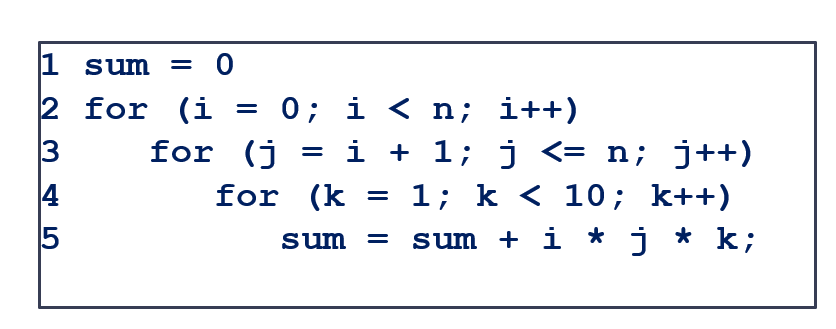
MSSV: 24521942

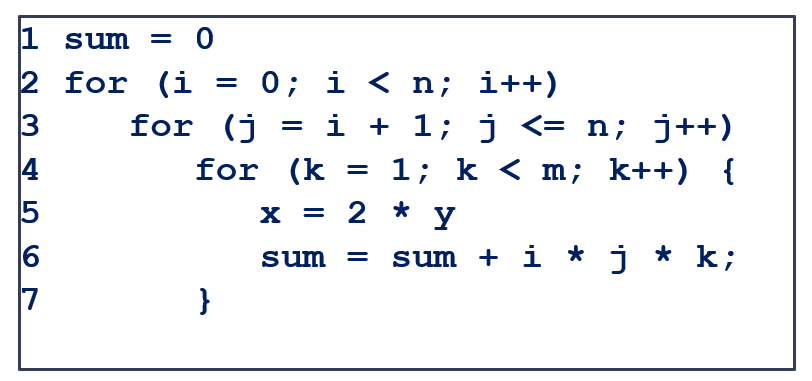
Họ và tên: Tô Đặng Minh Tuấn

Bài 1:

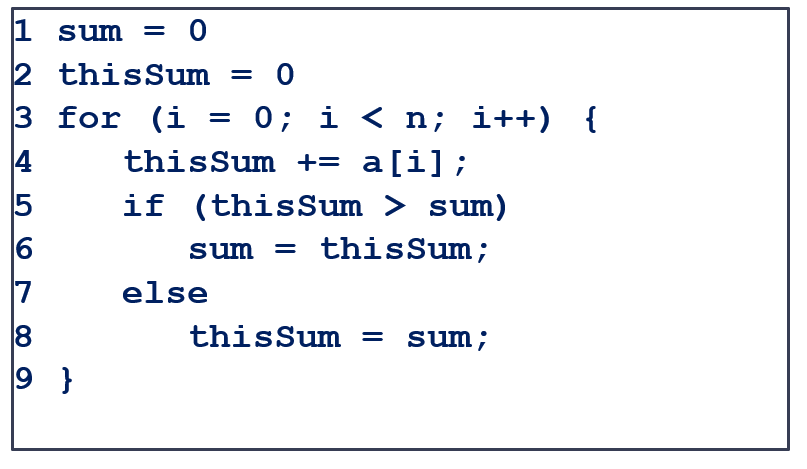
a.



b.



c.



Bài 2:

| #include <iostream> using namespace std;   void init(int a[100], int n) {  for(int i = 0; i < n; i++) {  cin >> a[i];  } }  long solve(int &start, int &stop, int a[100], int n) {  start = 0;  stop = 0;  long maxProduct = a[0];  for(int i = 0; i < n; i++) {  int tmp = 1;  for(int j = i; j < n; j++) {  tmp \*= a[j];  if(tmp > maxProduct) {  maxProduct = tmp;  start = i;  stop = j;  }  }  }  return maxProduct; }  void display(int a[], int start, int stop) {  for(int i = start; i <= stop; i++) {  cout << a[i] << " ";  }  cout << endl; }   int main() {  int a[100], n;  cin >> n;  init(a, n);  int start, stop, maxProduct;  maxProduct = solve(start, stop, a, n);  cout << "max product of sub array" << maxProduct << endl;  cout << "sub array create max product ";  display(a, start, stop);  return 0; } |
| --- |

Bài 3:

Merge sort hoạt động theo nguyên tắc chia để trị. Merge sort chia nhỏ nhiều lần một mảng (array) thành hai mảng con bằng nhau cho đến khi mỗi mảng con (subarray) bao gồm một phần tử duy nhất. Sau đó, các mảng con thành phần được ghép lại thành mảng duy nhất.

Bài 4.

Giải thuật thích hợp để giải quyết bài toán này là quay lui.

Khi xuất hiện các tổ hợp thỏa yêu cầu sẽ được ghi nhận vào biến count và lưu lại tổ hợp trong mảng 2 chiều theo từng trường hợp

Cài đặt C++:

| #include <iostream> using namespace std;  void Solve(int Arr[], int n, int rSum, int finalC[][100], int &resCnt, int current[], int &currentSize, int start) {  if(rSum < 0) {  return;  }  if(rSum == 0) {  for(int i = 0; i < currentSize; i++) {  finalC[resCnt][i] = current[i];  }  finalC[resCnt][currentSize] = -1;  resCnt++;  return;  }  for(int index = start; index < n; index++) {  int currentPoint = Arr[index];  current[currentSize++] = currentPoint;  Solve(Arr, n, rSum - currentPoint, finalC, resCnt, current, currentSize, index);  currentSize--;  } }  void fSolve(int Arr[], int n, int Sum) {  int finalc[100][100];  int resCnt = 0;  int current[100];  int currentSize = 0;   Solve(Arr, n, Sum, finalc, resCnt, current, currentSize, 0);   cout << "to hop de tao ra sum " << Sum << " : " << endl;  for(int i = 0; i < resCnt ; i++) {  cout << "[";  for(int j = 0; finalc[i][j] != -1; j++) {  cout << finalc[i][j] << " ";  }  cout << "]" << endl;  } }  int main() {  int Arr[100];  int n; cin >> n;  for(int i = 0; i < n; i++) {  cin >> Arr[i];  }  int sum; cin >> sum;  fSolve(Arr, n, sum);  return 0; } |
| --- |