



TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC

**BÁO CÁO CUỐI KỲ
TÍNH TOÁN SONG SONG**

Giảng viên hướng dẫn

TS. Đoàn Duy Trung

Nhóm báo cáo

Nhóm 05

Thành viên:

- | | |
|---------------------|----------|
| 1. Nguyễn Hữu Thuật | 20185410 |
| 2. Trần Xuân Hiếu | 20185354 |
| 3. Lại Tiến Long | 20185376 |
| 4. Trang Hải Long | 20185382 |

Hà Nội, Tháng 06 năm 2020

Lời cảm ơn

Chúng Em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đối với Thầy Đoàn Duy Trung - giảng viên học phần Tính toán song song đã vô cùng nhiệt tình giảng dạy Chúng Em trong suốt học kỳ qua, cũng như đã hướng dẫn đề tài này để Nhóm có thể hoàn thành tốt nhất bài báo cáo cũng như học phần này.

Trong quá trình học tập, báo cáo cũng như làm bài báo cáo khó tránh khỏi những sai sót, rất mong Thầy bỏ qua. Đồng thời cũng do trình độ lý luận cũng như kinh nghiệm thực tiễn còn hạn chế nên bài báo cáo không thể tránh khỏi những thiếu sót, Chúng Em rất mong nhận được ý kiến đóng góp Thầy để Chúng Em học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn các bài báo cáo sắp tới.

Chúng Em xin chân thành cảm ơn!

Phân công công việc

Việc đưa ra mục phân công công việc là tương đối, vì trước khi tiến hành viết chương trình, các thành viên nhóm đã cùng hội ý để: hiểu đề bài, đưa ra ý tưởng và thuật toán sơ bộ cho chương trình.

Dưới đây là phần phân công công việc của nhóm:

i) **Nguyễn Hữu Thuật:**

- Phân công, tổng hợp công việc;
- Làm bài 1 (phần bài tập chung);
- Cùng viết chương trình bài tập riêng;
- Làm báo cáo.

ii) **Trần Xuân Hiếu:**

- Cùng viết chương trình bài 2 (phần bài tập chung);
- Kiểm tra, chỉnh sửa chương trình bài tập riêng.

iii) **Lại Tiến Long:**

- Cùng viết chương trình bài 2 (phần bài tập chung);
- Kiểm tra, chỉnh sửa chương trình bài tập 1 (phần bài tập chung);
- Chạy chương trình lấy kết quả báo cáo.

iv) **Trang Hải Long:**

- Viết chính chương trình bài tập riêng;
- Kiểm tra, chỉnh sửa chương trình bài tập 2 (phần bài tập chung).

Mục lục

1	Bài tập chung	4
2	Bài tập riêng	10
2.1	Chương trình	10
2.2	Kết quả	13
2.3	Alchemi Console	13

1 Bài tập chung

Bài 1 Liệt kê các số chính phương từ 1 đến n , với n nhập từ bàn phím. Phân chia đoạn $[1; n]$ thành các đoạn để chạy trên các executor trong Lưới.

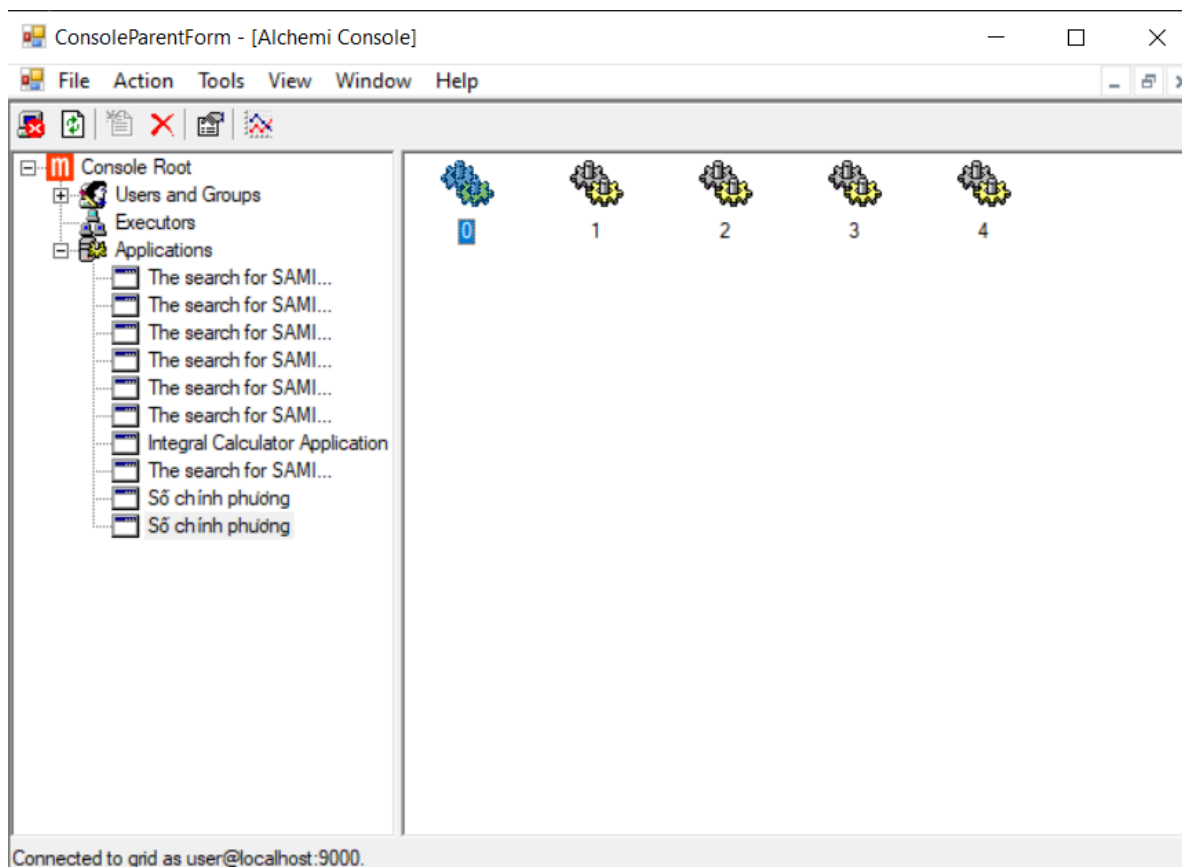
Kết quả:

