**CHƯƠNG TRÌNH MÔN HỌC**

**Tên môn học: THỰC TẬP KỸ THUẬT LẬP TRÌNH**

**Mã môn học: 229149**

**Thời gian thực hiện môn học**: 30 giờ; (Lý thuyết: 00 giờ; thực hành, thí nghiệm, thảo luận, bài tập: 29 giờ; kiểm tra 01 giờ).

1. **Vị trí, tính chất của môn học**

* Vị trí: học phần này học ở học kỳ 2
* Tính chất: học phần này học sau môn học Nhập môn lập trình - 229038, đây là học phần bắt buộc (học phần này phải học trước học phần CTDL&GT và lập trình hướng đối tượng nếu có).

1. **Mục tiêu môn học**

Sau khi tiếp nhận kiến thức về lập trình trong học phần nhập môn lập trình như: khái niệm tổng quát về ngôn ngữ lập trình, cũng như các thành phần cơ bản, hàm nhập/xuất, các cấu trúc điều khiển của một ngôn ngữ lập trình cấp cao (C/C++), cách tiếp cận bài toán bằng việc xây dựng hàm, cấu trúc chuỗi ký tự, cấu trúc mảng 1 chiều.

Trong học phần này, chúng ta tiếp tục làm rõ hơn các bài toán trên liên quan tới con trỏ, cấu trúc mảng 2 chiều, kiểu dữ liệu tự định nghĩa (struct), phương pháp lập trình đệ quy, phương pháp lập trình tập tin, hay những bài toán nâng cao hơn.

* Về năng lực tự chủ và trách nhiệm
* Nhận thức được tầm quan trọng của học phần là kiến thức nền tảng để học tốt các học phần chuyên ngành tiếp theo.
* Nhận thức được tầm quan trọng của các cấu trúc mới trong việc giải quyết các bài toán trong lập trình.
* Nhận thức được vai trò của học phần đối với việc phát triển tư duy logic, tư duy lập trình để trở thành 1 lập trình viên chuyên nghiệp.
* Về kiến thức
* Xác định các bước để xây dựng chương trình cụ thể.
* Hiểu được các kỹ thuật lập trình trên kiểu dữ liệu mảng hai chiều, kiểu dữ liệu cấu trúc, con trỏ và tập tin.
* Hiểu và phân tích được các bài toán đệ quy.
* Phân biệt các phương pháp lập trình và cách tiếp cận bài toán.
* Về kỹ năng
* Phân tích, đánh giá và giải quyết bài toán theo nhiều cách tiếp cận khác nhau.
* Áp dụng các kỹ thuật để giải một số bài toán trong thực tế, từ đó có đánh giá ưu/khuyết điểm của từng giải thuật.
* Cài đặt được các thao tác trên kiểu dữ liệu mảng hai chiều, kiểu cấu trúc, con trỏ và kiểu tập tin.
* Cài đặt được các bài toán đệ quy cơ bản.

1. **Nội dung môn học**
2. **Nội dung tổng quát và phân bổ thời gian**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Số TT** | **Tên chương, mục** | **Thời gian (giờ)** | | | | |
| **Tổng số** | **Lý thuyết** | **Thực hành, thí nghiệm, thảo luận, bài tập** | **Kiểm tra** |
| 1 | 1. **Kỹ thuật lập trình với con trỏ**    1. Con trỏ và địa chỉ    2. Con trỏ với kiểu dữ liệu cấu trúc    3. Con trỏ với hàm    4. Bài tâp | 5 | 3 | 2 | 0 |
| 2 | 1. **Thực hành con trỏ**    1. Bài toán 01    2. Bài toán 02    3. Bài toán 03    4. Bài toán 04 | 5 | 0 | 5 | 0 |
| 3 | 1. **Thực hành cấu trúc ma trận**    1. Bài toán 01    2. Bài toán 02    3. Bài toán 03 | 5 | 0 | 5 | 0 |
| 4 | 1. **Thực hành kiểu dữ liệu tự định nghĩa**    1. Bài toán 01    2. Bài toán 02    3. Bài toán 03    4. Bài toán 04 | 5 | 0 | 5 | 0 |
| 5 | 1. **Thực hành lập trình đệ quy**    1. Bài toán 01    2. Bài toán 02    3. Bài toán 03 | 5 | 0 | 4 | 1 |
| 6 | 1. **Thực hành ứng dụng**     1. Bài toán 01    2. Bài toán 02    3. Bài toán 03    4. Bài toán 04 | 5 | 0 | 5 | 0 |
|  | **Cộng** | 30 | 3 | 26 | 1 |

