A screen shot of a computer

Description automatically generated

#*include* <*bits/stdc++.h*>

*using* *namespace* std;

// *Câu 1. Hàm nhập / xuất mảng*

*void* nhapXuatMangRandom(*int arr*[], *int &n*){

    cout *<<* "*Hay nhap so phan tu ban muon random:* "; cin *>>* *n*;

    srand((*int*)time(0));

*for* (*int* i = 0; i < *n*; i++) {

*arr*[i] = rand() % 100 + 1;// *Random từ 1 -> 100*

    }

    cout *<<* "*1. Mang random la ->* ";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

*void* swap(*int &a*, *int &b*){// *Hàm hoán đổi vị trí*

*int* temp = *a*;

*a* = *b*;

*b* = temp;

}

// *Câu 2. Hàm sắp xếp tăng dần theo pp đổi chỗ trực tiếp*

*void* sapXepMangTangDanTrucTiep(*int arr*[], *int n*){// *InterchangeSort*

*for* (*int* i = 0; i < *n* - 1; i++) {

*for* (*int* j = i + 1; j < *n*; j++) {

*if* (*arr*[i] > *arr*[j])

                swap(*arr*[i], *arr*[j]);

        }

    }

    cout *<<* "*2. Mang tang dan theo pp doi cho truc tiep:* ";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

// *Câu 3. Hàm sắp xếp giảm dần theo pp đổi chỗ trực tiếp*

*void* sapXepMangGiamDanTrucTiep(*int arr*[], *int n*){// *InterchangeSort*

*for* (*int* i = 0; i < *n* - 1; i++) {

*for* (*int* j = i + 1; j < *n*; j++) {

*if* (*arr*[i] < *arr*[j])

                swap(*arr*[i], *arr*[j]);

        }

    }

    cout *<<* "*3. Mang giam dan theo pp doi cho truc tiep:* ";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

// *Câu 4. Hàm sắp xếp tăng dần theo pp chọn trực tiếp*

*void* sapXepTangDanChonTrucTiep(*int arr*[], *int n*){// *Selection Sort*

*for* (*int* i = 0; i < *n* - 1; i++) {

*int* min\_idx = i;

*for* (*int* j = i + 1; j < *n*; j++) {

*if* (*arr*[min\_idx] > *arr*[j])

                min\_idx = j;

        }

        swap(*arr*[i], *arr*[min\_idx]);

    }

    cout *<<* "*4. Mang tang dan theo pp chon truc tiep:* ";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

// *Câu 5. Hàm sắp xếp giảm dần theo pp chọn trực tiếp*

*void* sapXepGiamDanChonTrucTiep(*int arr*[], *int n*){// *Selection Sort*

*for* (*int* i = 0; i < *n* - 1; i++) {

*int* min\_idx = i;

*for* (*int* j = i + 1; j < *n*; j++) {

*if* (*arr*[min\_idx] < *arr*[j])

                min\_idx = j;

        }

        swap(*arr*[i], *arr*[min\_idx]);

    }

    cout *<<* "*5. Mang giam dan theo pp chon truc tiep:* ";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

// *Câu 6. Sắp xếp mảng tăng dần theo phương pháp chèn trực tiếp.*

*void* InsertionSort(*int arr*[], *int n*) {

*for* (*int* i = 1; i < *n*; i++) {

*int* key = *arr*[i];

*int* j = i - 1;

*while* (j >= 0 && *arr*[j] > key) {

*arr*[j + 1] = *arr*[j];

            j = j - 1;

        }

*arr*[j + 1] = key;

    }

    cout *<<* "*6. Mang tang dan theo pp chen truc tiep:*";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

*void* BubbleSort(*int arr*[], *int n*){

*bool* daThayDoi = *false*;// *Đặt biến daThayDoi để check xem có hoán đổi vị trí hay chưa*

*for* (*int* i = 0; i < *n* - 1; i++){// *Đặc biến đếm chạy đến cuối dãy (n-1)*

*for* (*int* j = 0; j < *n* - i - 1; j++)// *Duyệt từ j = 0 và cho chạy đến giới hạn là n - i - 1 Ví dụ Max n = 5*

            {// *thì 5-0-1 = 4 => chỉ chạy đến tối đa là arr[4] vì arr[5] đã là số lớn nhất và không cần check*

*if* (*arr*[j] > *arr*[j + 1]){// *nếu j lớn hơn j + 1 thì swap lại với nhau*

                swap(*arr*[j], *arr*[j + 1]);

                daThayDoi = *true*;// *Cho daThayDoi thành true để tránh việc lặp lại*

            }

        }

*if* (daThayDoi = *false*){

*break*;

        }

    }

    cout *<<* "*7. Mang tang dan theo pp noi bot:*";

*for* (*int* i = 0; i < *n*; i++) {

        cout *<<* *arr*[i] *<<* "";

    }

    cout *<<* endl;

}

*int* main(){

*int* n, arr[100];

    nhapXuatMangRandom(arr, n);

    sapXepMangTangDanTrucTiep(arr, n);

    sapXepMangGiamDanTrucTiep(arr, n);

    sapXepTangDanChonTrucTiep(arr, n);

    sapXepGiamDanChonTrucTiep(arr, n);

    InsertionSort(arr, n);

    BubbleSort(arr, n);

*return* 0;

}