Hình 1: Kết quả chạy chương trình với 5 luồng tham gia, mỗi luồng xử lý tối đa 2 số

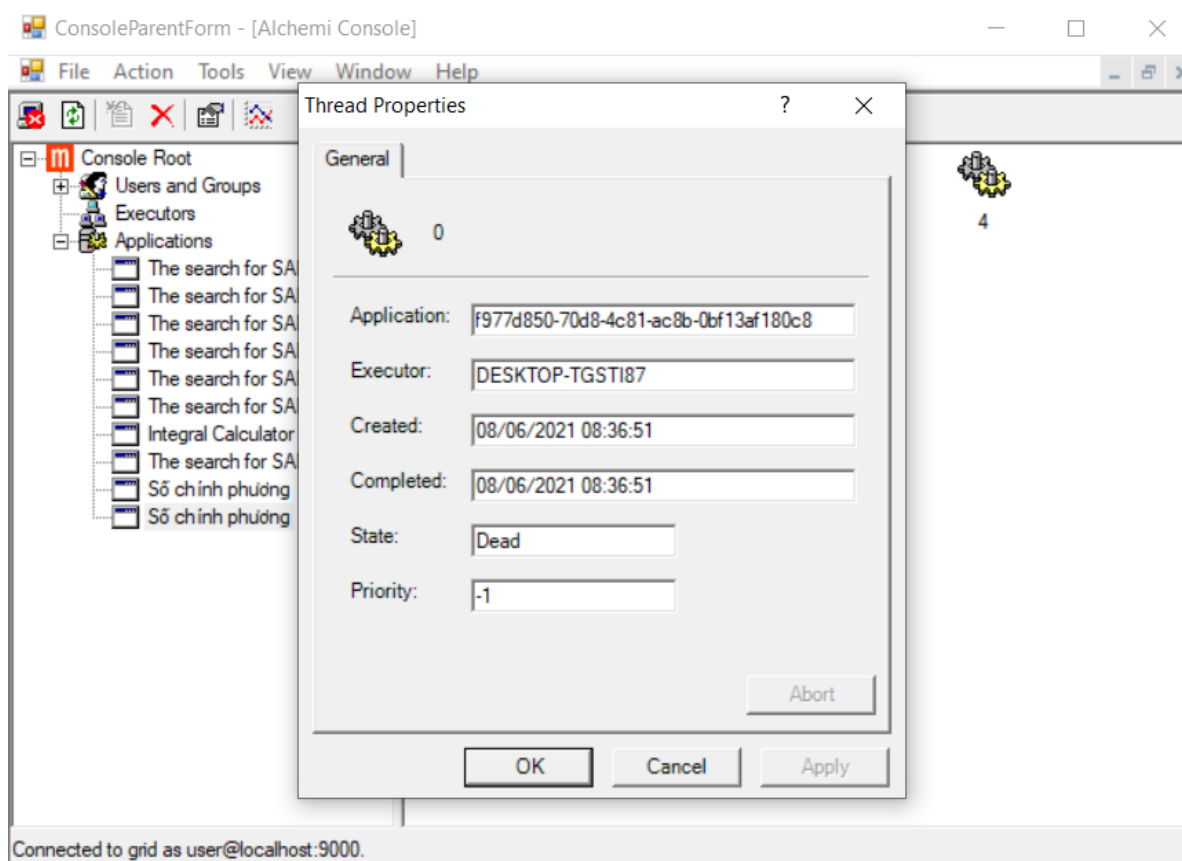
```
Host[localhost]:
Input your number : 10
Input your per of thread : 2
Thread Started
Thread 0 (1:2): 1
Thread 1 (3:4): 4
Thread 2 (5:6):
Thread 3 (7:8):
Thread 4 (9:10): 9
Calculation finished after 00:00:02.8998808 seconds
```