1. **Nội dung chi tiết**
2. **Kỹ thuật lập trình với con trỏ Thời gian: 5 giờ**

***Mục tiêu:***

* Hiểu được bản chất của con trỏ;
* Sử dụng được cú pháp khai báo con trỏ, các phép toán trên con trỏ, sử dụng con trỏ trong NNLT C/C++;
* Ứng dụng con trỏ.
  1. **Con trỏ và địa chỉ**
     1. Khái niệm biến con trỏ
     2. Khai báo biến con trỏ
     3. Các phép toán trên con trỏ
     4. Thao tác dịch con trỏ
     5. Các hàm cấp phát và giải phóng bộ nhớ
     6. Một số nguyên tắc an toàn khi dùng con trỏ
  2. **Con trỏ với kiểu dữ liệu cấu trúc**
     1. Cú pháp khai báo và cách dùng
     2. Cấp phát và hủy mảng cấu trúc
  3. **Con trỏ với hàm**
     1. Ví dụ 01
     2. Ví dụ 02
  4. Bài tập
     1. Bài tập 01

Hãy cho biết trong đoạn chương trình dưới đây câu lệnh nào là đúng, câu lệnh nào là sai.

|  |
| --- |
| int a;  int \*p;  a = 5;  p = 7;  p=&a;  a++;  \*p++;  cout << "\na= "<<a;  cout << "\np= "<<&p; |

* + 1. Bài tập 02

Hãy cho biết trong đoạn chương trình dưới đây câu lệnh nào là đúng, câu lệnh nào là sai.

|  |
| --- |
| int a = 15;  int \*p, \*k;  float \*q;  q = &a;  p = &a;  p = a;  q = a;  k = p;  k = q; |

* + 1. Bài tập 03

Hãy cho biết kết quả thực hiện của đoạn chương trình dưới đây.

|  |
| --- |
| int a;  int \*p;  a = 305;  p = &a;  a++;  cout << "\n Gia tri cua bien a= " << a;  cout << "\n Gia tri tai dia chi cua bien p dang tro toi: " << \*p; |

* + 1. Bài tập 04

Hãy cho biết kết quả thực hiện của đoạn chương trình dưới đây.

|  |
| --- |
| int a;  int \*p;  a = 305;  p = &a;  a++;  \*p++;  cout << "\n Gia tri cua bien a= " << a;  cout << "\n Gia tri tai dia chi cua bien p dang tro toi: " << \*p; |

1. **Thực hành con trỏ Thời gian: 05 giờ**

***Mục Tiêu:***

* Hiểu được bản chất con trỏ
* Sử dụng được con trỏ trong một số bài toán cơ bản về lập trình
  1. **Bài toán 01**

Hãy viết đoạn chương trình để khai báo biến số nguyên a và xuất ra địa chỉ ô nhớ được cấp phát cho biến này khi chương trình chạy.

* 1. **Bài toán 02**

Hãy khai báo biến con trỏ p các số nguyên và viết lệnh cấp phát không gian bộ nhớ để biến con trỏ p có thể chứa 100 số nguyên. Sau đó viết lệnh thu hồi lại không gian bộ nhớ đã cấp phát cho con trỏ.

* 1. **Bài toán 03**

Viết hàm hoán vị 2 số nguyên bằng cách sử dụng kỹ thuật con trỏ.

* 1. **Bài toán 04**

Cho một mảng 1 chiều động n số nguyên. Anh/Chị viết chương trình để giải quyết các yêu cầu dưới đây.

