**CTT12 – KỸ THUẬT LẬP TRÌNH**

**BÀI TẬP VỀ NHÀ**

**BTVN - 05: ĐỆ QUI**

Họ và tên: Huỳnh Tấn Thọ

Lớp: 19CTT2B

MSSV: **19120383**

**Bài 1:** Cài đặt bài toán tháp Hà Nội. So sánh thời gian thực hiện khi số đĩa tăng lên (N = 3, 5, 7, 9, 11, 13).

Đoạn chương trình dùng để chạy:

|  |
| --- |
| void towerOfHanoi(int n, char A, char C, char B)  {  if (n == 1)  {  cout << "Chuyen dia so 1 tu " << A << " sang cot " << C << endl;  return;  }  towerOfHanoi(n - 1, A, B, C);  cout << "Chuyen dia so " << n << " tu cot " << A << " sang cot " << C << endl;  towerOfHanoi(n - 1, B, C, A);  } |

Bảng kết quả:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Số đĩa (N) | Thời gian chạy và in kết quả ra màn hình (tính theo giây - seconds) | | | | |
| Lần 1 | Lần 2 | Lần 3 | Lần 4 | Lần 5 |
| 3 | 0.002 | 0.002 | 0.005 | 0.002 | 0.001 |
| 5 | 0.007 | 0.006 | 0.010 | 0.042 | 0.046 |
| 7 | 0.086 | 0.086 | 0.091 | 0.155 | 0.125 |
| 9 | 0.418 | 0.431 | 0.403 | 0.439 | 0.420 |
| 11 | 1.400 | 1.483 | 1.437 | 1.477 | 1.439 |
| 13 | 4.896 | 5.091 | 5.128 | 5.039 | 5.053 |

**Bài 2:** Cài đặt bài toán 8 hậu. So sánh thời giạn thực hiện khi kích thức bàn cờ và số hậu tăng lên

(8x8, 10x10, 12x12, 14x14, 16x16)

Đoạn chương trình dùng để chạy:

|  |
| --- |
| int a[100];  bool ktra(int x, int y) //kiem tra co dat duoc tai vi tri (x,y) khong  {  for (int i = 1; i <= x - 1; ++i)  {  if (a[i] == y) return false; //kiem tra cung cot  if (abs(i - x) == abs(a[i] - y)) return false; //kiem tra cung duong cheo  }  return true;  }  void try(int i, int n)  {  for (int j = 1; j <= n; ++j)  {  if (ktra(i, j))  {  a[i] = j;  if (i == n)  {  for (int i = 1; i <= n; ++i) cout << a[i] << " ";  cout << endl;  }  try(i + 1, n);  }  }  } |

Bảng kết quả

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Kích thước (N x N) | Thời gian chạy và in tất cả kết quả ra màn hình (tính theo giây - seconds) | | | | |
| Lần 1 | Lần 2 | Lần 3 | Lần 4 | Lần 5 |
| 8x8 | 0.153 | 0.102 | 0.110 | 0.091 | 0.096 |
| 10x10 | 1.209 | 1.151 | 1.188 | 1.319 | 1.091 |
| 12x12 | 19.484 | 19.186 | 19.517 | 20.136 | 20.186 |
| 14x14 | 462.051 | 460.959 | 461.063 | 469.128 | 463.525 |
| 16x16 | Thời gian chạy quá lâu (hơn 3 giờ) | | | | |

**Bài 3:** Cài đặt bài toán mã đi tuần. So sánh thời giạn thực hiện khi kích thức bàn cờ lên ( 8x8,

10x10, 12x12, 14x14, 16x16)

Đoạn chương trình dùng để chạy:

|  |
| --- |
| int KQ[16][16] = { 0 };  int p[8] = { -2,-2,-1,-1, 1, 1, 2, 2 };  int q[8] = { -1, 1,-2, 2,-2, 2,-1, 1 };  int d(0), k(0);  void xuat(int n)  {  for (int i = 0; i < n; ++i)  {  for (int j = 0; j < n; ++j) cout << KQ[i][j] << " ";  cout << endl;  }  cout << endl;  }  void diChuyen(int n, int x, int y)  {  d += 1;  KQ[x][y] = d;  for (int i = 0; i < n; ++i)  {  if (d == n \* n) //dem du so luong  {  cout << "CAC BUOC DI: \n";  xuat(n);  return;  }  int u, v;  u = x;  v = y;  for (int j = 0; j < 8; ++j)  {  u += p[i]; //di chuyen quan Ma  v += q[i];  if (u >= 0 && u < n && v >= 0 && v < n && KQ[u][v] == 0) //kiem tra vi tri (u,v) hop le  diChuyen(n, u, v);  }  }  d -= 1;  KQ[x][y] = 0;  } |

Bảng kết quả

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Kích thước (N x N) | Thời gian chạy và in 1 kết quả ra màn hình (tính theo giây - seconds) | | | | |
| Lần 1 | Lần 2 | Lần 3 | Lần 4 | Lần 5 |
| 8x8 | 1.125 | 1.233 | 1.003 | 1.452 | 1.186 |
| 10x10 | Thời gian chạy quá lâu (hơn 1 giờ) | | | | |
| 12x12 | Thời gian chạy quá lâu | | | | |
| 14x14 | Thời gian chạy quá lâu | | | | |
| 16x16 | Thời gian chạy quá lâu | | | | |

Nhận xét chung:

* Thời gian chạy giữa các lần thử của cùng một bài toán, với các tham số đầu vào như nhau, là khác nhau. Cụ thể:
* Đối với các bài có thời gian chạy rất nhỏ (dưới 1 giây) thì thời gian chênh lệnh giữa các lần chạy thử có thể hơn kém nhau nhiều lần, như bài 1 với N = 5 chênh lệch có thể hơn 4 lần.
* Đối với các bài có thời gian chạy tương đối dài, thì độ chênh lệch không đáng kể so với toàn bộ thời gian chạy, ví dụ như bài 2 ở kích thước 14x14 chỉ chênh lệch vài giây so với hơn 460 giây.
* Khi tăng kích thước hoặc số lượng đầu vào ở mỗi bài, thì thời gian chạy tăng lên khoảng từ 2 đến 10 lần, thậm chí nhiều hơn 20 lần.
* Một số bài có thời gian chạy quá lâu, ví dụ như bài 2 ở kích thước 16x16, hoặc bài 3 ở kích thước 10x10 trở đi.
* Bài 3 có thời gian chênh lệch giữa các kích thước là rất lớn, cụ thể 8x8 chỉ mất 1 giây để in ra 1 kết quả thỏa mãn yêu cầu, nhưng khi tăng lên 10x10 thì thậm chí sau 1 giờ vẫn không có kết quả.
* Thuật toán đệ quy back-tracking không hẳn là tốt cho bài 3, nó chỉ giải quyết tốt các bài có kích cỡ từ 8x8 trở xuống, còn lớn hơn thì tốn rất nhiều thời gian, cần giải thuật tối ưu hơn cho bài này.