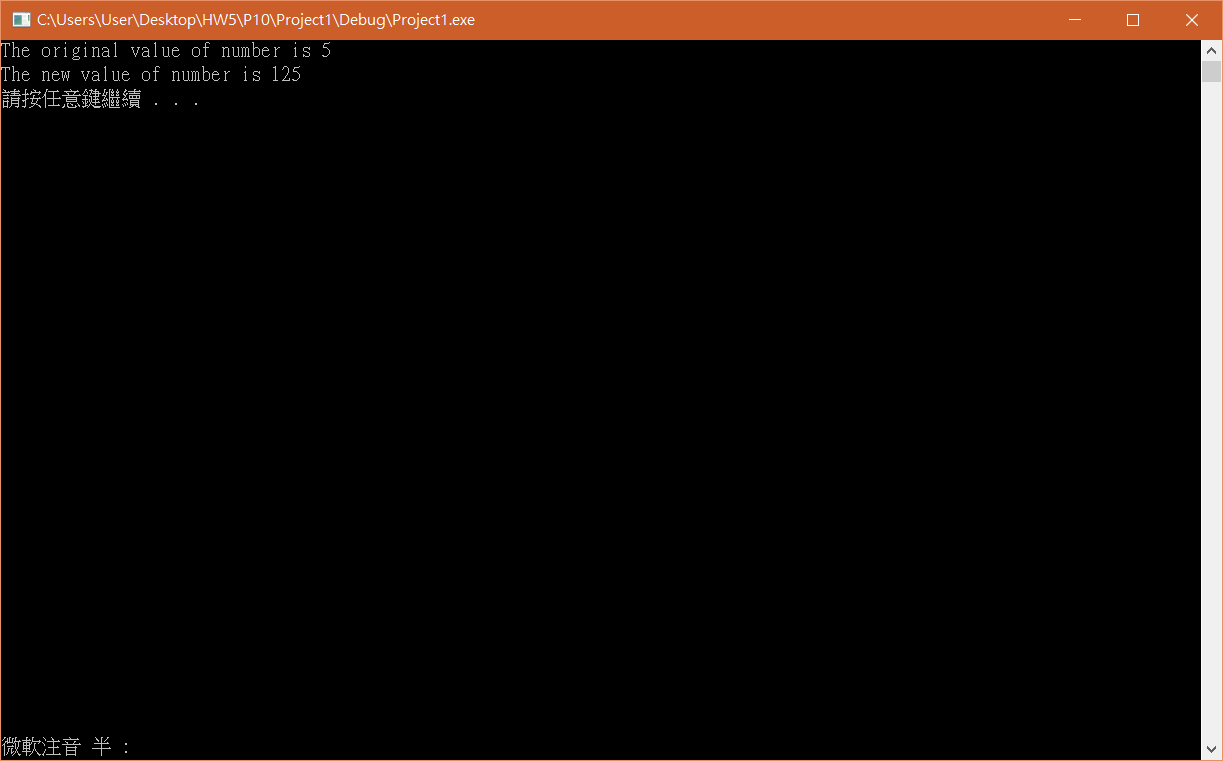
int cubeByValue(int n)

{

return n \* n \* n;

}

P10



#include <stdio.h>

#include <stdlib.h>

void cubeByReference(int \* nPtr);

int main(void)

{

int number = 5;

printf("The original value of number is %d", number);

cubeByReference(&number);

printf("\nThe new value of number is %d \n", number);

system("pause");

return 0;