* 1. Nhập/xuất mảng
  2. Tính tổng các phần tử
  3. Tính tổng các phần tử chẵn, âm, nguyên tố, chính phương
  4. Đếm số phần tử lẻ, bằng x
  5. Liệt kê các phần tử nguyên tố
  6. Liệt kê các phần tử lớn hơn phần tử đầu tiên
  7. Tìm max
  8. Tìm chẵn đầu
  9. Kiểm tra xem mảng có tồn tại số nguyên tố hay không
  10. Tìm max âm

1. **Thực hành cấu trúc ma trận Thời gian: 05 giờ**

***Mục Tiêu:***

* Cài đặt được các thao tác trên cấu trúc ma trận
* Ứng dụng được cấu trúc ma trận
  1. **Bài toán 01**

Cho ma trận có cấu trúc: float a[][MAX]; int m, n;

**Viết chương trình thực hiện các yêu cầu dưới đây** (Bạn nên phát triển từ chương trình mẫu Matran01, rồi sửa lại)

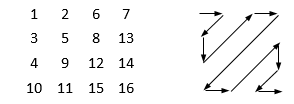
* 1. Nhập/xuất ma trận
  2. Tính tổng các phần tử trong ma trận
  3. Đếm số phần tử âm trong ma trận
  4. Đếm số phần tử âm trên dòng k
  5. Tính trung bình tất cả các phần tử trong ma trận
  6. Tính trung bình các phần tử trên cột k bất kỳ trong ma trận
  7. Liệt kê các phần tử có phần nguyên là một số nguyên tố
  8. Liệt kê các phần có phần nguyên là số chẵn
  9. Tính tổng các phần tử trên đường biên của ma trận (i=0, i=m-1, j=0, j=n-1)
  10. **Bài toán 02**

Cho một ma trận m\*n số nguyên kiểu int. Viết chương trình thực hiện các yêu cầu dưới đây *(Có thể phát triển từ chương trình mẫu Matran01)*:

1. Nhập ma trận ngẫu nhiên
2. Xuất ma trận
3. Tính tổng các phần tử trong ma trận có giá trị lớn hơn x.
4. Đếm số phần tử dương trên đường biên của ma trận.
5. Tìm phần tử lớn nhất không thuộc đường biên của ma trận.
6. Kiểm tra xem trong ma trận có tồn tại ít nhất 2 phần tử min hay không?
7. Liệt kê các cột có tồn tại số nguyên tố.
8. Xóa một cột k bất kỳ.
9. Thêm mới một dòng tại vị trí k, mà mỗi phần tử ở dòng đó bằng h.
   1. **Bài toán 03**

**Mô tả bài toán:** ZIZA - Zig Zag

Cho một ma trận vuông kích thước *n* x *n*, bạn hãy điền các số từ 1 đến *n*2 vào ma trận theo hình zigzag bắt đầu từ phía trên bên trái cho đến phía dưới bên phải. Ví dụ cho *n* = 4 thì ma trận zigzag là như sau:



**Dữ liệu nhập:**

Là số nguyên n (1 <= n <= 50)

**Dữ liệu xuất:**

Gồm *n* dòng biểu thị hình zig zag, mỗi dòng gồm *n* số nguyên, mỗi số cách nhau một khoảng trắng.

**Các test của bài toán**

* **Test #1, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 1084 KB**

Input

5

Output

1 2 6 7 15  
3 5 8 14 16  
4 9 13 17 22  
10 12 18 21 23  
11 19 20 24 25

Đáp án

1 2 6 7 15  
3 5 8 14 16  
4 9 13 17 22  
10 12 18 21 23  
11 19 20 24 25

* **Test #2, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 904 KB**

Input

3

Output

1 2 6  
3 5 7  
4 8 9

Đáp án

1 2 6  
3 5 7  
4 8 9

* **Test #3, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 824 KB**

Input

2

Output

1 2  
3 4

Đáp án

1 2  
3 4

* **Test #4, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 940 KB**

Input

8

Output

1 2 6 7 15 16 28 29  
3 5 8 14 17 27 30 43  
4 9 13 18 26 31 42 44  
10 12 19 25 32 41 45 54  
11 20 24 33 40 46 53 55  
21 23 34 39 47 52 56 61  
22 35 38 48 51 57 60 62  
36 37 49 50 58 59 63 64

Đáp án

1 2 6 7 15 16 28 29  
3 5 8 14 17 27 30 43  
4 9 13 18 26 31 42 44  
10 12 19 25 32 41 45 54  
11 20 24 33 40 46 53 55  
21 23 34 39 47 52 56 61  
22 35 38 48 51 57 60 62  
36 37 49 50 58 59 63 64

1. **Thực hành kiểu dữ liệu tự định nghĩa Thời gian: 05 giờ**

***Mục Tiêu:***

* Cài đặt được các bài toán cơ bản trên KDL tự định nghĩa
* Ứng dụng KDL tự định nghĩa
  1. **Bài toán 01**

Viết chương trình nhập vào 2 phân số từ bàn phím.

