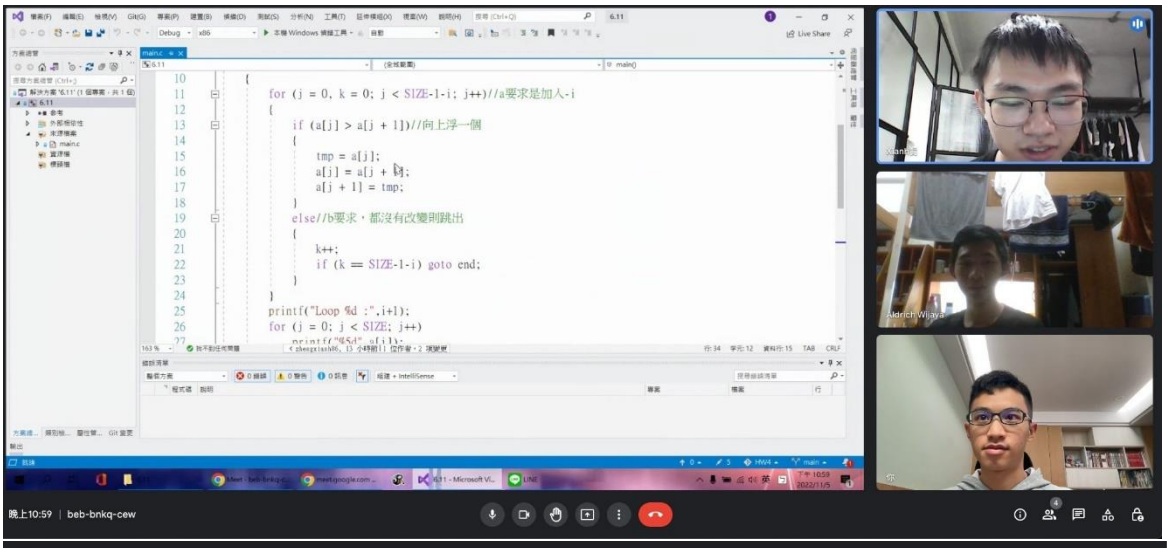
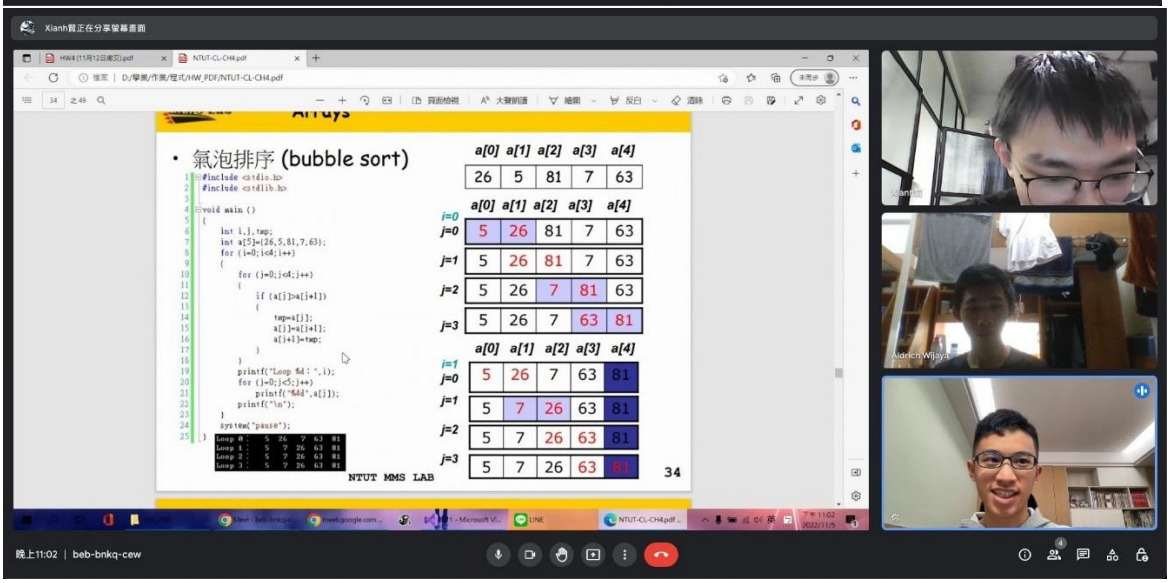
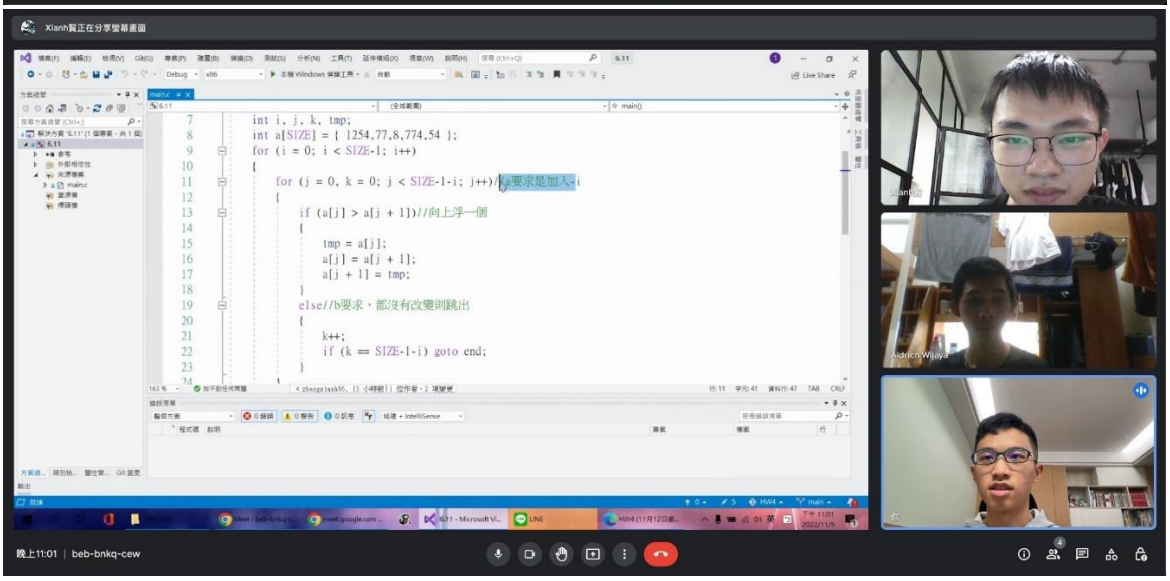
Discussion time: 5th November 2022, 22:57-23:04