}

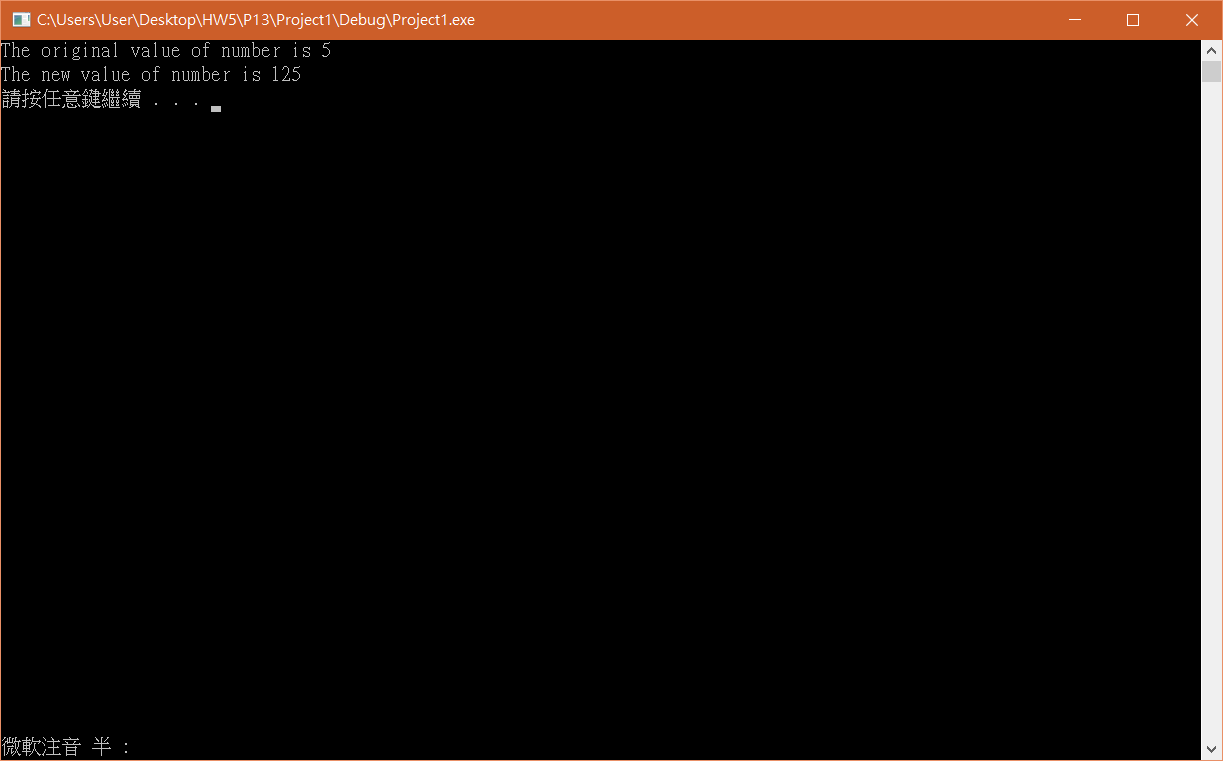
void cubeByReference(int \* nPtr)

{

\*nPtr = \*nPtr \* \* nPtr \* \* nPtr;

}

P13



#include <stdio.h>

#include <stdlib.h>

void cubeByReference(int &nPtr);

int main(void)

{

int number = 5;

printf("The original value of number is %d", number);

cubeByReference(number);

printf("\nThe new value of number is %d \n", number);

system("pause");

return 0;