**Thực hiện các yêu cầu sau:**

* 1. Xuất 2 phân số ra màn hình
  2. Rút gọn chúng (gợi ý dùng hàm tìm ước số chung lớn nhất)
  3. Tính tổng, hiệu, thương, tích 2 phân số
  4. Viết hàm so sánh hai phân số bất kỳ
  5. **Bài toán 02**

Cho một đối tượng Hóa đơn gồm các thông tin:

* Mã hóa đơn (mahd): kiểu số nguyên
* Họ và tên khách hàng (hoten): chuỗi tối đa 51 ký tự
* Điện thoại (dt): chuỗi 10 ký tự
* Ngày lập hóa đơn (nglap): chuỗi 10 ký tự
* Tổng tiền (ttien): số thực, đơn vị triệu đồng

Cho danh sách a, gồm n hóa đơn. Anh/Chị viết chương trình thực hiện các yêu cầu dưới đây:

1. Nhập/xuất danh sách hóa đơn.
2. Đếm số hóa đơn có tổng tiền trên k triệu đồng.
3. Liệt kê những hóa đơn của khách hàng “Nguyen Van Thi”.
4. Đếm số hóa đơn của những khách hàng họ “Tran”.
5. Tính tổng doanh thu của những hóa đơn được lập vào ngày k bất kỳ.
6. Thêm một hóa đơn k vào tại vị trí h bất kỳ.
7. Kiểm tra xem trong danh sách có hóa đơn nào của khách hàng họ "Hoang"?
8. Sắp danh sách giảm dần theo tổng tiền.
   1. **Bài toán 03**

**Mô tả bài toán**: NGAY - Ngày tiếp theo

Giả sử hôm này là ngày d tháng t năm n. Hỏi ngày mai là ngày mấy?

**Dữ liệu nhập:**

- Gồm 3 số d, t, n cách nhau một khoảng trắng biểu thị cho ngày, tháng, năm (1 ≤ d ≤ 31, 1 ≤ t ≤ 12, 0 ≤ n ≤ 9999). Dữ liệu cho đảm bảo là một ngày hợp lệ (Ví dụ ngày không hợp lệ: 31 2 2014)

**Dữ liệu xuất:**

- In ra ngày, tháng, năm của ngày tiếp theo, mỗi số cách nhau một khoảng trắng. Không in số 0 ở đầu ngày và tháng.

**Các test cho bài toán**

* **Test #1, kết quả: ACCEPTED, thời gian: 15 ms, bộ nhớ: 856 KB**

Input

1 1 2014

Output

2 1 2014

Đáp án

2 1 2014

* **Test #2, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 612 KB**

Input

31 1 2014

Output

1 2 2014

Đáp án

1 2 2014

* **Test #3, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 660 KB**

Input

5 5 2000

Output

6 5 2000

Đáp án

6 5 2000

* **Test #4, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 788 KB**

Input

28 2 2001

Output

1 3 2001

Đáp án

1 3 2001

* **Test #5, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 728 KB**

Input

31 3 2002

Output

1 4 2002

Đáp án

1 4 2002

* 1. **Bài toán 04**

Viết chương trình nhập vào 1 ngày đầy đủ từ bàn phím.