P04

main.c

```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int n[10], i;

    for (i = 0; i < 10; i++)
    {
        n[i] = 0;
    }

    printf("%s%10s\n", "Element", "Value");

    for (i = 0; i < 10; i++)
    {
        printf("%7d%10d\n", i, n[i]);
    }

    system ("pause");
    return 0;
}
```

P05

main.c

```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int n[10] = {15,14,25,74,26,82,58,24,85,24};
    int i;

    printf("%s%10s\n", "Element", "Value");

    for (i = 0; i < 10; i++)
    {
        printf("%7d%10d\n", i, n[i]);
    }

    system ("pause");
    return 0;
}
```

P06

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 10

int main()
{
    int n[SIZE];
    int i;

    for (i = 0; i < SIZE; i++)
    {
        n[i] = 2+2*i;
    }

    printf("%s%10s\n", "Element", "Value");

    for (i = 0; i < SIZE; i++)
    {
        printf("%7d%10d\n", i, n[i]);
    }

    system ("pause");
    return 0;
}
```

P07

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 10

int main()
{
    int n[SIZE] = {15,14,25,74,26,82,58,24,85,24};
    int i;
    int total = 0;

    for (i = 0; i < SIZE; i++)
    {
        total += i[n];
    }

    printf("Total of array element value is: %d\n", total);

    system ("pause");
    return 0;
}
```