Alchemi Console:

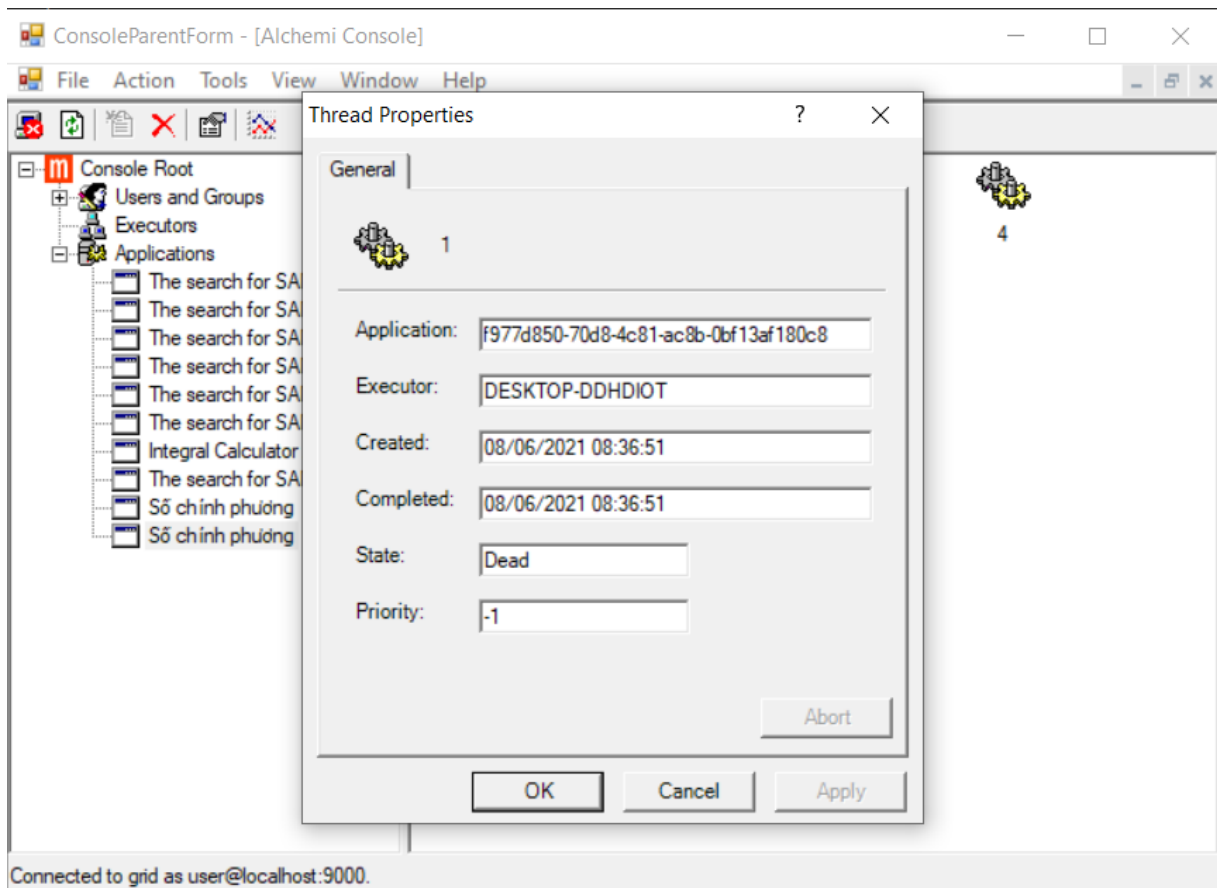
Hình 2: 5 luồng đã được 2 Executor là DESKTOP-DDHDIOT và DESKTOP-TGSTI87 thực hiện



Hình 3: Luồng 0 do DESKTOP-TGSTI87 thực hiện



Hình 4: Luồng 1 do DESKTOP-DDHDIOT thực hiện



Bài 2: Cho n nhập từ bàn phím. Tính gần đúng tích phân sau:

$$\int_0^n f(x)dx \quad (1)$$

Ở đó hàm $f(x)$ tùy ý;

Phân chia $[0; n]$ vào các Executor để chạy trong Lưới.