**Thực hiện các yêu cầu sau:**

1. Xuất ngày vừa nhập ra màn hình
2. Kiểm tra xem ngày có hợp lệ không?
3. Tìm ngày kế tiếp
4. Tìm ngày trước đó
5. Tìm ngày trước đó k ngày
6. Tìm ngày sau đó k ngày
7. Tính khoảng cách giữa 2 ngày bất kỳ (hai ngày đó cách nhau bao nhiêu ngày)
8. Tính thứ khi biết một ngày bất kỳ
9. So sánh hai ngày
10. **Thực hành lập trình đệ quy Thời gian: 05 giờ**

***Mục Tiêu:***

* Cài đặt được các bài toán đệ quy cơ bản
* Ứng dụng được phương pháp lập trình đệ quy
  1. **Bài toán 01**

Cho dãy Fibonacci mô tả như sau: F0=1, F1=1, Fn=Fn-1+Fn-2 với n>1. Viết chương trình thực hiện các yêu cầu dưới đây:

1. Cài đặt bài toán trên bằng phương pháp lặp dùng cấu trúc mảng một chiều.
2. Cài đặt bài toán trên bằng phương pháp lặp dùng 03 biến.
3. Cài đặt bài toán trên bằng phương pháp đệ quy.
   1. **Bài toán 02**

Bài toán tính lãy xuất ngân hàng. Giả sử thuở ban đầu ông A có số tiền là X triệu, với lãi xuất hằng năm của ngân hàng là 6.5%/1 năm. Hỏi nếu ông A gửi vào ngân hàng thì sau n năm ông A có tổng cộng là bao nhiều tiền? Anh/Chị hãy xây dựng chương trình giải quyết bài toán trên bằng 2 phương pháp:

1. Phương pháp lặp
2. Phương pháp đệ quy
   1. **Bài toán 03**

**Mô tả bài toán: LUTH - Lũy thừa**

Cho hai số nguyên x và n, hãy tính lũy thừa xn.

**Dữ liệu nhập:**

- Là hai số nguyên x và n cách nhau một khoảng trắng (1 ≤ x ≤ 1000, 1 ≤ n ≤ 1012)

**Dữ liệu xuất:**

- Là 4 số cuối của lũy thừa xn (xn mod 104)

**Các test của bài toán:**

|  |
| --- |
| * **Test #1, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 1064 KB**   Input  2 3  Output  8  Đáp án  8   * **Test #2, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 860 KB**   Input  3 2  Output  9  Đáp án  9   * **Test #3, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 812 KB**   Input  3 3  Output  27  Đáp án  27   * **Test #4, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 796 KB**   Input  4 3  Output  64  Đáp án  64   * **Test #5, kết quả: ACCEPTED, thời gian: 15 ms, bộ nhớ: 812 KB**   Input  1 5  Output  1  Đáp án  1 |

1. **Thực hành ứng dụng Thời gian: 05 giờ**

***Mục Tiêu:***

* Nhận dạng được cấu trúc cần dùng cho một bài toán cơ bản
* Cài đặt được bài toán
  1. **Bài toán 01**

**Mô tả bài toán**: NGAY - Ngày tiếp theo

Giả sử hôm này là ngày d tháng t năm n. Hỏi ngày mai là ngày mấy?

**Dữ liệu nhập:**

- Gồm 3 số d, t, n cách nhau một khoảng trắng biểu thị cho ngày, tháng, năm (1 ≤ d ≤ 31, 1 ≤ t ≤ 12, 0 ≤ n ≤ 9999). Dữ liệu cho đảm bảo là một ngày hợp lệ (Ví dụ ngày không hợp lệ: 31 2 2014)

**Dữ liệu xuất:**

- In ra ngày, tháng, năm của ngày tiếp theo, mỗi số cách nhau một khoảng trắng. Không in số 0 ở đầu ngày và tháng.

**Các test cho bài toán**

* **Test #1, kết quả: ACCEPTED, thời gian: 15 ms, bộ nhớ: 856 KB**

Input

1 1 2014

Output

2 1 2014

Đáp án

2 1 2014

* **Test #2, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 612 KB**

Input

31 1 2014

Output

1 2 2014

Đáp án

1 2 2014

* **Test #3, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 660 KB**

Input

5 5 2000

Output

6 5 2000

Đáp án

6 5 2000

* **Test #4, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 788 KB**

Input

28 2 2001

Output

1 3 2001

Đáp án

1 3 2001

* **Test #5, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 728 KB**

Input

31 3 2002

Output

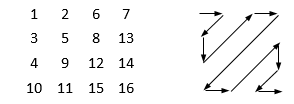
1 4 2002

Đáp án

1 4 2002

* 1. **Bài toán 02**
* **Mô tả bài toán**

Cho một ma trận vuông kích thước n x n, bạn hãy điền các số từ 1 đến n2 vào ma trận theo hình zigzag bắt đầu từ phía trên bên trái cho đến phía dưới bên phải. Ví dụ cho n = 4 thì ma trận zigzag là như sau:



