

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP HỒ CHÍ MINH
KHOA: CÔNG NGHỆ THÔNG TIN**

Ths. NGUYỄN THỊ THANH BÌNH

**BÀI GIẢNG
THỰC HÀNH NGÔN NGỮ LẬP TRÌNH**

**TRÌNH ĐỘ: ĐẠI HỌC
NGÀNH: CÔNG NGHỆ THÔNG TIN
MÔN: NGÔN NGỮ LẬP TRÌNH
SỐ TIẾT: 60**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2016

MỤC LỤC

| | |
|--|----|
| 1. THỰC HÀNH CHƯƠNG 1-2 – CƠ BẢN VỀ C | 4 |
| Mục tiêu: | 4 |
| 1.1. Cài đặt C-Free 5.0 và triển khai bài tập ví dụ:..... | 4 |
| 1.2. Cài đặt Microsoft Visual Studio 2012 và triển khai bài tập ví dụ: | 5 |
| 1.3. Bài tập Thực hành trên lớp | 9 |
| 1.4. Bài tập làm thêm: | 13 |
| 1.5. Kết luận buổi thực hành:..... | 14 |
| 2. THỰC HÀNH CHƯƠNG 3 – CÂU LỆNH CÓ CẤU TRÚC | 16 |
| Mục tiêu: | 16 |
| 2.1. Lệnh if..... | 16 |
| 2.2. Lệnh switch..... | 18 |
| 2.3. Cấu trúc lặp..... | 18 |
| 2.4. Bài tập làm thêm | 20 |
| 2.5. Kết luận buổi thực hành:..... | 21 |
| 3. THỰC HÀNH CHƯƠNG 4 - HÀM | 22 |
| Mục tiêu: | 22 |
| 3.1. Ví dụ - Truyền tham trị | 22 |
| 3.2. Truyền tham biến | 26 |
| 3.3. Bài tập làm thêm | 27 |
| 3.4. Kết luận buổi thực hành:..... | 29 |
| 4. THỰC HÀNH CHƯƠNG 5 – MẢNG | 31 |
| Mục tiêu: | 31 |
| 4.1. Mảng một chiều – Ví dụ | 31 |
| Bài tập mảng một chiều | 32 |
| 4.2. Mảng nhiều chiều – Ví dụ | 33 |
| Bài tập mảng nhiều chiều | 35 |
| 4.4. Kết luận buổi thực hành:..... | 36 |
| 9. ÔN TẬP | 38 |
| Mục tiêu: | 38 |
| Bài tập..... | 38 |
| Kết luận buổi thực hành:..... | 38 |
| TÀI LIỆU THAM KHẢO..... | 39 |
| 1. Tài liệu tiếng Việt | 39 |

| | |
|-----------------------------|----|
| 2. Tài liệu tiếng Anh | 40 |
|-----------------------------|----|

1. THỰC HÀNH CHƯƠNG 1-2 – CƠ BẢN VỀ C

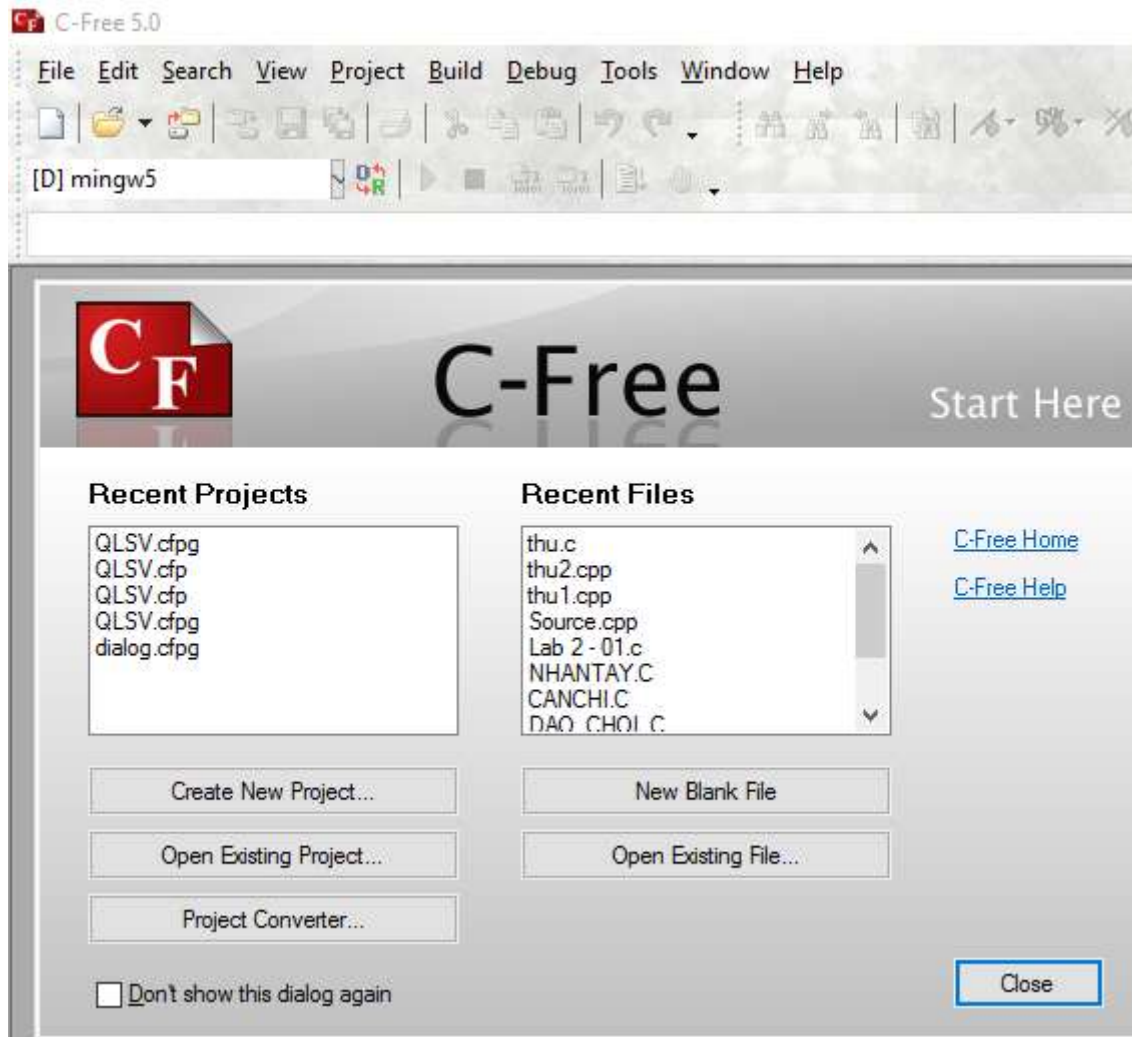
Mục tiêu:

- Ôn tập củng cố kiến thức cơ bản về các phần tử cơ bản của C:
 - + Từ khóa, Biến, Hằng, Biểu thức, Phép gán, Lỗi giải thích;
 - + Các kiểu dữ liệu trong C;
 - + Câu lệnh, vào, ra trong C;
 - + Các phép toán; thứ tự ưu tiên các phép toán.
- Hiểu cấu trúc lập trình
- Cài đặt, triển khai chương trình C
- Rèn luyện kỹ năng lập trình, vận dụng lý thuyết vào bài tập ở các nội dung trên và có liên quan.

1.1. Cài đặt C-Free 5.0 và triển khai bài tập ví dụ:

1. Tiến hành cài đặt phần mềm C-Free 5.0 .

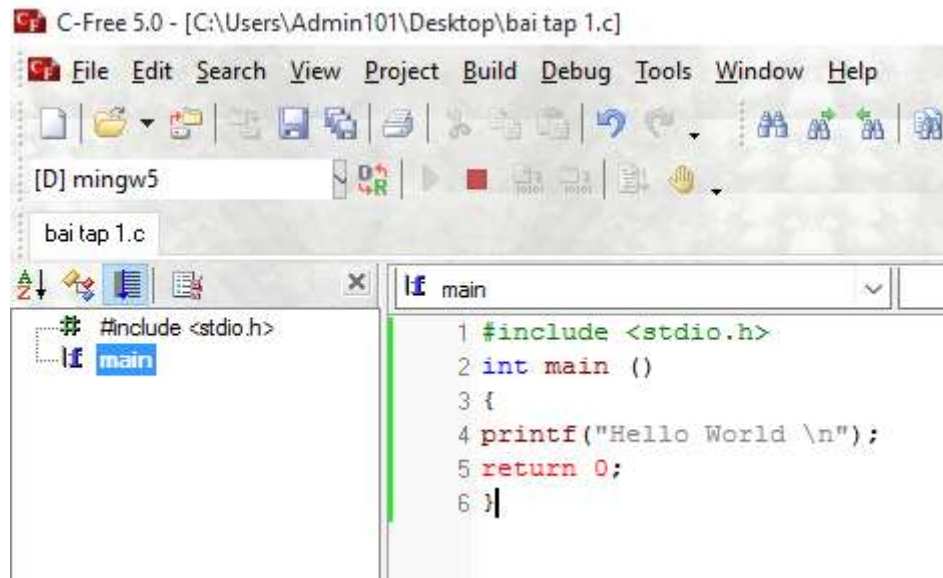
Khởi động phần mềm C-Free 5.0 và chọn New Blank File.



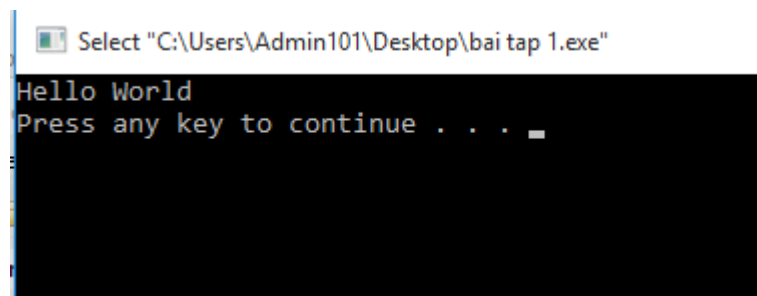
Mặc định thì C-Free tạo tệp Untitled1.cpp, những lệnh đặt trong cả tệp .c và .cpp đều chạy miễn là khai báo đủ các hàm thư viện tương ứng. Nhưng để chủ động cho việc quản lý và debug chương trình khi thực hành về C và thuận tiện cho chạy

chương trình trên các môi trường khác (chẳng hạn trong Linux), ta tạo thư mục bài tập theo từng chương (Chương 1, Chương 2, ..) và chủ động đặt tên tệp có phần mở rộng là .c hoặc .cpp bằng cách chọn File\Save (Ctrl+S) (hoặc File\Save As), chọn nơi lưu tệp (Chương 1) và nhập tên tệp, chẳng hạn là bai tap 1.c; bài tap 1.cpp.