Kết quả:

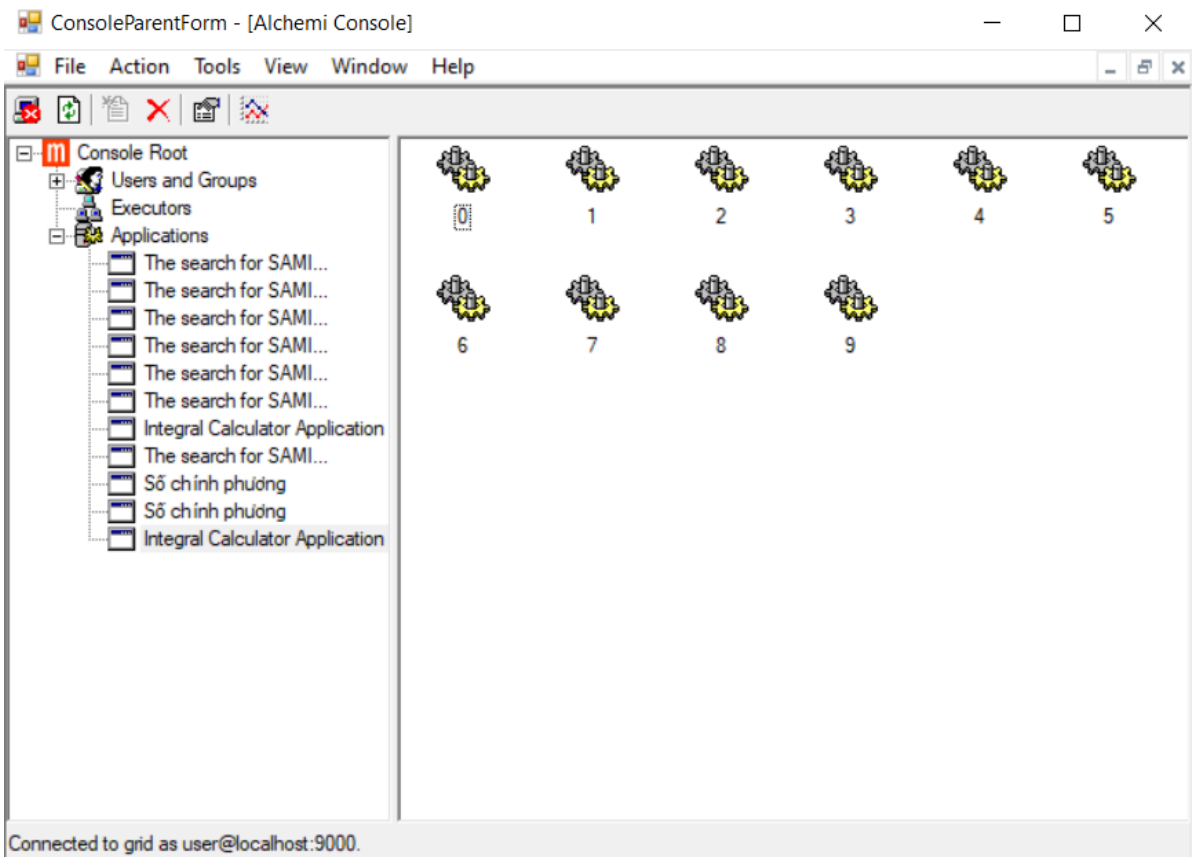
Hình 5: Kết quả chạy chương trình với 10 luồng tham gia

```
Host [default=localhost] :
Port [default=9000] :
Username [default=user] :
Password [default=user] :

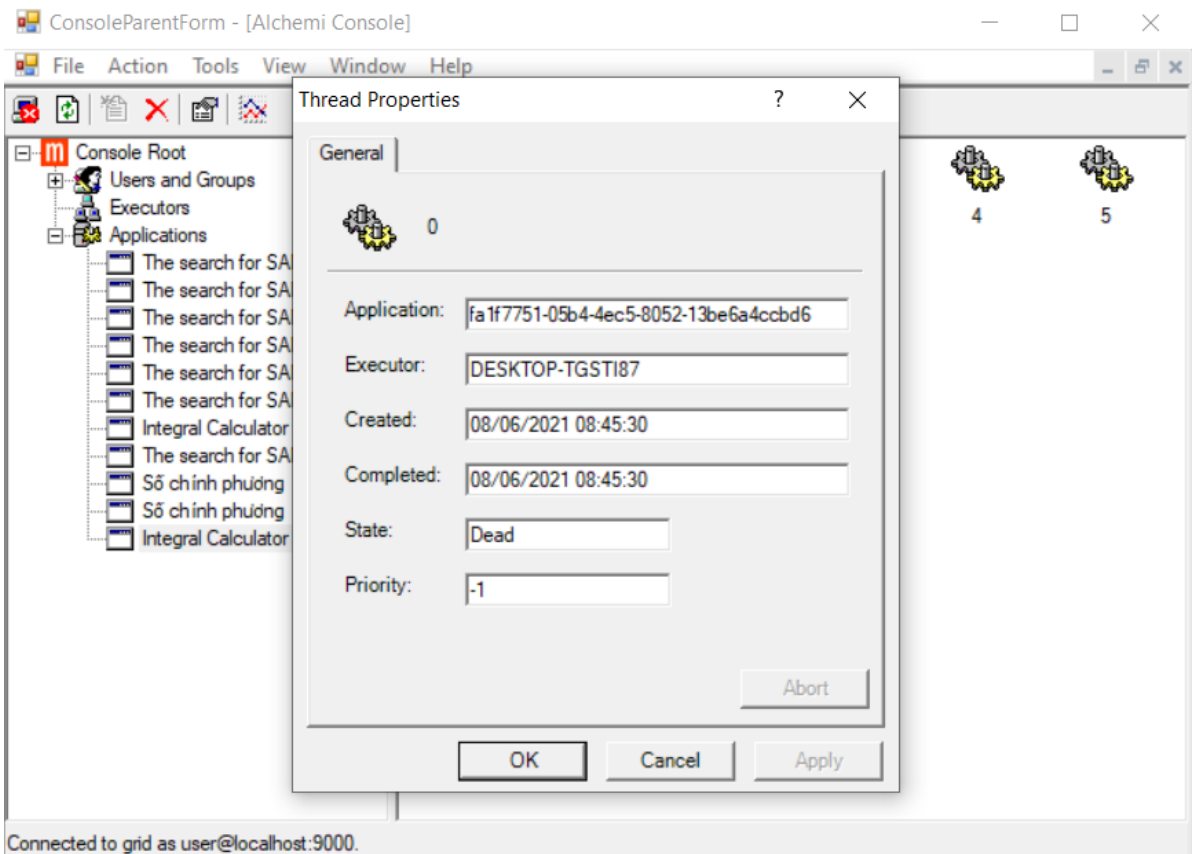
Thread 0 is finished
0.118538063633949
Thread 1 is finished
0.263294183914427
Thread 2 is finished
0.430371351262804
Thread 3 is finished
0.642681602408728
Thread 4 is finished
0.900863570239101
Thread 5 is finished
1.20880576023055
Thread 6 is finished
1.5676829860375
Thread 7 is finished
1.97822583858767
Thread 8 is finished
2.44092568508945
Thread 9 is finished
2.95613190411613
12.5075209455203
Appcation finished!!!
```