**Dữ liệu nhập:**

Là số nguyên n (1 <= n <= 50)

**Dữ liệu xuất:**

Gồm n dòng biểu thị hình zig zag, mỗi dòng gồm n số nguyên, mỗi số cách nhau một khoảng trắng.

**Ví dụ:**

* **input**

5

**output**

1 2 6 7 15  
3 5 8 14 16  
4 9 13 17 22  
10 12 18 21 23  
11 19 20 24 25

* **input**

3

**output**

1 2 6  
3 5 7  
4 8 9

* 1. **Bài toán 03**
* **Mô tả bài toán**

Số tự nhiên N được gọi là số đẹp nếu cộng các chữ số của N lại ta có một số mà kết thúc bằng 9. Ví dụ một số số đẹp là 18 (1+8=9), 234 (2+3+4=9), 658 (6+5+8=19). Cho một số N, hãy kiểm tra xem N có phải là số đẹp hay không.

**Dữ liệu nhập:**

- Là số nguyên N (0 ≤ n ≤ 109)

**Dữ liệu xuất:**

- Nếu N là số đẹp, in ra "YES", nếu không in ra "NO"

**Ví dụ:**

* **input**

27

**output**

YES

* **input**

658

**output**

YES

* **input**

111

**output**

NO

* 1. **Bài toán 04**

**Mô tả bài toán: LUTH - Lũy thừa**

Cho hai số nguyên x và n, hãy tính lũy thừa xn.

**Dữ liệu nhập:**

- Là hai số nguyên x và n cách nhau một khoảng trắng (1 ≤ x ≤ 1000, 1 ≤ n ≤ 1012)

**Dữ liệu xuất:**

- Là 4 số cuối của lũy thừa xn (xn mod 104)

**Các test của bài toán:**

|  |
| --- |
| * **Test #1, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 1064 KB**   Input  2 3  Output  8  Đáp án  8   * **Test #2, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 860 KB**   Input  3 2  Output  9  Đáp án  9   * **Test #3, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 812 KB**   Input  3 3  Output  27  Đáp án  27   * **Test #4, kết quả: ACCEPTED, thời gian: 0 ms, bộ nhớ: 796 KB**   Input  4 3  Output  64  Đáp án  64   * **Test #5, kết quả: ACCEPTED, thời gian: 15 ms, bộ nhớ: 812 KB**   Input  1 5  Output  1  Đáp án  1 |

1. **Điều kiện thực hiện môn học**
2. Phòng học chuyên môn hóa/nhà xưởng: Phòng học có máy chiếu, phòng thực hành tin học.
3. Trang thiết bị máy móc: máy vi tính có phần mềm chuyên dụng, máy chiếu.
4. Học liệu, dụng cụ, nguyên vật liệu: sách, tập, máy tính có phần mềm Visual Studio, NetSupport School, trang học tập, group học tập, …
5. Các điều kiện khác: Có mạng Internet
6. **Năng lực đạt được và phương pháp đánh giá**
7. **Năng lực đạt được**

* **Kiến thức:** Biết cách phân tích và giải một số bài toán trên cấu trúc ma trận, con trỏ, kiểu dữ liệu tự định nghĩa, kỹ thuật lập trình đệ quy.
* **Kỹ năng:** Cài đặt được các bài toán cơ bản trên cấu trúc ma trận, kiểu dữ liệu cấu trúc, con trỏ; Cài đặt được một số bài toán bằng phương pháp lập trình đệ quy, đọc và ghi một số cấu trúc cơ bản.
* **Năng lực tự chủ và trách nhiệm:** Có ý thức hơn trong việc rèn luyện tư duy lập trình nhằm phát triển nghề nghiệp sau này.