2. Nhập vào đoạn code chương trình sau:



3. Chọn Build\Run hoặc ấn phím F5 để chạy chương trình, ta có kết quả sau:



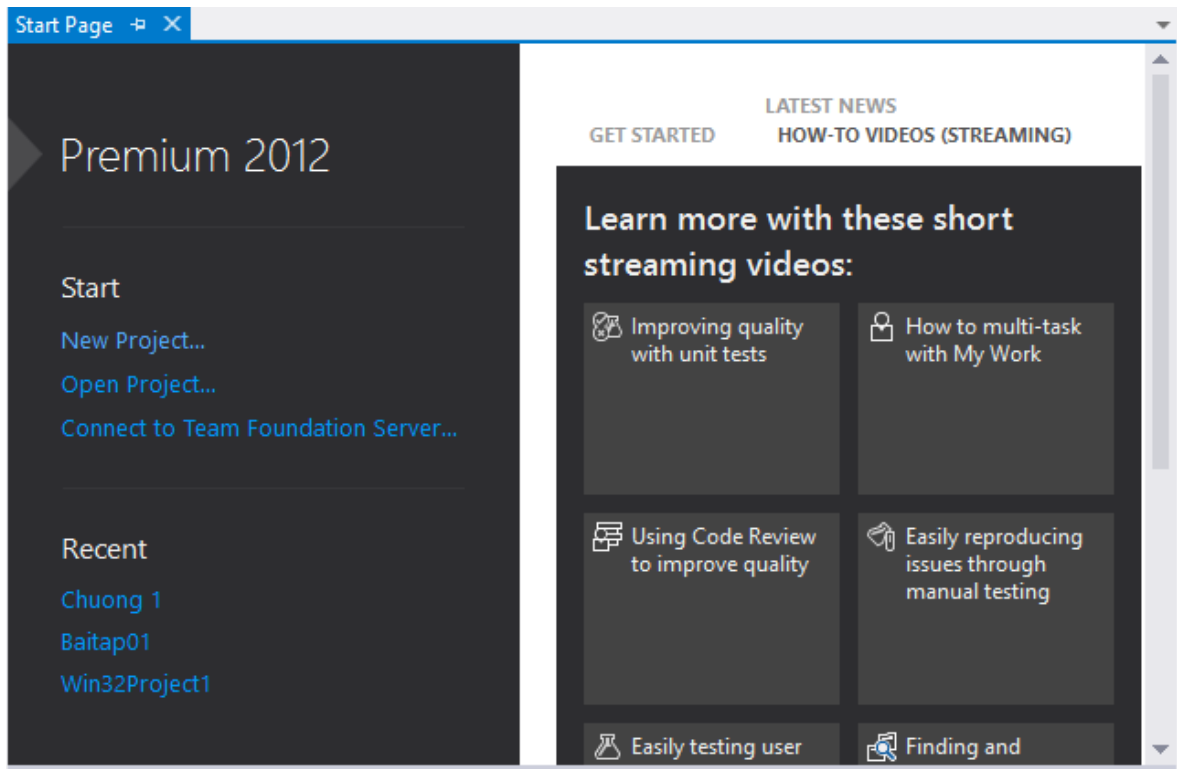
4. Nhấn phím bất kỳ để thoát về môi trường lập trình.

5. Thực hành các bài tập tiếp theo bằng cách vào menu File\New (hoặc ấn đồng thời Ctrl+N), và làm theo các bước từ bước 2 trên.

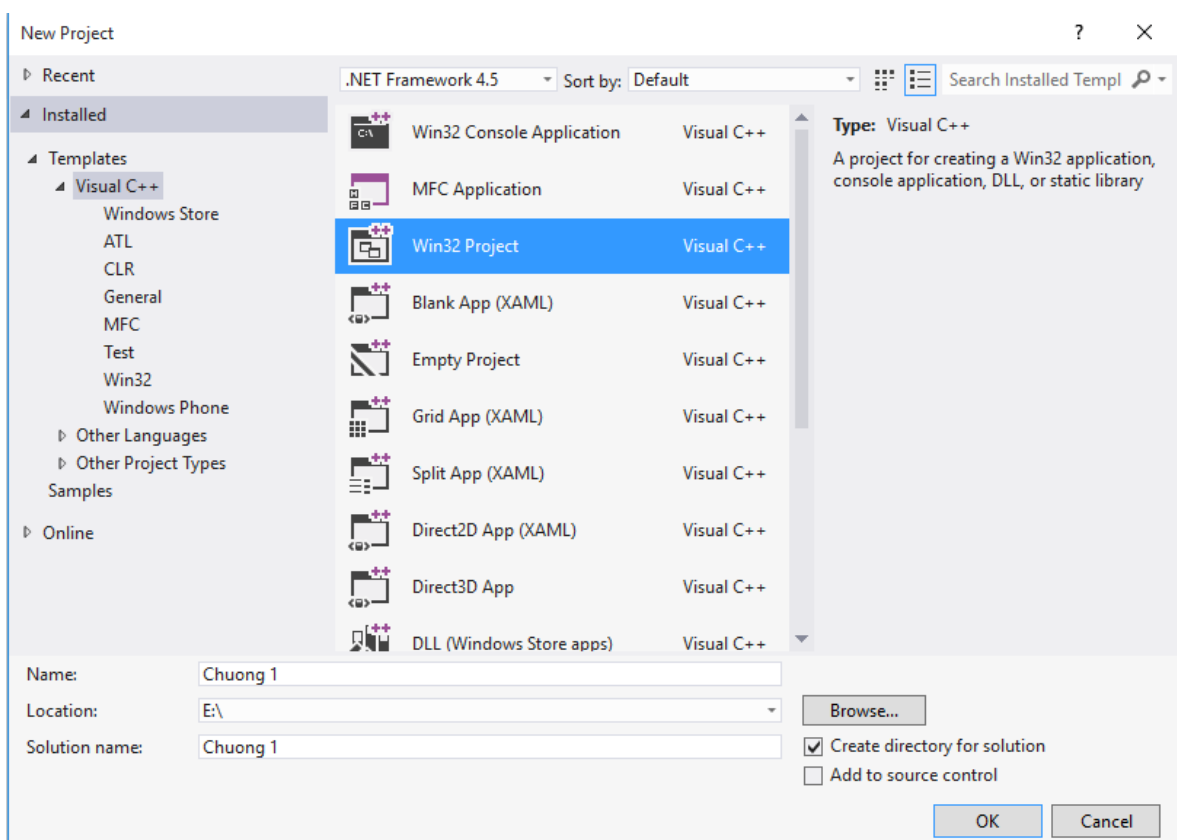
1.2. Cài đặt Microsoft Visual Studio 2012 và triển khai bài tập ví dụ:

1. Tiến hành cài đặt phần mềm, trong quá trình cài đặt thì chọn Visual C++ (Các gói phần mềm khác có thể chọn theo mục đích công việc hoặc không chọn để cải thiện tốc độ máy và bộ nhớ).

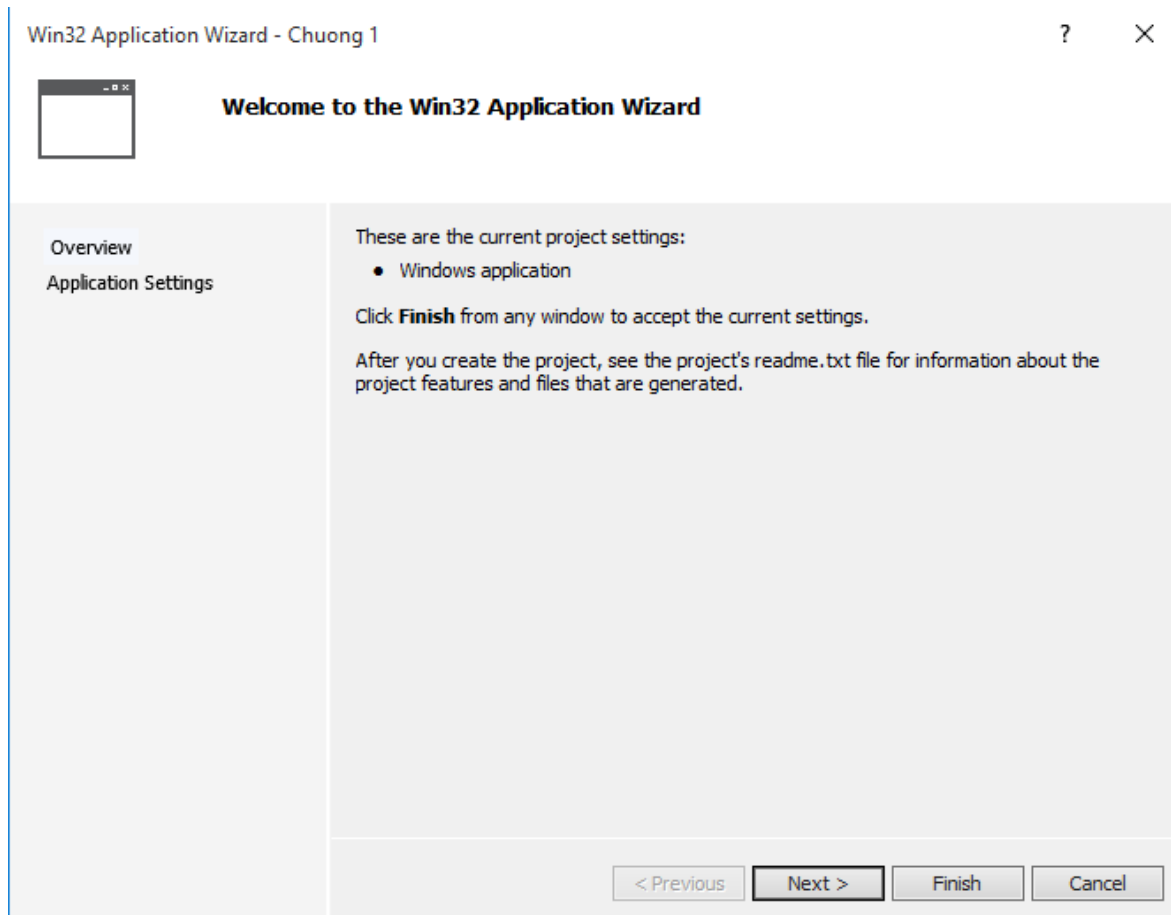
Khởi động Visual Studio 2012, và chọn New Project trong mục Start.