Alchemi Console:

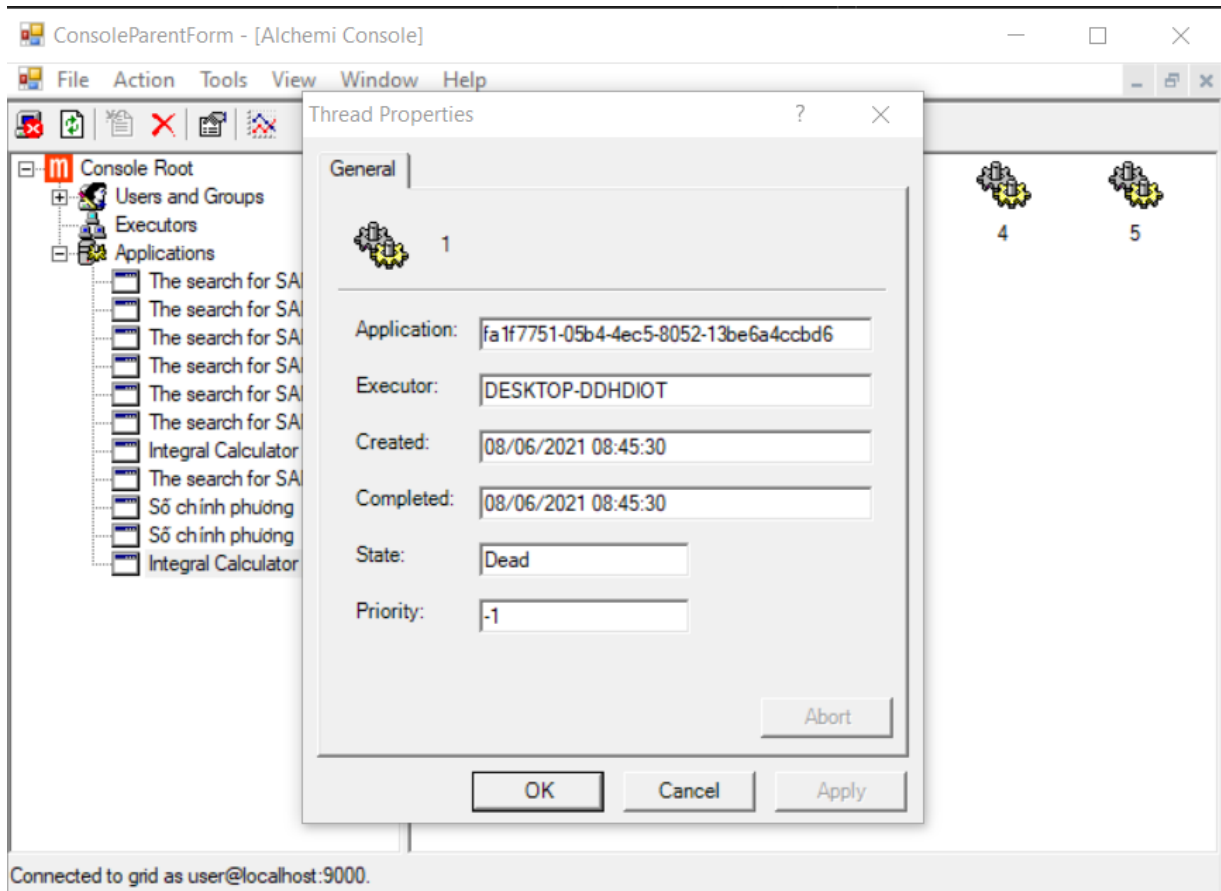
Hình 6: 10 luồng đã được 2 Executor là DESKTOP-DDHDIOT và DESKTOP-TGSTI87 thực hiện