1. **Phương pháp đánh giá**

* Đánh giá quá trình: 40%; trong đó các hình thức đánh giá:

+ Đánh giá thường xuyên

+ Bài tập cá nhân

+ Kiểm tra

* + - Tính trung bình các cột (số cột đánh giá và tỷ lệ % được giảng viên thống nhất với lớp học cụ thể trong buổi học đầu tiên).
* Thi cuối học kỳ: 60%

+ Hình thức: **thi vấn đáp hoặc thi thực hành đề đóng**

+ Thời gian: **từ 60 đến 120 phút**

+ Phân bổ điểm theo từng chương

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Nội dung** | **Điểm/%** | **Ghi chú** |
| 1 | Chương 1&2 | 2.5 điểm | 2 câu nhỏ |
| 2 | Chương 3 | 2.5 điểm | 2-3 câu nhỏ |
| 2 | Chương 4 | 2.5 điểm | 2-3 câu nhỏ |
| 3 | Chương 5&6 | 2.5 điểm | 2 câu nhỏ |

**Chú ý:** Nếu chọn hình thức thi vấn đáp, sẽ hỏi bất kỳ bài tập nào trong 06 buổi lên lớp.

1. **Hướng dẫn thực hiện môn học**
2. Môn tiên quyết trước khi học môn này: nhập môn lập trình
3. Môn học tiếp theo phụ thuộc vào môn này: Cấu trúc dữ liệu & giải thuật, Lập trình hướng đối tượng, lập trình Windows, …
4. Điều kiện được dự thi kết thúc môn: tham dự từ 80% thời lượng học trên lớp
5. Yêu cầu đối với người học

* Phải chuẩn bị bài trước ở nhà (nên trả lời danh mục các câu hỏi cho trước theo từng chương), đọc các vấn đề lý thuyết liên quan đến bài thực hành.
* Lắng nghe, đặt câu hỏi và trả lời câu hỏi, giải quyết tình huống giảng viên đưa ra.
* Hoàn thành các bài tập.
* Tham dự các kỳ thi và kiểm tra theo quy định môn học
* Tham dự tối thiểu 80% thời lượng tại phòng học

1. Yêu cầu đối với giảng viên và phương pháp giảng dạy

* Thuyết trình: trình giảng, trình chiếu, tóm tắt.
* Đối thoại: thảo luận, phỏng vấn, động não
* Đào tạo kỹ năng mềm kết hợp
* Hoạt động cá nhân, hoạt động nhóm
* Giao bài tập về nhà và hướng dẫn sửa các bài tập trên lớp

1. Giảng viên đảm nhiệm môn học, người hướng dẫn chuyên môn: có trình độ chuyên môn tối thiểu tốt nghiệp đại học chuyên ngành.
2. Các nội dung khác cần chú ý:

* Phân tích được ưu/nhược điểm của phương pháp lập trình từ đó vận dụng vào bài toán lập trình.
* Vận dụng và xử lý các bài toán về mảng hai chiều, kiểu dữ liệu cấu trúc bằng nhiều kỹ thuật lập trình.
* Phân tích, đánh giá, cài đặt kỹ thuật đệ quy.
* Phân loại được tập tin, cài đặt và ứng dụng tập tin vào các bài tập về mảng, kiểu dữ liệu cấu trúc.

1. **Tài liệu tham khảo**

***Tài liệu chính***

[1]. Giáo trình Kỹ thuật lập trình Khoa công nghệ thông tin trường Cao đẳng Công Thương Tp. HCM; Tác giả: Trần Anh Khoa – Nguyễn Thị Thu Cúc;

***Tài liệu tham khảo thêm***

[2]. Trần Đan Thư, Nguyễn Thanh Phương, Đinh Bá Tiến, Trần Minh Triết – Nhập môn lập trình – Nhà xuất bản khoa học và kỹ thuật 2011;

[3]. Trần Đan Thư, Nguyễn Thanh Phương, Đinh Bá Tiến, Trần Minh Triết, Đặng Bình Phương – Kỹ thuật lập trình – Nhà xuất bản khoa học và kỹ thuật 2014;

[4]. Nguyễn Tấn Trần Minh Khang – Bài tập kỹ thuật lập trình tập 1 – ĐH Khoa Học Tự Nhiên TP.HCM.