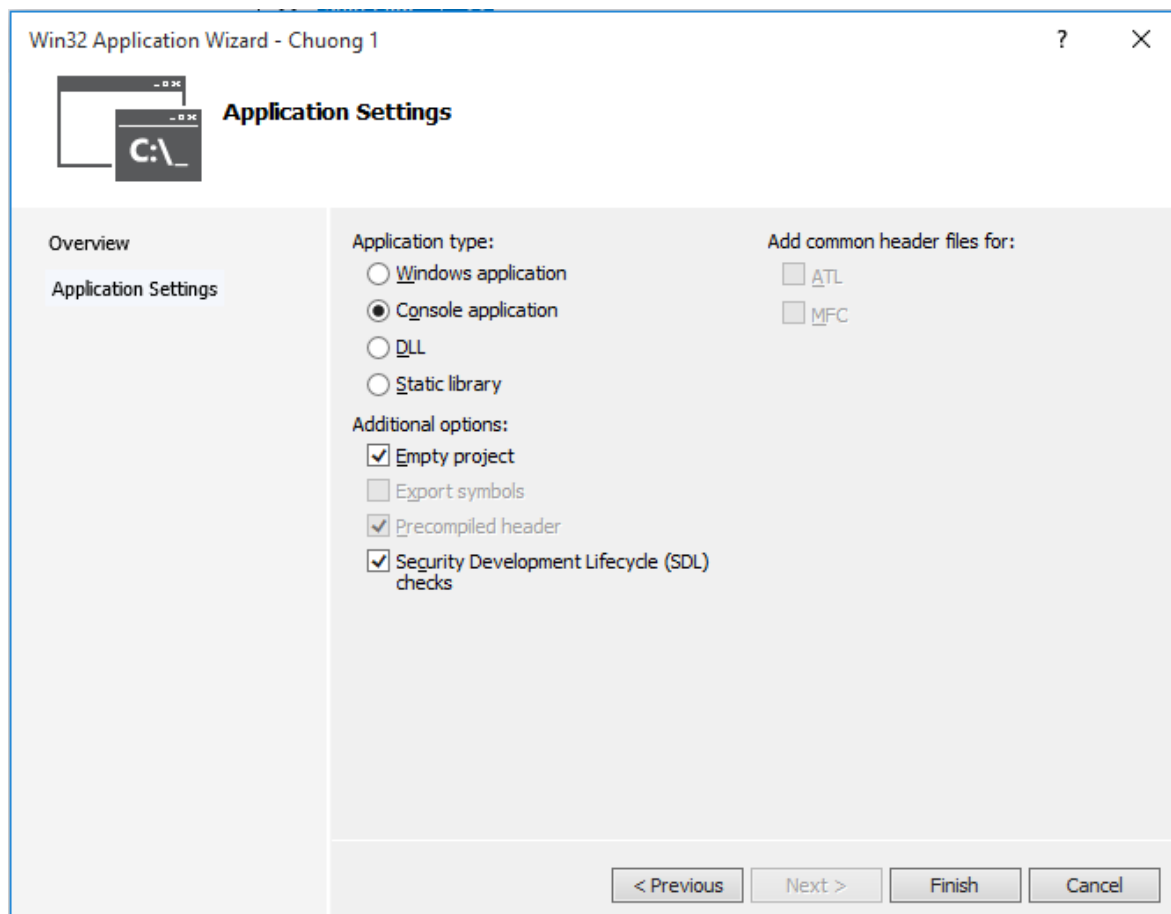
2. Lựa chọn và cài đặt các mục như hình vẽ, bao gồm cả các mục Name, Location, Solution name. Cuối cùng nhấp OK.