Hình 7: Luồng 0 do DESKTOP-TGSTI87 thực hiện



Hình 8: Luồng 1 do DESKTOP-DDHDIOT thực hiện



2 Bài tập riêng

Đề 5: Đọc vào 1 đoạn văn bản dưới dạng file input.txt, tìm kiếm cụm từ SAMI xuất hiện đầu tiên trong 1 câu (1 câu kết thúc bởi dấu chấm). Phân chia văn bản thành nhiều k văn bản nhỏ hơn (k nhập từ bàn phím), mỗi văn bản nhỏ một vài câu. Yêu cầu mỗi văn bản nhỏ thực hiện trên 1 nút tính toán. Tổng hợp vị trí các cụm từ SAMI trên từng dòng vào file output.txt, nếu không có cụm từ SAMI thì dòng đó để NULL.

2.1 Chương trình

```
1 using System;
2 using System.Collections.Generic;
3 using Alchemi.Core.Owner;
4
5 namespace Parallel_Search_String
6 {
7     class Program
8     {
9         static GApplication App;
10
11         // Read the file as one string.
12         static string text = System.IO.File.ReadAllText(@"..\..\
            document.txt");
13         static List<string> sentences = new List<string>(text.Split
            ('. '));
14         static int NumberOfSentences = sentences.Count;
15         static int SentencesPerThread = 2;
16         static int NumThreads = 0;
17
18         [STAThread]
19         static void Main(string[] args)
20         {
21             // Connect to Manager
22             GConnection connection = GConnection.FromConsole("
                localhost", "9000", "user", "user");
23
24             // Create an application object that represents the
                application you are running
25             App = new GApplication(connection); // Initialization
26             App.ApplicationName = "The search for SAMI..."; //
                Assign the name of the application, this name will be
                displayed in the Alchemi Console window
27             App.Manifest.Add(new ModuleDependency(typeof(
                StringSearchGridThread).Module));
28
29             // Always the number of calculations to use
30             NumThreads = (Int32)Math.Floor(((double)NumberOfSentences
                / SentencesPerThread));
31             if (SentencesPerThread * NumThreads < NumberOfSentences)
32             {
33                 NumThreads++;
```

```

34         }
35
36         // Create and add streams
37         for (int i = 0; i < NumThreads; i++)
38         {
39             int StartIndex = i * SentencesPerThread;
40             int SentencesForThisThread = Math.Min(
31                 SentencesPerThread, NumberOfSentences - i *
32                 SentencesPerThread);
41
42             StringSearchGridThread thread = new StringSearchGridThread(sentences.
33                 GetRange(StartIndex, SentencesForThisThread));
43                 App.Threads.Add(thread);    // Add thread to the
34                 application
44         }
45
46         // Events
47         App.ThreadFinish += new GThreadFinish(OnThreadFinish);
48
49         // App starts running
50         App.Start();
51
52         Console.ReadLine();
53     }
54
55     // Event to end a thread
56     static public void OnThreadFinish(GThread th)
57     {
58         Console.WriteLine("Thread {0} finished the job", th.Id);
59         StringSearchGridThread thread = (StringSearchGridThread)
30             th;
60         foreach (int i in thread.Result)
61         {
62             if (i == -1)
63             {
64                 Console.WriteLine("NULL");
65             }
66             else
67             {
68                 Console.WriteLine(i);
69             }
70         }
71     }
72 }
73 [Serializable]
74 class StringSearchGridThread : GThread
75 {
76
77     private List<int> _Result = new List<int>();
78     private int _StartIndex;
79     private int _NumSentences;
80     private List<string> _Sentences = new List<string>();
81
82     // Decorator
83     public List<int> Result
84     {
85         get
86         {

