6421604781 จิรภัทร โพธิ์สร้อย

Lab 35

```
#include <stdio.h>
#include <time.h>

void random(int arr[5][5][5], int d_size, int h_size, int w_size);
void sort_3d_array(int arr[5][5][5], int d_size, int h_size, int w_size, int select);
int arr[5][5][5], sortArr[125], d_size = 0, h_size = 0, w_size = 0;
int index = 0, i = 0, j = 0, k = 0, select = 0;

main()
{
    char finish = 'y', run = 0;
    while (finish == 'y')
    {
        for (i = 0; i < d_size; i++)
        {
            for (j = 0; j < h_size; j++)
            {
                for (k = 0; k < w_size; k++)
                {
                    arr[i][j][k] = 0;
                    sortArr[index] = 0;
                    index++;
                }
            }
        }
        while (w_size <= 0 || w_size > 5)
        {
            printf("Please input Array Width (maximum = 5): ");
            scanf("%d", &w_size);
```

```c
        if (w_size == 0)
        {
            printf("\n\"Please enter array width  > 0\"\n");
        }
        if (w_size > 5)
        {
            printf("\n\"Array Width maximum = 5\"\n\n");
            w_size = -1;
        }
    }
    while (h_size <= 0 || h_size > 5)
    {
        printf("Please input Array Height size (maximum = 5): ");
        scanf("%d", &h_size);
        if (h_size == 0)
        {
            printf("\n\"Please enter height size > 0\"\n");
        }
        if (h_size > 5)
        {
            printf("\n\"Array Height maximum = 5\"\n\n");
            h_size = -1;
        }
    }
    while (d_size <= 0 || d_size > 5)
    {
        printf("Please input Array Depth (maximum = 5): ");
        scanf("%d", &d_size);
        if (d_size == 0)
        {
            printf("\n\"Please enter depth  > 0\"\n");
        }
        if (d_size > 5)
        {
            printf("\n\"Array Depth maximum = 5\"\n\n");
            d_size = -1;
        }
    }
    while (select != 1 && select != 2)
    {
        printf("\n\"Min -> Max(1) or Max -> Min(2)\" (Enter 1 or Enter 2) : ");
        scanf("%d", &select);
    }
    printf("\n");
    random(arr,d_size, h_size, w_size);
    for(i=0;i<100;i++)
                {
                            printf("-");
                }
                printf("\n\n");

    sort_3d_array(arr, d_size, h_size, w_size, select);
    run = 1;
    while (run == 1)
    {
        printf("\nContinue Program ? (y/N) : ");
        scanf(" %c", &finish);
        if (finish == 'y' || finish == 'N')
        {
            run = 0;
```

```
                index = 0;
                d_size = 0;
                h_size = 0;
                w_size = 0;
                select = 0;
                printf("\n");
            }
            else
            {
                printf("Enter only \" y \" or \"N\"");
            }
        }
        if (finish == 'N')
        {
            printf("\"End Program\"");
        }
    }
}

void random(int arr[5][5][5], int d_size, int h_size, int w_size)
{
          srand(time(NULL));
          printf("\"Before Sort\"\n");
          for (i = 0; i < d_size; i++)
          {
             for (j = 0; j < h_size; j++)
             {
                    for (k = 0; k < w_size; k++)
                  {
                     arr[i][j][k] = rand() % 200 + 1;
                     printf("Array[%d][%d][%d] : %d\n", i, j, k, arr[i][j][k]);
                     sortArr[index] = arr[i][j][k];
                     index++;
                   }
             }
    }
}

void sort_3d_array(int arr[5][5][5], int d_size, int h_size, int w_size, int select)
{
    int tmp = 0;
    printf("\"After Sort\"\n");
    switch (select)
    {
    case 1:
        for (i = 0; i < w_size * h_size * d_size; i++)
        {
            for (j = 0; j < w_size * h_size * d_size; j++)
            {
                if (sortArr[j] > sortArr[j + 1])
                {
                    tmp = sortArr[j];
                    sortArr[j] = sortArr[j + 1];
                    sortArr[j + 1] = tmp;
                }
            }
        }
        break;
    case 2:
        for (i = 0; i < w_size * h_size * d_size; i++)
```

```
            {
                for (j = 0; j < w_size * h_size * d_size; j++)
                {
                    if (sortArr[j] < sortArr[j + 1])
                    {
                        tmp = sortArr[j + 1];
                        sortArr[j + 1] = sortArr[j];
                        sortArr[j] = tmp;
                    }
                }
            }
            break;
        default:
            break;
        }
        index = 0;
        for (i = 0; i < d_size; i++)
        {
            for (j = 0; j < h_size; j++)
            {
                for (k = 0; k < w_size; k++)
                {
                    arr[i][j][k] = sortArr[index];
                    index++;
                    printf("Array[%d][%d][%d] : %d \n", i, j, k, arr[i][j][k]);
                }
            }
        }
    }
```