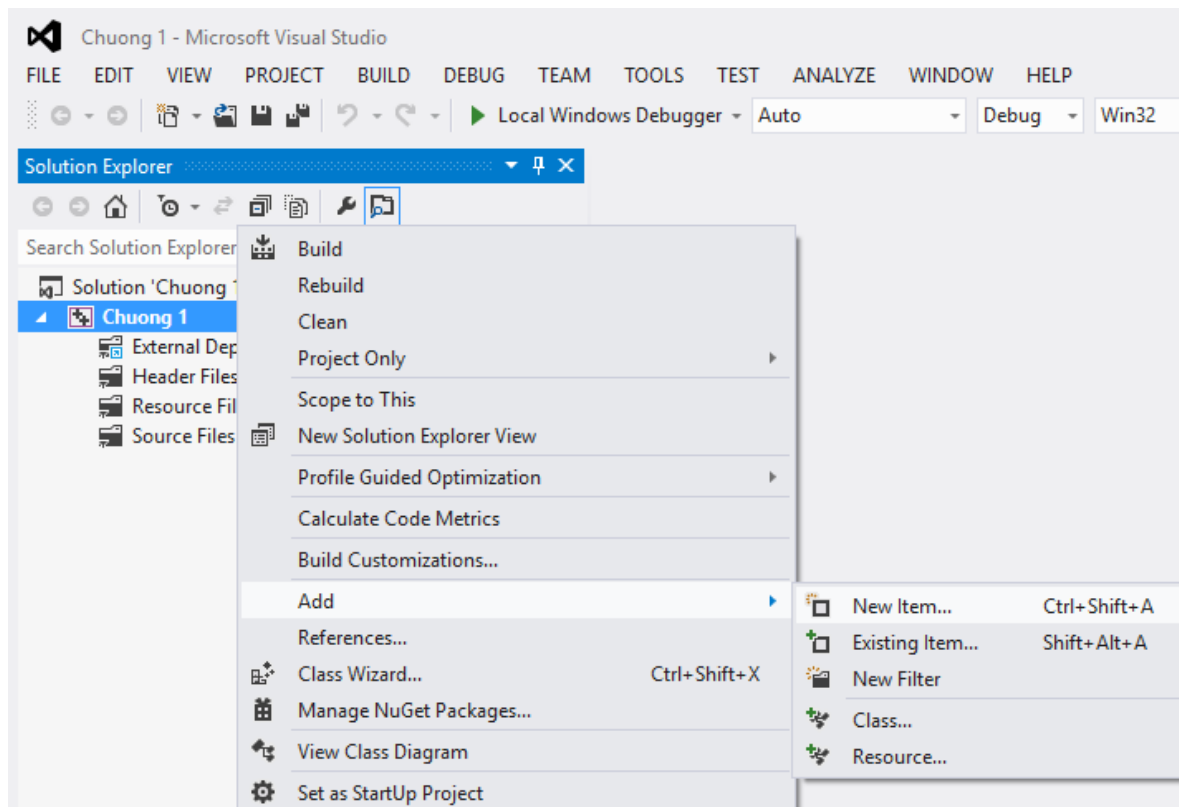
3. Chọn Next



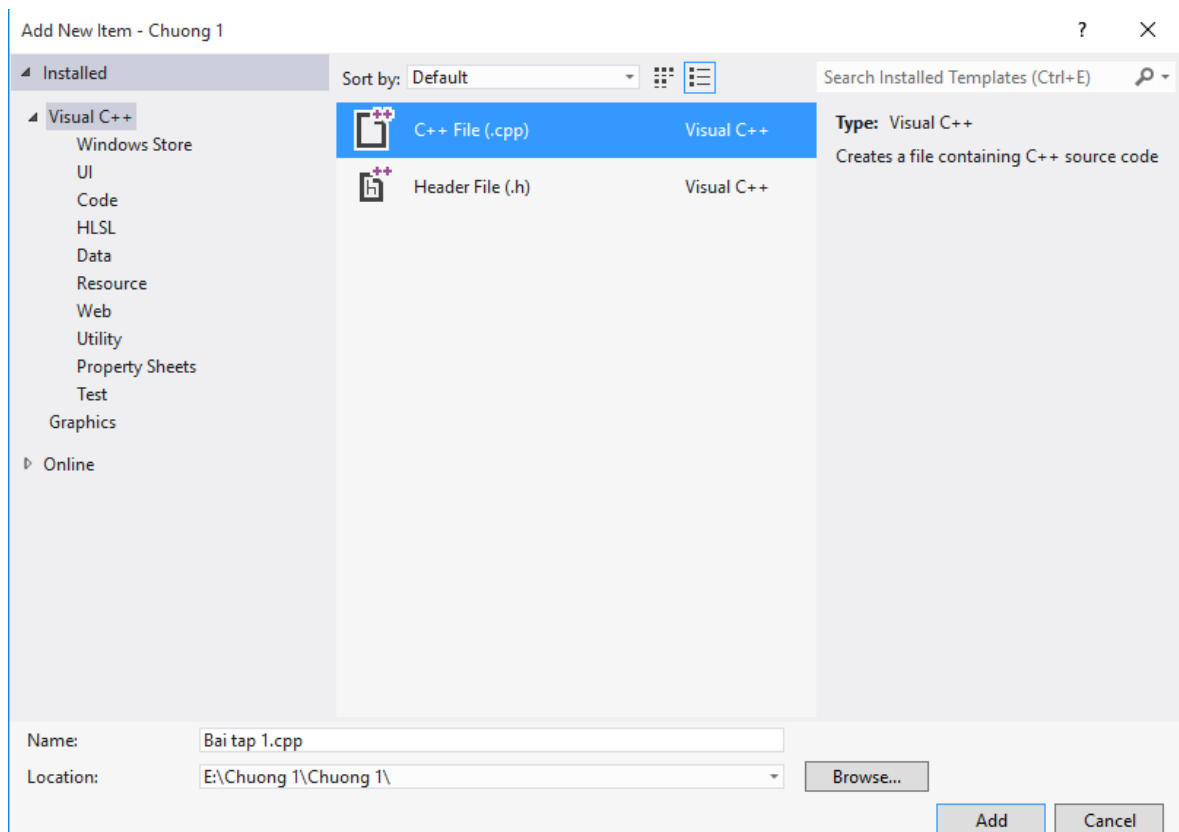
4. Lựa chọn các mục như hình vẽ



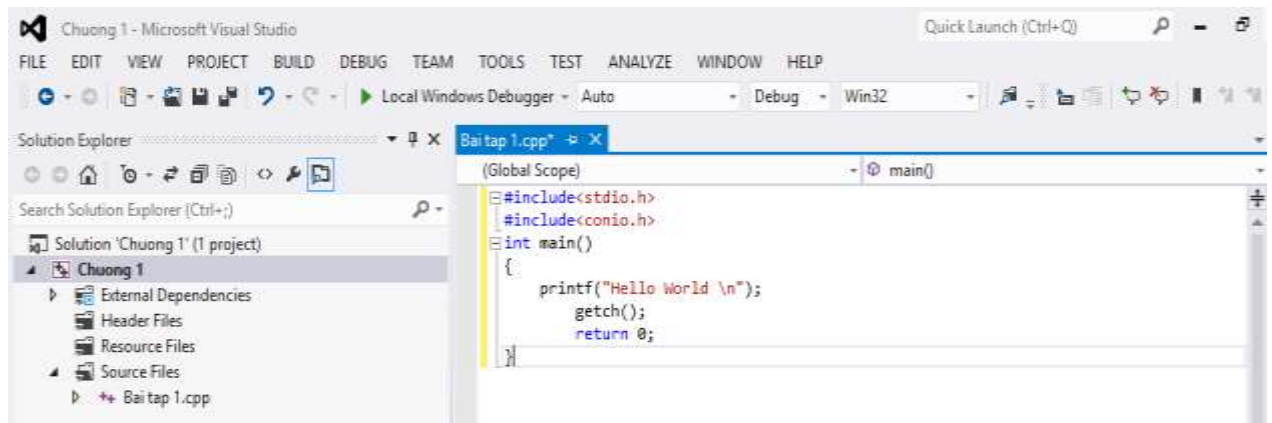
5. Nhấp phải chuột tại “Chương 1”, chọn Add, chọn New Item



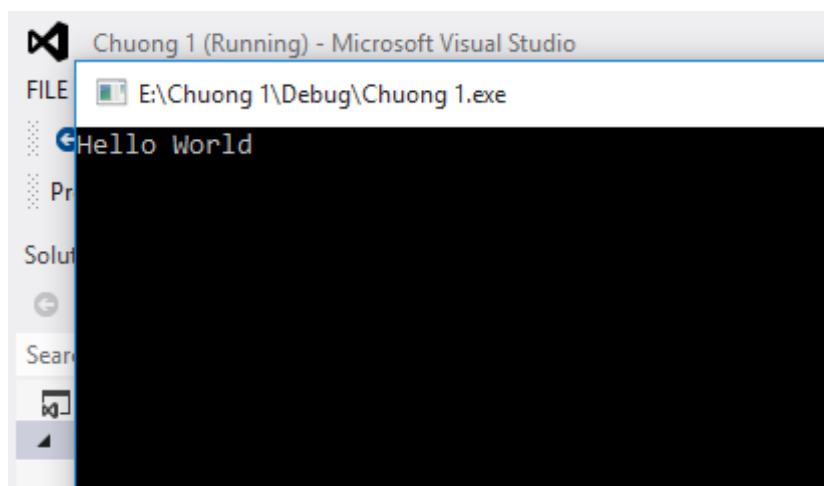
6. Cài đặt các mục như hình vẽ, rồi Add.



7. Nhập đoạn chương trình ví dụ mẫu, xong thực thi bằng lệnh F5



8. Kết quả ta được:



Chú ý: Nếu không có lệnh getch() thì sẽ không xuất hiện màn hình Console trên để ta xem được kết quả.

1.3. Bài tập Thực hành trên lớp

Bài tập 3.1. a) Nhập và chạy chương trình sau, ghi lên đĩa với tên tệp là Chuong1\Bai tap 31.c:

```
#include<stdio.h>
#include<conio.h>
int main()
{
    printf("Hello World \n");
    getch();
    return 0;
}
```

b) Xóa dấu ; cuối lệnh printf, chạy lại chương trình và giải thích kết quả. Ghi nhớ thông báo lỗi.

c) Trả lại dấu ; và xóa dòng lệnh return 0; chạy lại chương trình và giải thích kết quả. Ghi nhớ thông báo lỗi.

d) Trả lại lệnh return 0; và xóa dòng lệnh khai báo thư viện #include<conio.h> và chạy lại chương trình. Xem và giải thích kết quả. Ghi nhớ thông báo lỗi.

d) Trả lại khai báo #include<conio.h> đồng thời thêm dấu ; cuối một trong hai khai báo thư viện này và chạy lại chương trình, quan sát kết quả và giải thích. Ghi nhớ thông báo lỗi.

Bài tập 3.2. Cho biết kết quả của các câu lệnh sau; kiểm tra bằng cách nhập đoạn chương trình vào máy để test.

| | |
|--|--|
| <pre>int i=2; printf("XX %d XX",i); printf("XX %3d XX",i); printf("XX %-3d XX",i); printf("XX %03d XX",i); printf("XX %+3d XX",i);</pre> | <pre>float x=5.5; printf("XX %e XX",x); printf("XX %E XX",x); printf("XX %f XX",x); printf("XX %10.5f XX",x); printf("XX %-10.5f XX",x);</pre> |
| <pre>float x=5.5; printf("XX %f XX",x); printf("XX %1.0f XX",x); printf("XX %1.2f XX",x); printf("XX %5.2f XX",x);</pre> | <pre>char ch= 'A'; printf("XX %x XX",ch); printf("XX %4x XX",ch); printf("XX %04x XX",ch); printf("XX %cXX",ch); printf("XX %3cXX",ch); printf("XX %-3cXX",ch);</pre> |
| <pre>int i=2; char ch= 'A'; float x=5.5; printf("XX %3dYY%cZZ%4.1fXX\n",i,ch,x); printf("X %3d %c %4.1f",i,ch,x);</pre> | <pre>printf("DHCN :\" "); printf("DHCN :%% ");</pre> |

Bài tập 3.3. Cho biết kết quả sau khi thực hiện chương trình sau:

```
#include<stdio.h>
```

```
main()
```

```
{
    int i=10,j=4,k;
    float x;
    k=i+j;
    printf("Tong la: %d\n",k);
    k=j-i;
    printf("Hieu la: %d\n",k);
    x=i/j;
    printf("Thuong la: %f\n",x);
    x=(float)i/j;
    printf("Thuong la: %f\n",x);
    k=i*j;
    printf("Du cua phep chia la: %d\n",k);
    x=1.5;
    i=5;
    printf("x+i= %f\n",x+i);
}
```

Bài tập 3.4. Cho biết kết quả của các câu lệnh sau; kiểm tra bằng cách nhập đoạn chương trình vào máy để test.

Cho trước $i=3; j=5;$

| | | | |
|-------------|--|-------------|--|
| $i=++j;$ | | $i=j++;$ | |
| $i++;$ | | $j=++i+5;$ | |
| $j=i++ +5;$ | | $j--;$ | |
| $i=-j -5;$ | | $i=j-- -5;$ | |

Bài tập 3.5. Nhập và thực thi đoạn chương trình sau:

```
#include<stdio.h>
```

```
main()
```

```
{int a,b;
```

```
float tb;
```

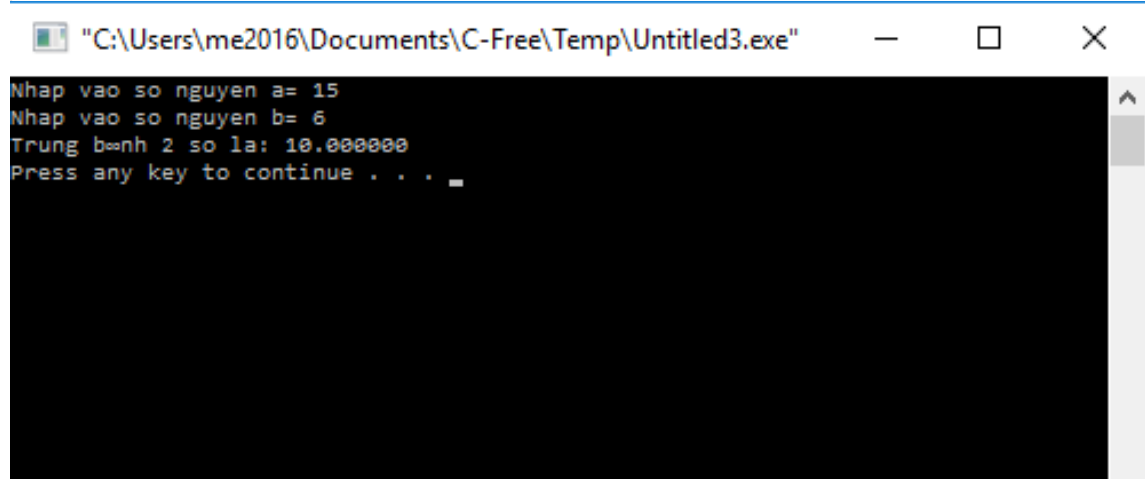
```
printf("Nhập vào số nguyên a= ");scanf("%d",&a);
```

```
printf("Nhập vào số nguyên b= ");scanf("%d",&b);
```

```
tb=(a+b)/2;
```

```
printf("Trung bình 2 số là: %f \n",tb);
```

```
}
```



- Nhận xét kết quả và giải thích nguyên nhân kết quả sai ?
- Thay đổi lệnh $tb = (a+b)/2$ trong chương trình mẫu 2 thành $tb = (a+b)/2.0$, sau đó biên dịch lại, thực thi chương trình và xem kết quả.
- Thay đổi lệnh `printf("Trung bình 2 số là: %f", tb)` trong chương trình thành `printf("Trung bình 2 số là: %d", tb)` sau đó biên dịch lại, thực thi chương trình và xem kết quả.
- Thay đổi và bổ sung thêm lệnh trong chương trình sao cho chương trình cho phép nhập vào điểm 3 môn học: toán, lý, hóa và tính điểm trung bình, in ra kết quả.

Bài tập 3.6. Viết chương trình nhập vào bán kính hình cầu, tính và in ra diện tích, thể tích của hình cầu đó.