```

```

87         return _Result;
88     }
89 }
90
91 public List<string> Sentences
92 {
93     get
94     {
95         return _Sentences;
96     }
97 }
98 public int StartIndex
99 {
100     get
101     {
102         return _StartIndex;
103     }
104 }
105
106 public int NumSentences
107 {
108     get
109     {
110         return _NumSentences;
111     }
112 }
113
114 // Constructor
115 public StringSearchGridThread(List<string> Sentences)
116 {
117     _Sentences = Sentences;
118 }
119 public override void Start()
120 {
121     for (int i = 0; i < _Sentences.Count; i++)
122     {
123         _Result.Add(_Sentences[i].IndexOf("SAMI", 0));
124     }
125 }
126 }
127 }

```

File Document.txt

¹ Institute of Applied Mathematics and Informatics (SAMI), Hanoi University of Science and Technology, is a prestigious undergraduate and graduate research and training institution in the field of Mathematics and Informatics. SAMI... The Institute's main tasks are: Teaching and researching mathematics, applied mathematics and informatics; Conduct research and teaching cooperation with domestic and foreign training institutions; Coordinating with branches, levels and enterprises in applying mathematics and informatics in fields such as economics, finance, construction, engineering, etc. SAMI. Institute of Applied Mathematics and Informatics (SAMI), Hanoi University of Science

and Technology, is a prestigious undergraduate and graduate research and training institution in the field of Mathematics and Informatics. SAMI... The Institute's main tasks are: Teaching and researching mathematics, applied mathematics and informatics; Conduct research and teaching cooperation with domestic and foreign training institutions; Coordinating with branches, levels and enterprises in applying mathematics and informatics in fields such as economics, finance, construction, engineering, etc. SAMI. Institute of Applied Mathematics and Informatics (SAMI), Hanoi University of Science and Technology, is a prestigious undergraduate and graduate research and training institution in the field of Mathematics and Informatics. SAMI... The Institute's main tasks are: Teaching and researching mathematics, applied mathematics and informatics; Conduct research and teaching cooperation with domestic and foreign training institutions; Coordinating with branches, levels and enterprises in applying mathematics and informatics in fields such as economics, finance, construction, engineering, etc. SAMI.