P08

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 10

int main()
{
    int n[SIZE] = {10,1,9,2,8,3,7,4,6,5};
    int i, j;

    printf("%s%10s%15s\n", "Element", "Value", "Histogram");

    for (i = 0; i < SIZE; i++)
    {
        printf("%7d%10d    ", i, n[i]);
        for (j = 1; j <= n[i]; j++)
        {
            printf("*");
        }
        printf("\n");
    }

    system ("pause");
    return 0;
}
```

P13

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 10

void printer(const int a[][3]);

int main()
{
    int A[2][3] = {{1,2,3},{4,5,6}};
    int B[2][3] = {1,2,3,4,5};
    int C[2][3] = {{1,2},{3}};

    printf("Values is A by row is:\n");
    printer(A);

    printf("Values is B by row is:\n");
    printer(B);

    printf("Values is C by row is:\n");
    printer(C);

    system ("pause");
    return 0;
}

void printer(const int a[][3])
{
    int i, j;

    for (i = 0; i <= 1; i++)
    {
        for (j = 0; j <= 2; j++)
        {
            printf("%d ", a[i][j]);
        }
        printf("\n");
    }
}
```

P14

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define STUDENT 3
#define EXAM 4

void printer(const int grade[][EXAM], int pupil, int test);
int min(const int grade[][EXAM], int pupil, int test);
int max(const int grade[][EXAM], int pupil, int test);
double avg(const int setgrade[], int test);

int main()
{
    int student;
    const int sGrade[STUDENT][EXAM] =
    {{67,89,90,79},{80,90,94,85},{78,87,98,70}};

    printf("Array:\n");
    printer(sGrade, STUDENT, EXAM);

    printf("\n\nLowest grade: %d\nHighest grade: %d\n", min(sGrade, STUDENT, EXAM), max(sGrade, STUDENT, EXAM));

    for (student = 0; student < STUDENT; student++)
    {
        printf("The average score of student %d is: %.2f\n", student, avg(sGrade[student], EXAM));
    }

    system ("pause");
    return 0;
}

void printer(const int grade[][EXAM], int pupil, int test)
{
    int i, j;

    printf("         [0]   [1]   [2]   [3]");

    for (i = 0; i < pupil; i++)
    {
        printf("\nsGrade[%d] ", i);
        for (j = 0; j < test; j++)
        {
            printf("%-5d ", grade[i][j]);
        }
    }
}

int min(const int grade[][EXAM], int pupil, int test)
{
    int i, j;
    int low = 100;

    for (i = 0; i < pupil; i++)
    {
        for (j = 0; j < test; j++)
        {
            if (grade[i][j]<low)
                low = grade[i][j];
        }
    }
    return low;
}

int max(const int grade[][EXAM], int pupil, int test)
{
    int i, j;
    int high = 0;

    for (i = 0; i < pupil; i++)
    {
        for (j = 0; j < test; j++)
        {
            if (grade[i][j]>high)
                high = grade[i][j];
        }
    }
    return high;
}

double avg(const int setgrade[], int test)
{
    int i;
    int total = 0;

    for (i = 0; i < test; i++)
    {
        total += setgrade[i];
    }

    return (double)total/test;
}
```

P19

main.c

```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    char string1[20];
    char string2[] = "Not Bruh Moment";
    int i;

    printf("Enter a string: ");
    scanf("%s", string1);

    printf("string1 is: %s\nstring2 is: %s\n"
           "string 1 with spaces in between characters is: \n", string1, string2);

    for (i = 0; string1[i] != '\0'; i++)
    {
        printf("%c ", string1[i]);
    }
    printf("\n");
    system ("pause");
    return 0;
}

```

P21

main.c

```c
#include <stdio.h>
#include <stdlib.h>

void Static(void);
void Auto(void);

int main()
{
    printf("First call to each function:\n");
    Static();
    Auto();

    printf("\n\n\nSecond call to each function:\n");
    Static();
    Auto();

    printf("\n");
    system ("pause");
    return 0;
}

void Static(void)
{
    static int array1[3];
    int i;

    printf("\nStatic former value:\n");

    for (i = 0; i <= 2; i++)
        printf("array1[%d] = %d  ", i, array1[i]);

    printf("\nStatic latter value:\n");

    for (i = 0; i <= 2; i++)
        printf("array1[%d] = %d  ", i, array1[i]+=5);
}

void Auto(void)
{
    int array2[3];
    int i;

    printf("\n\nAuto former value:\n");

    for (i = 0; i <= 2; i++)
        printf("array2[%d] = %d  ", i, array2[i]);

    printf("\nAuto latter value:\n");

    for (i = 0; i <= 2; i++)
        printf("array2[%d] = %d  ", i, array2[i]+=5);
}

```

P25

main.c