Hướng dẫn: $S = 4\pi R^2$ và $V = (4/3)\pi R^3$

Bài tập 3.7. Viết chương trình nhập vào tọa độ của hai điểm (x_1, y_1) và (x_2, y_2)

- Tính hệ số góc của đường thẳng đi qua hai điểm đó theo công thức:

$$\text{Hệ số góc} = (y_2 - y_1) / (x_2 - x_1)$$

- Tính khoảng cách giữa hai điểm theo công thức:

$$\text{Khoảng cách} = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}$$

Bài tập 3.8. Viết chương trình:

- a) Nhập vào một ký tự, và:
 1. In ra mã ASCII của ký tự đó.
 2. In ra ký tự kế tiếp của nó.
- b) Nhập vào một số tự nhiên trong khoảng từ 0->255, và:
 1. In ra ký tự tương ứng mã ASCII số đó.
 2. In ra ký tự kế tiếp.

Bài tập 3.9. Viết chương trình nhập vào các giá trị điện trở R_1 , R_2 , R_3 của một mạch điện : Tính tổng trở tương đương theo công thức:

$$1/R_{td} = 1/R_1 + 1/R_2 + 1/R_3$$

Bài tập 3.10. Nhập đoạn chương trình sau và phân tích để hiểu độ lớn kiểu dữ liệu:

```
#include<stdio.h>
main()
{
    struct S {char ch[10];
               int i;
               }st;

    int j=2;
    printf("Kích thước kiểu kí tự: %d \n",sizeof(char));
    printf("Kích thước kiểu nguyên: %d \n",sizeof(int));
    printf("Kích thước kiểu nguyên dương: %d \n",sizeof(unsigned));
    printf("Kích thước kiểu short: %d \n",sizeof(short));
    printf("Kích thước kiểu longint: %d \n",sizeof(long));
    printf("Kích thước kiểu thực: %d \n",sizeof(float));
    printf("Kích thước kiểu double: %d \n",sizeof(double));
    printf("Kích thước của j: %d \n",sizeof(j));
    printf("Kích thước con trỏ chuỗi: %d \n",sizeof(char*));
    printf("Kích thước biến cấu trúc st %d \n",sizeof(st));
    printf("Kích thước cấu trúc: %d \n",sizeof(struct S));
}

#include <iostream.h>
main()
{
    cout <<"Size of data types:\n";
    cout <<"Size of char = " << sizeof(char) << " bytes\n";
    cout <<"Size of unsigned char = " << sizeof(unsigned char) << " bytes\n";
    cout <<"Size of signed char = " << sizeof(signed char) << " bytes\n";
    cout <<"Size of int = " << sizeof(int) << " bytes\n";
    cout <<"Size of unsigned int = " << sizeof(unsigned int) << " bytes\n";
    cout <<"Size of signed int = " << sizeof(signed int) << " bytes\n";
    cout <<"Size of short = " << sizeof(short) << " bytes\n";
    cout <<"Size of unsigned short = " << sizeof(unsigned short) << " bytes\n";
    cout <<"Size of signed short = " << sizeof(signed short) << " bytes\n";
```

```

cout << "Size of long = " << sizeof(long) << " bytes\n";
cout << "Size of unsigned long = " << sizeof(unsigned long) << " bytes\n";
cout << "Size of signed long = " << sizeof(signed long) << " bytes\n";
cout << "Size of float = " << sizeof(float) << " bytes\n";
cout << "Size of double = " << sizeof(double) << " bytes\n";
cout << "Size of long double = " << sizeof(long double) << " bytes\n";
}

```

1.4. Bài tập làm thêm:

1.4.1. What will be the output if a=5.

```

printf("%d",++a);
printf("%d",a++);

```

1.4.2. What will be the output if a=5,

```

printf("%d",a++);
printf("%d",++a);

```

1.4.3. Write down C statements to perform the following operations:

```

z = 4.2(x+y)5/z - 0.52x/(y+z);
x = a2 + 2ab + b2;

```

1.4.4. Giả sử a, b là hai số thực. Biểu thức nào dưới đây viết không đúng theo cú pháp của ngôn ngữ lập trình C:

- a) (a+=b)
- b) (a-=b)
- c) (a*=b)
- d) (a>>=b)

1.4.5. Cho biết kết quả của chương trình sau:

```

#include<stdio.h>
main()
{ int i,j=20,k=2;
  i=(j<=k);
  printf("i=%d\n",i);
}

```

Gợi ý: Phép dịch trái số học << n bits thực hiện bằng nhân số đó với 2^n với điều kiện không quá dung lượng; còn dịch phải số học >> n bits thực hiện bằng cách chia số đó với 2^n .

1.4.6. Cho biết kết quả của chương trình sau:

```

#include<stdio.h>
main()
{ int i,j=20,k=2;
  i=(j>=k);
  printf("i=%d\n",i);
}

```

1.4.7. Viết chương trình nhập vào một số thực a và in ra màn hình giá trị diện tích và chu vi tam giác đều cạnh a:

1.4.8. Viết chương trình nhập vào hai số thực a và b. In ra màn hình giá trị:

$$\frac{2a^2 + 3b^3 + 4}{a + b}$$

1.4.9. Viết chương trình cho phép nhập vào giờ, phút và giây, hãy đổi sang giây và in kết quả ra màn hình.

1.4.10. Viết chương trình cho phép nhập vào thời gian của một công việc nào đó tính bằng giây. Hãy chuyển đổi và in ra màn hình thời gian trên dưới dạng bao nhiêu giờ, bao nhiêu phút, bao nhiêu giây.

1.4.11. Viết chương trình nhập vào tổng số giây, đổi sang giờ, phút, giây và xuất kết quả ra màn hình theo dạng giờ:phút:giây (nếu số có một chữ số thì xuất thêm số 0 ở đầu – Ví dụ: 03:20:04).

1.4.12. Viết chương trình đảo ngược một số nguyên dương có đúng 3 chữ số.

Hướng dẫn: x-số nhập vào có 3 chữ số => đảo ngược của x:

$$x = (x \% 10) * 100 + ((x / 10) \% 10) * 10 + x / 100$$

1.4.13. Sinh viên hãy làm lại các bài tập, nhưng trong môi trường C++.

1.5. Kết luận buổi thực hành:

1.5.1. Danh sách sinh viên vắng:

.....

1.5.2. Công tác chuẩn bị bài ở nhà:

.....

.....

1.5.3. Tinh thần thái độ thực hành:

.....

.....

1.5.4. Những lỗi thường mắc phải trong quá trình thực hành:

a/ Lỗi cài đặt phần mềm:

.....

.....

b/ Lỗi tạo tập tin, thao tác chỉnh sửa dữ liệu.....

.....

.....

.....

c/ Lỗi về kiến thức cơ bản của ngôn ngữ lập trình.....

.....

.....

.....

.....

.....

.....

.....

.....

1.5.5. Hướng dẫn công việc ở nhà:

.....

.....

.....

.....

.....

.....

.....

.....

.....

2. THỰC HÀNH CHƯƠNG 3 – CÂU LỆNH CÓ CẤU TRÚC

Mục tiêu:

- Ôn tập củng cố kiến thức cơ bản về:
 - + Xác định dữ liệu vào, ra;
 - + Phân tích các bài toán đơn giản;
 - + Ý nghĩa, các bước lập trình;
 - + Hình thành thuật toán giải từng bài toán;
- Hiểu câu lệnh đơn, lệnh ghép, cấu trúc rẽ nhánh, lặp; thoát khỏi lặp, lệnh goto; Sử dụng các cấu trúc lồng nhau;
- Rèn luyện kỹ năng lập trình, vận dụng lý thuyết vào bài tập ở các nội dung trên và có liên quan.

2.1. Lệnh if

Bài tập 1: Cho biết kết quả của đoạn chương trình sau:

```
int a=9, b=6;
a++;
a=a+b--;
a=a+(-b);
if(a%2==0)
printf("Gia tri cua a la chan");
printf("Tong cua a va b la: %d", a+b);
```

Bài tập 2: Cho biết kết quả của đoạn chương trình sau:

```
int a=7, b=8;
a++;
a=a+(b--);
--b;
a--;
a=(-a)+(-b);
if(a%2!=0)
printf("\n a la so le");
else
printf("\n a la so chan");
printf("\na = %d",a);
```

Bài tập 3: Cho biết kết quả của đoạn chương trình sau:

```
int x=5, y;
y=x++ + 5;
printf("x=%d, y=%d\n", x, y);
y*=6;
x=y%7;
printf("x=%d,y=%d,y/x=%d", x, y, y/x);
```

Bài tập 4: Lập chương trình nhập vào 2 số nguyên a và b và in ra màn hình giá trị:

$$S = \begin{cases} \frac{a^3 + 3b^2}{a^2 - 1} & \text{Nếu } a \text{ khác } 1 \\ 0 & \text{Nếu } a = 1 \end{cases}$$

Bài tập 5: Lập chương trình nhập vào một số thực dương a và in ra màn hình kết quả:

$$S = \frac{a^5 + 3a}{2\sqrt{a}}$$

Bài tập 6: Nhập vào hai số nguyên a, b. In ra màn hình giá trị lớn nhất.

Lưu ý: Sử dụng cả 2 cách: cấu trúc If hoặc toán tử ?.

Bài tập 7: Viết chương trình nhập vào 4 số nguyên. Tìm và in ra số lớn nhất.

Hướng dẫn: Ta có 4 số nguyên a, b, c, d. Tìm 2 số nguyên lớn nhất x, y của 2 cặp (a, b) và (c, d). Sau đó so sánh 2 số nguyên x, y để tìm ra số nguyên lớn nhất.

Bài tập 8: Viết chương trình nhập vào số nguyên dương, in ra thông báo số chẵn hay lẻ.

Hướng dẫn: Nhập vào số nguyên dương x. Kiểm tra nếu x chia chẵn cho 2 thì x là số chẵn (hoặc chia cho 2 dư 0) ngược lại là số lẻ.

Bài tập 9: Viết chương trình giải phương trình bậc nhất: $ax + b = 0$ với a, b nhập vào từ bàn phím.

Hướng dẫn: Nhập vào 2 biến a, b.

Nếu a khác 0 thì kết luận pt có nghiệm duy nhất $x = -b/a$;

Ngược lại

Nếu $b=0$ thì kết luận pt vô định;

Ngược lại thì kết luận pt vô nghiệm.