2.2 Kết quả

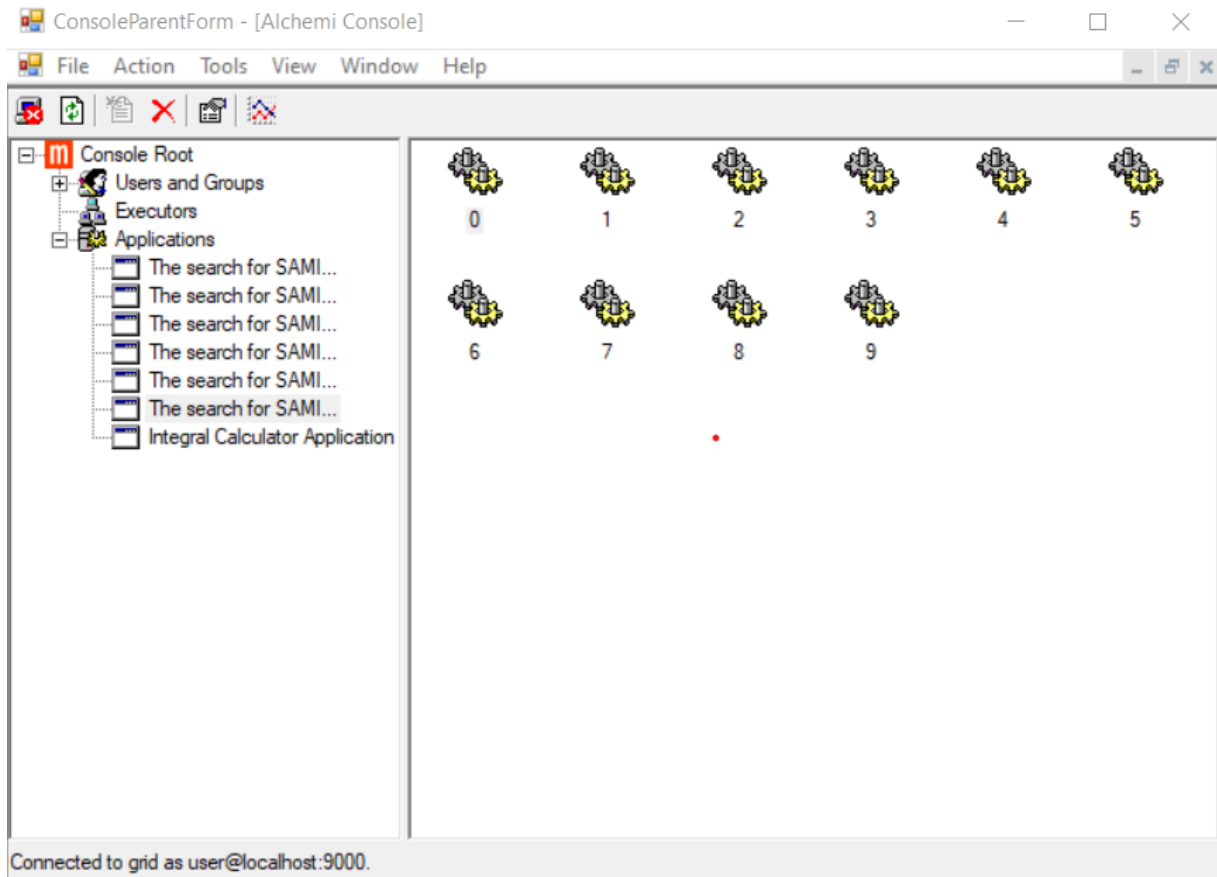
Hình 9: Kết quả chạy chương trình với 10 luồng tham gia, mỗi luồng xử lý tối đa 2 câu

```
Port [default=9000] :
Username [default=user] :
Password [default=user] :

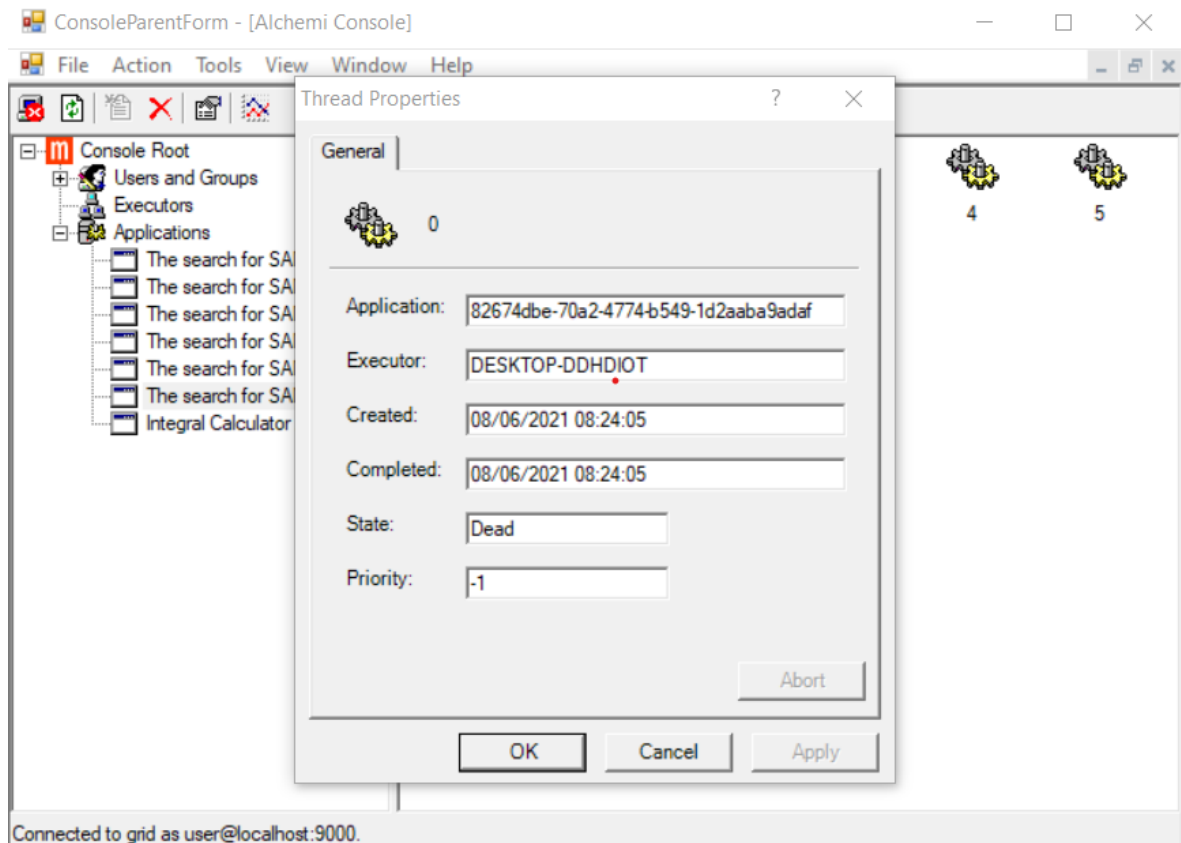
Thread 0 finished the job
50
1
Thread 1 finished the job
NULL
NULL
Thread 2 finished the job
NULL
0
Thread 3 finished the job
50
1
Thread 4 finished the job
NULL
NULL
Thread 5 finished the job
NULL
0
Thread 6 finished the job
50
1
Thread 7 finished the job
NULL
NULL
Thread 8 finished the job
NULL
0
Thread 9 finished the job
NULL
```

2.3 Alchemi Console

Hình 10: 10 luồng đã được 2 Executor là DESKTOP-DDHDIOT và DESKTOP-TGSTI87 thực hiện



Hình 11: Luồng 0 do DESKTOP-DDHDIOT thực hiện



Hình 12: Luồng 1 do DESKTOP-TGSTI87 thực hiện