}

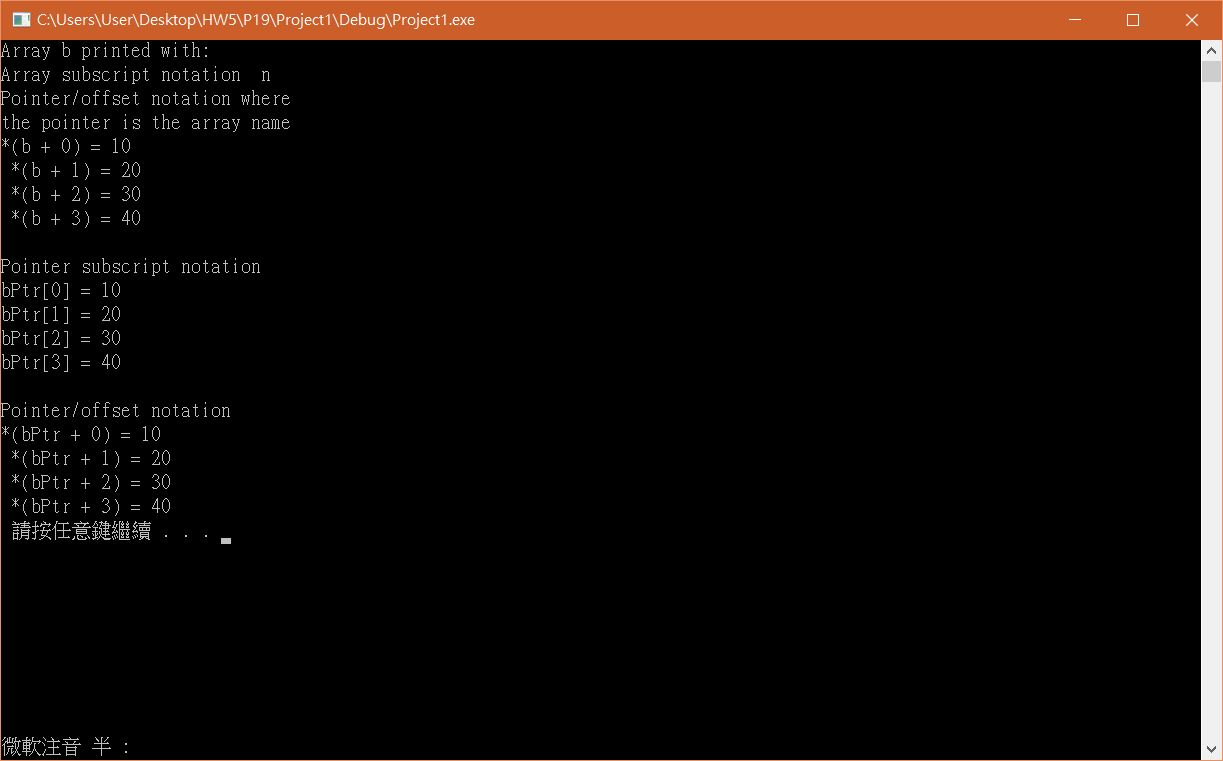
void cubeByReference(int &nPtr)

{

nPtr = nPtr \* nPtr \* nPtr;

}

P15



#include <stdio.h>

#include <stdlib.h>

int main(void)

{

int b[] = { 10,20,30,40 };

int \* bPtr = b;

int i;

int offset;

printf("Array b printed with: \nArray subscript notation \ n");

printf("\nPointer/offset notation where\n"

"the pointer is the array name\n");

for (offset = 0; offset < 4; offset++)

printf("\*(b + %d) = %d \n ", offset, \*(b + offset));

printf("\nPointer subscript notation\n");

for (i = 0; i < 4; i++)

printf("bPtr[%d] = %d\n", i, bPtr[i]);

printf("\nPointer/offset notation\n");

for (offset = 0; offset < 4; offset++)