Bài tập 10: Viết chương trình giải phương trình bậc 2: $ax^2 + bx + c = 0$, với a, b, c nhập vào từ bàn phím.

Hướng dẫn: Nhập vào 3 biến a, b, c.

Tính $\Delta = b^2 - 4ac$

Nếu $\Delta < 0$ thì

Phương trình vô nghiệm

Ngược lại

Nếu $\Delta = 0$ thì

$$x_1 = x_2 = -b/(2a)$$

Ngược lại

$$x_1 = (-b + \sqrt{\Delta})/(2a)$$

$$x_2 = (-b - \sqrt{\Delta})/(2a)$$

Bài tập 11: Viết chương trình nhập vào giờ phút giây (hh:mm:ss). Cộng thêm số giây nhập vào và in ra kết quả dưới dạng hh:mm:ss.

Hướng dẫn: Nhập vào giờ phút giây vào 3 biến gio, phut, giay và nhập vào giây cộng thêm vào biến them:

Nếu $giay + them < 60$ thì

$$giay = giay + them$$

Ngược lại

{ giay = (giay + them) - 60

phut = phut + 1 }

Nếu phut >= 60 thì

{ phut = phut - 60

gio = gio + 1 }

2.2. Lệnh switch

Bài tập 1. Lập chương trình nhập vào một số nguyên là điểm môn học. Đưa ra xếp loại lực học theo tiêu chuẩn sau:

Nếu điểm từ 1-2 thì xếp loại Kém

Nếu điểm từ 3-4 thì xếp loại Yếu

Nếu điểm từ 5-6 thì xếp loại TB

Nếu điểm từ 7-8 thì xếp loại Khá

Nếu điểm từ 9-10 thì xếp loại Giỏi

Bài tập 2.1. Viết chương trình nhập vào tháng, in ra tháng đó có bao nhiêu ngày.

Hướng dẫn: Nhập vào tháng

Nếu là tháng 1, 3, 5, 7, 8, 10, 12 thì có 30 ngày

Nếu là tháng 4, 6, 9, 11 thì có 31 ngày

Nếu là tháng 2 và là năm nhuận thì có 29 ngày ngược lại 28 ngày (Năm nhuận là năm chia hết cho 4 nhưng không chia hết cho 100 hoặc chia hết cho 400)

Bài tập 2.2. Viết chương trình xác định biến ký tự color rồi in ra thông báo

RED, nếu color = 'R' hoặc color = 'r'

GREEN, nếu color = 'G' hoặc color = 'g'

BLUE, nếu color = 'B' hoặc color = 'b'

BLACK, nếu color có giá trị khác.

Bài tập 3. Viết chương trình nhập vào 2 số x, y và 1 trong 4 toán tử +, -, *, /. Nếu là + thì in ra kết quả x + y, nếu là - thì in ra x - y, nếu là * thì in ra x * y, nếu là / thì in ra x / y (nếu y = 0 thì thông báo không chia được).

Bài tập 4. Làm lại bài tập phân cấu trúc lệnh switch nhưng sử dụng cấu trúc lệnh if. Hãy kiểm tra trong các bài tập cấu trúc lệnh if nào có thể sử dụng được cấu trúc lệnh switch.

2.3. Cấu trúc lặp

Trong phần bài tập phần này, với mỗi bài tập, sinh viên hãy cố gắng lập trình theo tất cả các cấu trúc lặp đã được học nếu có thể, đó là:

- Cấu trúc lặp với số lần lặp biết trước: *for...*
- Cấu trúc lặp với số lần lặp không biết trước, thực hiện lệnh trước, kiểm tra điều kiện sau: *do..while*

- Cấu trúc lặp với số lần lặp không biết trước, kiểm tra điều kiện trước, thực hiện lệnh sau nếu thỏa điều kiện: *while...*

Sau đó hãy lựa chọn cho mình giải pháp tốt nhất ứng với bài toán đó.

Bài tập 1: Lập chương trình nhập vào số tự nhiên N:

- Tính tổng các số từ 0 đến N, in kết quả ra màn hình.
- Tính tổng các số chẵn, in kết quả ra màn hình.
- Tính tổng các số tự nhiên chia hết cho 3 và nhỏ hơn 100, in kết quả ra màn hình.
- Tính tổng các bình phương các số lẻ từ 1 đến N, in kết quả ra màn hình
- Tính giai thừa của N, in kết quả ra màn hình.

Bài tập 2: Lập chương trình tìm một số tự nhiên có hai chữ số biết:

- Tổng hai chữ số là 10 và hiệu hai chữ số là 2. In kết quả ra màn hình.
- Tổng hai chữ số là 12 và thương hai chữ số là 2. In kết quả ra màn hình.

Bài tập 3. Lập chương trình tìm số tự nhiên n lớn nhất để khi thay vào n thì được $3n^2 - 17n < 5$. In kết quả ra màn hình.

Bài tập 4: Lập chương trình giải bài toán cổ:

Vừa gà vừa chó
Bó lại cho tròn
Ba mươi sáu con
Một trăm chân chẵn

Bài tập 5: Lập chương trình giải bài toán cổ:

Trăm trâu trăm cỏ
Trâu đứng ăn năm
Trâu nằm ăn ba
Lụ khụ trâu già
Ba con một bó.

Bài tập 6: Lập chương trình:

- In ra bảng mã ASCII
- Nhập vào một số nguyên rồi in ra tất cả các ước số của số đó.
- Tính tổng nghịch đảo của N số nguyên đầu tiên theo công thức : $S = 1 + 1/2 + 1/3 + \dots + 1/N$
- Tìm USCLN, BSCNN của 2 số.
- Nhập vào một số và kiểm tra xem số đó có phải là số nguyên tố hay không?
- Tính x^n với x, n được nhập vào từ bàn phím.
- Lặp lại nhiều lần công việc nhập một ký tự và in ra mã ASCII của ký tự đó, khi nào nhập số 0 thì dừng.

2.4. Bài tập làm thêm

2.4.1. Cho ba số a, b, c đọc vào từ bàn phím. Hãy in ra màn hình theo thứ tự tăng dần các số. (Chỉ được dùng thêm hai biến phụ) .

2.4.2. Viết chương trình nhập vào điểm 3 môn thi: Toán, Lý, Hóa của học sinh. Nếu tổng điểm ≥ 15 và không có môn nào dưới 4 thì in kết quả đậu. Nếu đậu mà các môn đều lớn hơn 5 thì in ra lời phê "Học đều các môn", ngược lại in ra "Học chưa đều các môn", các trường hợp khác là "Thi hỏng".

2.4.3. Viết chương trình nhập vào 3 giá trị nguyên dương a, b, c. Kiểm tra xem a, b, c có phải là 3 cạnh của tam giác không? Nếu là 3 cạnh của tam giác thì hãy cho biết tam giác đó thuộc loại nào? (thường, cân, vuông, đều, v.v...).

Tính diện tích của tam giác theo công thức sau:

$$S = \sqrt{p(p-a)(p-b)(p-c)} \quad \text{Trong đó } p \text{ là nửa chu vi của tam giác}$$

Hướng dẫn: Nếu a, b, c là 3 cạnh của tam giác thì phải thỏa mãn $(a+b > c)$ và $(a+c > b)$ và $(b+c > a)$.

2.4.4. Viết chương trình tính tiền cước TAXI, biết rằng:

- km đầu tiên là 5000đ.
- 29 km tiếp theo là 12000đ.
- Từ km thứ 31 trở đi giá 10000đ.

Hãy nhập số km sau đó in ra số tiền phải trả.

2.4.5. Viết chương trình nhập vào số nguyên dương n. Kiểm tra xem n có phải là số chính phương hay không?

Hướng dẫn: Số chính phương là số bằng bình phương của một số nguyên.

2.4.6. Viết chương trình nhập số giờ làm và lương giờ rồi tính số tiền lương tổng cộng. Nếu số giờ làm lớn hơn 40 thì những giờ làm dôi ra được tính 1,5 lần.

2.4.7. Nhập vào ngày, tháng, năm. Kiểm tra xem ngày, tháng, năm đó có hợp lệ hay không? In kết quả ra màn hình.

2.4.8. Nhập vào giờ, phút, giây. Kiểm tra xem giờ, phút, giây đó có hợp lệ hay không? In kết quả ra màn hình.

2.4.9. Viết chương trình nhập vào ngày, tháng, năm hợp lệ. Cho biết năm này có phải là năm nhuận hay không? In kết quả ra màn hình.

2.4.10. Viết chương trình nhập vào ngày tháng năm (dd:mm:yy), cho biết đó là thứ mấy trong tuần.

$$HD: F = Y - 1 + (Y - 1) / 4 + (Y - 1) / 100 + (Y - 1) / 400 + C$$

Y: Năm hiện tại

C: số ngày từ ngày đầu năm đến ngày cần tính – Chú ý năm nhuận

$F \% 7 == 0 \Rightarrow CN, \dots$

2.4.11. Viết chương trình nhập vào một số nguyên n gồm ba chữ số. Xuất ra màn hình chữ số lớn nhất ở vị trí nào?

Ví dụ: $n=291$. Chữ số lớn nhất nằm ở hàng chục (9).

2.4.12. Viết chương trình nhập vào số nguyên n gồm ba chữ số. Xuất ra màn hình theo thứ tự tăng dần của các chữ số.

Ví dụ: $n=291$. Xuất ra 129.

2.4.13. Sinh viên hãy làm lại các bài tập, nhưng trong môi trường C++.

2.5. Kết luận buổi thực hành:

2.5.1. Danh sách sinh viên vắng:

.....

2.5.2. Công tác chuẩn bị bài ở nhà:

.....

.....

2.5.3. Tinh thần thái độ thực hành:

.....

.....

2.5.4. Những lỗi thường mắc phải trong quá trình thực hành:

a/ Lỗi cài đặt phần mềm:

.....

.....

b/ Lỗi tạo tập tin, thao tác chỉnh sửa dữ liệu.....

.....

.....

c/ Lỗi về kiến thức cơ bản của ngôn ngữ lập trình.....

.....

.....

2.5.5. Hướng dẫn công việc ở nhà:

.....

.....

.....