```c
#include <stdio.h>
#include <stdlib.h>

int add1(int);

int main()
{
    int x = 100;
    int y = add1(x);
    printf("x=%d\n", x);
    system ("pause");
}

int add1(int xref)
{
    xref++;
    printf("xref=%d\n", xref);
    return xref;
}
```

P26

main.c

```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    char array[5];
    printf("    array = %p\n&array[0] = %p\n   &array = %d", array, &array[0], &array);

    system ("pause");
    return 0;
}
```

P28

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 5

void modArray (int b[], int size);
void modElement(int e);

int main()
{
    int a[SIZE] = {0,1,2,3,4};
    int i;

    printf("Effects of passing entire array by reference:\n\n"
           "The values of the original array are:\n");

    for (i = 0; i < SIZE; i++)
    {
        printf("%3d", a[i]);
    }
    printf("\n");

    modArray(a, SIZE);
    printf("The values of the modified array are:\n");
    for (i = 0; i < SIZE; i++)
    {
        printf("%3d", a[i]);
    }
    printf("\n");

    printf("Effects of passing array element by value:\n\n"
           "The values a[3] is: %d\n", a[3]);

    modElement(a[3]);
    printf("The value of a[3] is: %d\n", a[3]);

    system ("pause");
    return 0;
}

void modArray(int b[], int size)
{
    int j;

    for (j = 0; j < size; j++)
    b[j] *= 2;
}

void modElement(int e)
{
    printf("Value in modElement is: %d\n", e *= 2);
}
```

P30

main.c

```c
#include <stdio.h>
#include <stdlib.h>

void inv (int*);

int main()
{
    int a[3] = {1,2,3};
    int i;

    for (i = 0; i < 3; i++)
    {
        printf("%d   ", a[i]);
    }
    printf("\n");

    inv(a);

    for (i = 0; i < 3; i++)
    {
        printf("%d   ", a[i]);
    }
    printf("\n");

    system ("pause");
    return 0;
}

void inv(int *b)
{
    int temp[3], i;
    for (i = 0; i < 3; i++)
        temp[2-i] = b[i];

    for (i = 0; i < 3; i++)
        b[i] = temp[i];
}
```

P37

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 100

int linear(const int array[], int key, int size);

int main()
{
    int a[SIZE];
    int x;
    int searchKey;
    int element;

    for (x = 0; x < SIZE; x++)
        a[x] = 2*x;

    printf("Enter integer search key:\n");
    scanf("%d", &searchKey);

    element = linear(a, searchKey, SIZE);

    if (element != -1)
    {
        printf("Found value in element %d\n", element);
    }

    else
    {
        printf("Value not found!\n");
    }

    system ("pause");
    return 0;
}

int linear(const int array[], int key, int size)
{
    int n;
    for (n = 0; n < size; ++n)
    {
        if (array[n] == key)
        {
            return n;
        }
    }
     return -1;
}
```

6.11

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 5

int main()
{
    int i, j, k, temp;
    int a[SIZE] = {26,63,7,81,5};

    for (i = 0; i < SIZE - 1; i++)
    {
        for (j = 0, k = 0; j < SIZE - 1 - i; j++)
        {
            if (a[j]>a[j+1])
            {
                temp = a[j];
                a[j] = a[j+1];
                a[j+1] = temp;
            }

            else
            {
                k++;
                if (k == SIZE - 1 - i) goto Bruh;
            }
        }
        printf("Loop %d: ", i);
        for (j = 0; j < 5; j++)
            printf("%4d", a[j]);
        printf("\n");
    }
    Bruh:
    system ("pause");
    return 0;
}
```

Conclusion:
The usage of arrays is very useful if you are using a lot of values. The usage of call by value, call by address, or even call by reference can simplify a program by a lot, especially when you no longer need to assign a function to a data just because you need to. Static arrays are also useful if you need to store values.

Code: https://github.com/AldrichWijaya/Homework.git