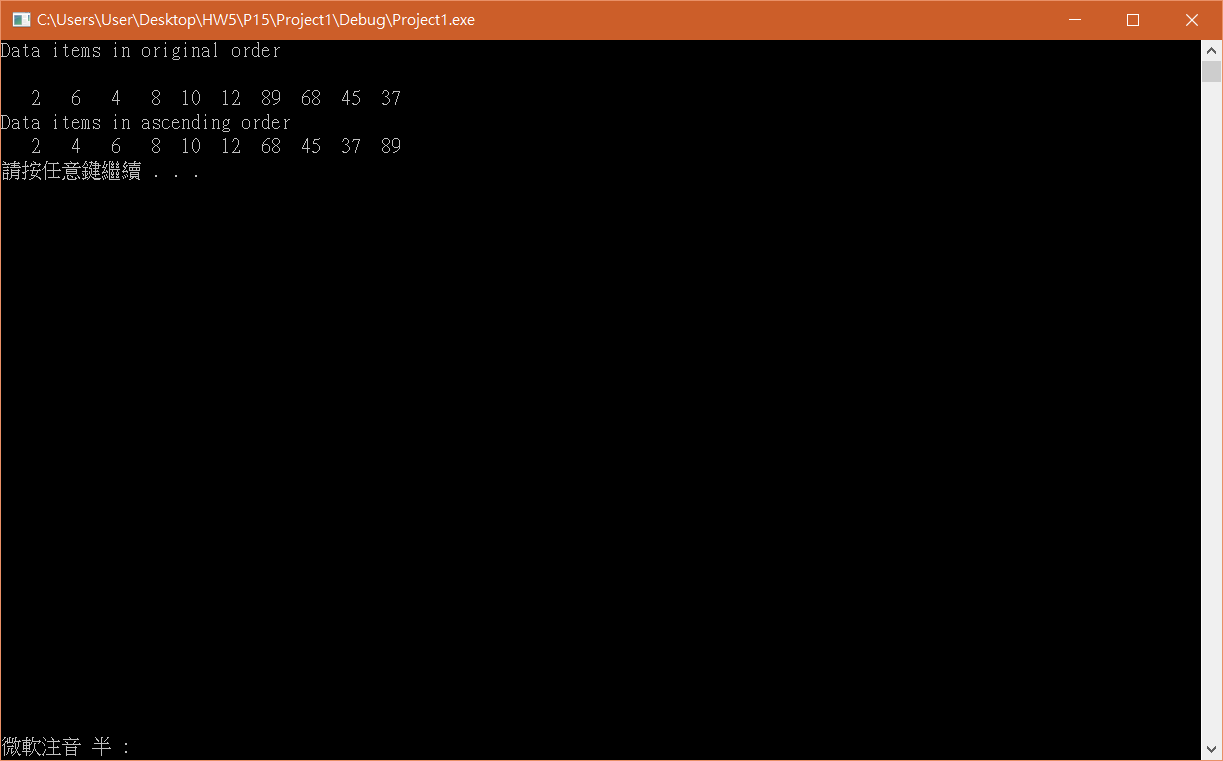
printf("\*(bPtr + %d) = %d \n ", offset, \*(bPtr + offset));

system("pause");

return 0;

}

P19



#include<stdio.h>

#include<stdlib.h>

#define SIZE 10

void bubbleSort(int \* const array, const int size);

void swap(int \*element1Ptr, int \*element2Ptr);

int main(void)

{

int a[SIZE] = { 2,6,4,8,10,12,89,68,45,37 };

int i;

puts("Data items in original order\n");

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

printf("%4d", a[i]);

bubbleSort(a, SIZE);

printf("\nData items in ascending order\n");

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

printf("%4d", a[i]);

printf("\n");

system("pause");

return 0;

}

void bubbleSort(int \* const array, const int size)

{

int pass,j;

for ( j = 0; j < size-1; j++)

{

if (array[j] > array[j + 1])

swap(&array[j], &array[j + 1]);

}

}

void swap(int \*element1Ptr, int \*element2Ptr)