3. THỰC HÀNH CHƯƠNG 4 - HÀM

Mục tiêu:

- Ôn tập củng cố kiến thức cơ bản về:
 - + Cấu trúc hàm, prototype;
 - + Tham trị, tham biến và cách truyền
 - + Miền tồn tại các biến, phân phối vào bộ nhớ
 - + Sử dụng biến cục bộ, toàn cục trong hàm.
 - + Sử dụng tiền xử lý #define
 - + Cách viết các chương trình dạng hàm khác nhau
- Rèn luyện kỹ năng lập trình, vận dụng lý thuyết vào bài tập ở các nội dung trên và có liên quan.

3.1. Ví dụ - Truyền tham trị

Ví dụ 1: Viết hàm tính lũy thừa một số

```
#include <stdio.h>
// khai bao prototype
long power(int, int);
main()
{
    printf("2 mu 5 = %ld\n", power(2, 5));
    printf("2 mu 10 = %ld\n", power(2, 10));
    printf("2 mu 20 = %ld\n", power(2, 20));
}
// ham tinh so mu
long power(int ix, int in)
{
    int i, ip = 1;
    for(i = 1; i <= in; i++)
        ip *= ix;
    return ip;
}
```

Ví dụ 2: Hàm kiểm tra số chẵn, lẻ có sử dụng biến toàn cục

```
#include <stdio.h>
// khai bao prototype
void oddeven();
void negative();
//khai bao bien toan cuc
int inum;
main()
{
    printf("Nhap vao 1 so nguyen : ");scanf("%d",&inum);
    oddeven();
    negative();
}
```

```
// ham kiem tra chan le
void oddeven()
{
    if (inum % 2)
        printf("%d la so le.\n", inum);
    else
        printf("%d la so chan.\n", inum);
}
//ham kiem tra so am
void negative()
{
    if (inum < 0)
        printf("%d la so am.\n", inum);
    else
        printf("%d la so duong.\n", inum);
}
```

Bài 1: Nhập vào chương trình dạng hàm tính tổng, hiệu, tích thương của 2 số thực nhập vào từ bàn phím.

```
#include<stdio.h>
float tong(float a,float b);
float hieu(float a,float b);
float tich(float a,float b);
float thuong(float a,float b);
void main()
{
    float x,y;
    printf("Nhap vao 2 so: ");scanf("%f%f",&x,&y);
    printf("Tong cua 2 so la %1.2f: \n",tong(x,y));
    printf("Hieu cua 2 so la %1.2f: \n",hieu(x,y));
    printf("Tich cua 2 so la %1.2f: \n",tich(x,y));
    printf("Thuong cua 2 so la %1.2f: \n",thuong(x,y));
}
float tong(float a,float b)
{
    return a+b;
}
float hieu(float a,float b)
{
    return a-b;
}
float tich(float a,float b)
{
    return a*b;
}
float thuong(float a,float b)
{
    if (b==0)
```

```

        {printf("Khong the chia cho 0 !");
        exit(1);
        }
    else return a/b;
}

```

Yêu cầu thêm: Thực hiện cách khác là khi khai báo Prototype thì đồng thời viết chương trình con đầy đủ luôn; đặt chương trình con ở các vị trí khác nhau trong chương trình.

Bài 2. Lập chương trình dạng hàm giải và biện luận phương trình bậc 2 có 1 ẩn số
Gợi ý: Cần viết 2 hàm: Khi hệ số của x^2 bằng 0 thì chuyển về giải và biện luận phương trình bậc nhất 1 ẩn; ngược lại thì giải và biện luận phương trình bậc 2.

Bài 3. Viết hàm kiểm tra số nguyên n có phải là **số nguyên tố** hay không?

Hướng dẫn: **Viết hàm ktra_ngto(int x)** kiểm tra x là số nguyên tố hay không

Khai báo biến i, ktra

Cho i chạy từ 2 đến x

Nếu $x \% i = 0$ thì thoát

Nếu $x == i$ thì ktra=1

Ngược lại, ktra=0

Hàm ktra_ngto trả về giá trị ktra

Chương trình chính:

Khai báo biến n, kq

Nhập số n

Gọi hàm: kq = **ktra_ngto(n)**

Nếu kq == 1 thì n là số nguyên tố

Ngược lại, n không phải là số nguyên tố

Bài 4: Viết hàm kiểm tra số nguyên n có phải là **số chính phương** hay không?
 ($25=5^2$, $16=4^2$, $9=3^2$; các số 25, 16, 9 là những số chính phương)

Hướng dẫn: Viết hàm ktra_chinhphuong(int x): kiểm tra x có là số chính phương hay không

Khai báo biến i, ktra

Cho i chạy từ 1 đến x (nếu dùng sqrt(x) thì không cần lệnh thoát)

Nếu $i * i == x$ thì

{ ktra = 1

Thoát }

Nếu ktra==1 thì x là số chính phương

Ngược lại, x không phải là số chính phương

Chương trình chính:

Khai báo biến n, kq

Nhập số n

Gọi hàm: kq = ktra_chinhphuong(n)

Bài 5: Viết hàm tính n! (n là số nguyên nhập vào từ bàn phím)

Hướng dẫn: Viết hàm (function) Giaithua(int n):giai thừa của n
 Khai báo biến I, GT
 Cho i chạy từ 1 đến n
 GT=1
 GT=GT*i
 Trả về kết quả tìm được cho hàm Giaithua
 Chương trình chính:
 Khai báo biến n
 Nhập số n
 Gọi hàm: kq = **Giaithua(n)**

Bài 6: Viết thủ tục tìm **UCLN** và **BCNN** của hai số nhập vào từ bàn phím

Hướng dẫn: Viết hàm (function) int USCLN(int a,int b) tìm ước số chung lớn nhất của hai số a và b
 Lấy trị tuyệt đối hai số a và b
 Chừng nào (a!=0 và b!=0) làm
 Nếu a>b thì a=a-b
 Ngược lại, b=b-a
 Nếu a == 0 thì b là USCLN
 Ngược lại, a là USCLN
 Viết hàm (function) int BSCNN(int a, int b) tìm bội số chung nhỏ nhất của hai số a và b
 BSCNN= (a*b)/USCLN(a,b)
 Chương trình chính:
 Khai báo biến a, b, US, BS
 Nhập hai số a,b
 Gọi hàm:
 US = USCLN(a, b)
 BS = BSCNN(a, b)
 Xuất BS, US ra màn hình

Bài 7: Viết hàm, thủ tục tìm **số lớn nhất** trong hai số. Áp dụng hàm này vào việc tìm số lớn nhất trong ba số nhập vào từ bàn phím

Hướng dẫn: Viết hàm (function) int solonnhat(int a,int b) tìm số lớn nhất trong hai số a và b
 Khai báo biến max
 Gán max = a
 Nếu max<b thì max=b
 Hàm **solonnhat(a,b)** trả về giá trị max
 Chương trình chính:
 Khai báo biến x,y,z;
 Nhập ba số x,y,z;
 In ra số lớn nhất: solonnhat(solonnhat(x,y),z));

3.2. Truyền tham biến

3.2.1. Cho biết kết quả a, b sau khi chạy chương trình sau:

```
#include<stdio.h>
void swap(int &x, int &y)
{
    int t = x; x = y; y = t;
}

int main()
{int a = 5, b = 3;
  swap(a, b);
  printf("Gia tri sau doi la: %d %d",a,b);
  return 0;
}
```

Hỏi: Nếu sửa lại khai báo void swap(int &x, int &y) thành void swap(int x, int y), thì kết quả sau khi chạy chương trình được a, b là bao nhiêu ? Giải thích ?

3.2.2. Cho biết kết quả x, y sau khi chạy chương trình sau:

```
#include<stdio.h>
void TD(int &a,int b)
{
    a=a*2;
    b=b*3;
}

main()
{int x=2,y=3;
  TD(x,y);
  x=x+1; y=y+2;
  printf("x=%d\n y=%d\n",x,y);
}
```

3.2.3. Cho biết kết quả x, y sau khi chạy chương trình sau:

```
#include<stdio.h>
void triple_val(int x);
void triple_adr(int &y);
main()
{int x=2,y=2;
  triple_val(x);
  triple_adr(y);
  printf("x=%d\n",x);
  printf("y=%d\n",y);
}

void triple_val(int x)
{
    x=x*3;
}

void triple_adr(int &y)
{
    y=y*3;
}
```

```

}
3.2.4. Cho biết kết quả x, y sau khi chạy chương trình sau:
#include<iostream.h>
void test(int a,int b);
main()
{
    int x=100,y=1;
    test(x,y);
    cout<<x<<" "<<y;
}
void test(int a,int b)
{
    int x,y;
    a++;
    b=b+4;
    x++;
    y++;
}

```

3.3. Bài tập làm thêm

3.3.1. Viết chương trình tính diện tích và chu vi hình tròn với bán kính được nhập từ bàn phím.

3.3.2. Nhập số nguyên dương n ($0 \leq n < 1000$) và in ra cách đọc của n.

Ví dụ: Nhập $n = 105$. In ra màn hình: *Mot tram le nam*.

Tham khảo chương trình trên C++ sau đây, sau đó viết lại chương trình trong C:

```

#include<iostream.h>
using namespace std;
void Nhap(int &a)
{
    cout<<"Nhap so nguyen n:";
    cin>>a;
}
void DocSo(int n)
{
    int tram=n/100;
    int chuc=n/10%10;
    int dv=n%10;
    switch(tram)
    {
        case 1: cout<<"Mot Tram ";break;
        case 2: cout<<"Hai Tram ";break;
        case 3: cout<<"Ba Tram ";break;
        case 4: cout<<"Bon Tram ";break;
        case 5: cout<<"Nam Tram ";break;
        case 6: cout<<"Sau Tram ";break;
        case 7: cout<<"Bay Tram ";break;
        case 8: cout<<"Tam Tram ";break;
    }
}

```

```

case 9: cout<<"Chin Tram ";break;
}
switch(chuc)
{
case 0: if (dv==0)
{cout<<" ";break;}
else
{cout<<"Le ";break;}
case 1: cout<<"Muoi ";break;
case 2: cout<<"Hai Muoi ";break;
case 3: cout<<"Ba Muoi ";break;
case 4: cout<<"Bon Muoi ";break;
case 5: cout<<"Nam Muoi ";break;
case 6: cout<<"Sau Muoi ";break;
case 7: cout<<"Bay Muoi ";break;
case 8: cout<<"Tam Muoi ";break;
case 9: cout<<"Chin Muoi ";break;
}
switch(dv)
{
case 5:if (chuc==0)
{cout<<"Lam ";break;}
else
{cout<<"Nam ";break;}
case 1: cout<<"Mot"<<endl;break;
case 2: cout<<"Hai"<<endl;break;
case 3: cout<<"Ba"<<endl;break;
case 4: cout<<"Bon"<<endl;break;
case 6: cout<<"Sau"<<endl;break;
case 7: cout<<"Bay"<<endl;break;
case 8: cout<<"Tam"<<endl;break;
case 9: cout<<"Chin"<<endl;break;
}
}
main()
{
    int n;
    Nhap(n);
    DocSo(n);
}

```

3.3.3. Nhập vào 6 số thực a, b, c, d, e và f. Giải hệ phương trình sau :

$$\begin{cases} ax + by = c \\ dx + ey = f \end{cases}$$

3.3.4. Viết chương trình nhập số nguyên dương n gồm k chữ số ($0 < k \leq 5$), đếm xem n có bao nhiêu chữ số chẵn và bao nhiêu chữ số lẻ.