{

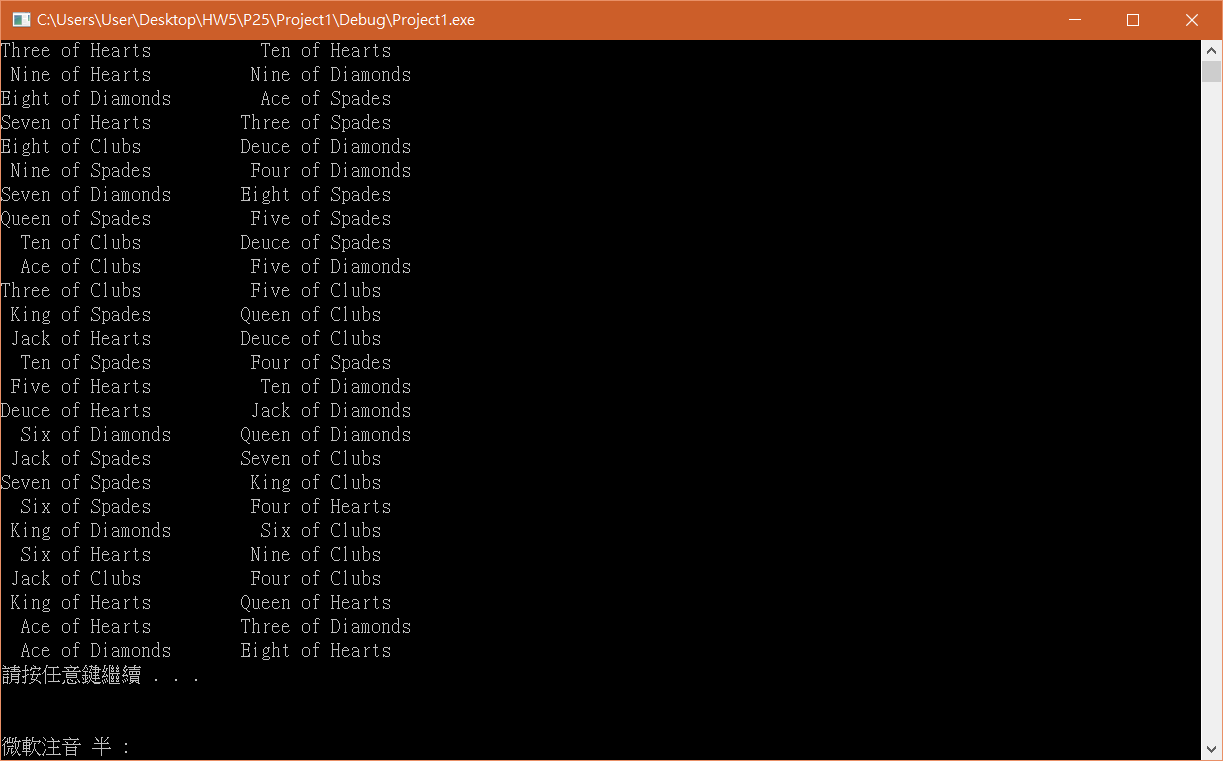
int hold = \*element1Ptr;

\*element1Ptr = \*element2Ptr;

\*element2Ptr = hold;

}

P25



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void shuffle(int wDeck[][13]);

void deal(const int wDeck[][13], const char \*wFace[],

const char \*wSuit[]);

int main(void)

{

const char \*suit[4] = { "Hearts","Diamonds","Clubs","Spades" };

const char \*face[13] =

{ "Ace" ,"Deuce","Three","Four",

"Five","Six" ,"Seven","Eight",

"Nine","Ten" ,"Jack" ,"Queen","King" };

int deck[4][13] = { 0 };

srand(time(0));

shuffle(deck);

deal(deck, face, suit);

system("pause");

return 0;

}

void shuffle(int wDeck[][13])

{

int row, column, card;

for (card = 1; card <= 52; card++)

{

do

{

row = rand() % 4;

column = rand() % 13;

} while (wDeck[row][column] != 0);

wDeck[row][column] = card;

}

}

void deal(const int wDeck[][13], const char \*wFace[],

const char \*wSuit[])

{

int card, row, column;

for (card = 1; card <= 52; card++)

{

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

{

for (column = 0; column <= 12; column++)

{

if (wDeck[row][column] == card)

{

printf("%5s of %-8s%c", wFace[column], wSuit[row],

card % 2 == 0 ? '\n' : '\t');

}

}

}

}

}

Git

心得:

對於這次所教的指數變數有所了解，雖然有點抽象但在老師與助教的教導下學會了。