3.3.5. Viết chương trình nhập số nguyên dương n gồm k chữ số ($0 < k \leq 5$), đếm xem n có bao nhiêu chữ số là số nguyên tố.

3.3.6. Viết chương trình nhập số nguyên dương n gồm k chữ số ($0 < k \leq 5$), tính tổng các ước số dương của n .

Ví dụ: Nhập $n=6$, Tổng các ước số từ 1 đến n : $1+2+3+6=12$

3.3.7. Viết chương trình nhập số nguyên dương n gồm k chữ số ($0 < k \leq 5$), sau đó nhập một số nguyên x có một chữ số, tìm vị trí xuất hiện của chữ số có giá trị x trong n .

Ví dụ: Nhập $n=1526$, $x=2$

Kết quả: Chuỗi 2 ở vị trí thứ 3.

3.3.8. Viết chương trình nhập số nguyên dương n gồm k chữ số ($0 < k \leq 5$), kiểm tra xem các chữ số của n có được sắp thứ tự không.

Ví dụ: Nhập $n=1569$ hoặc $n=8521$

Kết quả: Có thứ tự.

3.3.9. Viết chương trình nhập 2 số a, b sao cho: số lớn nhất trong 2 số phải là một số dương và chia hết cho 7. Nếu nhập sai phải yêu cầu nhập lại cho đến khi đúng.

3.3.10. Viết chương trình nhập số nguyên dương n gồm k chữ số ($0 < k \leq 5$), tính giá trị trung bình các chữ số chẵn trong n .

3.3.11. Sinh viên hãy làm lại các bài tập, nhưng trong môi trường C++.

3.4. Kết luận buổi thực hành:

3.4.1. Danh sách sinh viên vắng:

.....

3.4.2. Công tác chuẩn bị bài ở nhà:

.....

.....

3.4.3. Tinh thần thái độ thực hành:

.....

.....

3.4.4. Những lỗi thường mắc phải trong quá trình thực hành:

a/ Lỗi cài đặt phần mềm:

.....

.....

b/ Lỗi tạo tập tin, thao tác chỉnh sửa dữ liệu.....

.....
.....
.....
c/ Lỗi về kiến thức cơ bản của ngôn ngữ lập trình.....

.....
.....
.....
.....
.....
.....
3.4.5. Hướng dẫn công việc ở nhà:

4. THỰC HÀNH CHƯƠNG 5 – MẢNG

Mục tiêu:

- Ôn tập củng cố kiến thức cơ bản về:
 - + Cấu trúc mảng, khai báo, các loại mảng, truy cập các phần tử mảng;
 - + Chuỗi, khai báo, truy cập các phần tử chuỗi, các phép toán về chuỗi;
 - + Cách viết các chương trình dạng mảng, chuỗi khác nhau
- Rèn luyện kỹ năng lập trình, vận dụng lý thuyết vào bài tập ở các nội dung trên và có liên quan.

4.1. Mảng một chiều – Ví dụ

Ví dụ 1: Viết chương trình nhập vào n số nguyên. Tính và in ra trung bình cộng.

```
#include <stdio.h>
```

```
main()
```

```
{
    int ia[50], i, in, isum = 0;
    printf("Nhập vào giá trị n: ");scanf("%d", &in);
    //Nhập dữ liệu vào mảng
    for(i = 1; i <= in; i++)
    {
        printf("Nhập vào phần tử thứ %d: ", i);
        scanf("%d", &ia[i]); //Nhập giá trị cho phần tử thứ i
    }
    //Tính tổng giá trị các phần tử
    for(i = 1; i <= in; i++)
        isum += ia[i]; //cộng dồn từng phần tử vào isum
    printf("Trung bình cộng: %.2f\n", (float) isum/in);
}
```

Ví dụ 2: Nhập vào một dãy n số; in dãy số vừa nhập; sắp xếp các số theo thứ tự tăng; in dãy số sau khi sắp xếp.

Yêu cầu: Sinh viên thực hành làm, sau đó tham khảo bài mẫu sau và sửa chương trình nhập mảng tự động dùng hàm thư viện rand():

```
#include<conio.h>
```

```
#include<stdio.h>
```

```
void Nhap(int a[],int N)
```

```
{
    int i;
    for(i=0; i< N; i++)
    {
        printf("Phần tử thứ %d: ",i);scanf("%d",&a[i]);
    }
}
```

```
void InMang(int a[], int N)
```

```
{
    int i;
    for (i=0; i<N;i++)
```

```

        printf("%d ",a[i]);
        printf("\n");
    }
void SapXep(int a[], int N)
{
    int t,i;
    for(i=0;i<N-1;i++)
        for(int j=i+1;j<N;j++)
            if (a[i]>a[j])
            {
                t=a[i]; a[i]=a[j]; a[j]=t;
            }
}
main()
{
    int b[20], N;
    printf("So phan tu thuc te cua mang N= ");scanf("%d",&N);
   Nhap(b,N);
    printf("Mang vua nhap: ");
    InMang(b,N);
    SapXep(b,N); /* G?i hàm s?p x?p */
    printf("Mang sau khi sap xep: ");
    InMang(b,N);
    getch();
}

```

Kết quả chạy chương trình:

```

So phan tu thuc te cua mang N= 5
Phan tu thu 0: 2
Phan tu thu 1: 7
Phan tu thu 2: 1
Phan tu thu 3: -3
Phan tu thu 4: 4
Mang vua nhap: 2 7 1 -3 4
Mang sau khi sap xep: -3 1 2 4 7

```

Bài tập mảng một chiều

Bài tập 2. Viết chương trình sắp xếp một mảng các số nguyên theo thứ tự tăng dần sau khi đã loại bỏ các phần tử trùng nhau.

Bài tập 3. Viết chương trình nhập vào một mảng các số thực, hãy xuất ra màn hình:

- Phần tử lớn nhất của mảng.
- Phần tử nhỏ nhất của mảng.
- Tổng của các phần tử là số nguyên tố trong mảng .
- Tổng của các phần tử trong mảng là các số âm .
- Tổng của các phần tử trong mảng là các số chẵn .

Bài tập 4. Viết chương trình nhập vào một dãy các số theo thứ tự tăng, nếu nhập sai quy cách thì yêu cầu nhập lại. In dãy số sau khi đã nhập xong. Nhập thêm một số

mới và chèn số đó vào dãy đã có sao cho dãy vẫn đảm bảo thứ tự tăng. In lại dãy số để kiểm tra.

Bài tập 5. Viết chương trình nhập vào một mảng số tự nhiên. Hãy xuất ra màn hình:

- Dòng 1 : gồm các số lẻ, tổng cộng có bao nhiêu số lẻ.
- Dòng 2 : gồm các số chẵn, tổng cộng có bao nhiêu số chẵn.
- Dòng 3 : gồm các số nguyên tố.
- Dòng 4 : gồm các số không phải là số nguyên tố.

Bài tập 6. Viết chương trình tính tổng bình phương của các số âm trong một mảng các số nguyên.

Bài tập 7. Viết chương trình thực hiện việc đảo một mảng một chiều.

Ví dụ : 1 2 3 4 5 7 9 10 đảo thành 10 9 7 5 4 3 2 1

Bài tập 8. Cho dãy a gồm n số nguyên có thứ tự tăng dần. Nhập vào một phần tử nguyên X, viết hàm chèn X vào dãy sao cho dãy vẫn có thứ tự tăng dần (*không sắp xếp*).

Bài tập 9. Viết chương trình tìm số lẻ nhỏ nhất lớn hơn mọi số chẵn có trong mảng.

Bài tập 10. Viết hàm tìm giá trị chẵn nhỏ nhất nhỏ hơn mọi giá trị lẻ trong mảng các số nguyên.

Bài tập 11. Viết hàm tìm phần tử xuất hiện nhiều nhất trong mảng các số nguyên.

Bài tập 12. Viết chương trình đếm và liệt kê các mảng con tăng dần trong mảng một chiều các số nguyên.

Ví dụ: 6 5 3 2 3 4 2 7 các dãy con tăng dần là 2 3 4 và 2 7

Bài tập 13. Viết chương trình tìm mảng con tăng dần có tổng lớn nhất trong mảng một chiều

Bài tập 14. Viết chương trình tách 1 mảng các số nguyên thành 2 mảng a và b, sao cho kết quả thu được là: - Mảng a chứa toàn số lẻ tăng dần. - Mảng b chứa toàn số chẵn giảm dần. (*Không dùng sắp xếp*)

Hướng dẫn : Tìm vị trí chèn thích hợp khi trích phần tử từ mảng ban đầu. Ví dụ:

Mảng ban đầu: 9 3 8 2 7 5 1 0 10

Mảng a: 1 3 5 7 9

Mảng b: 10 8 2

4.2. Mảng nhiều chiều – Ví dụ

Ví dụ 1: Viết chương trình cho phép nhập 2 ma trận a, b có m dòng n cột, thực hiện phép toán cộng hai ma trận a, b và in ma trận kết quả lên màn hình:

Yêu cầu: Sinh viên thực hành làm, sau đó tham khảo bài mẫu sau:

(Trong ví dụ này, có sử dụng hàm để làm ngắn gọn hơn chương trình. Ta sẽ viết các hàm: nhập 1 ma trận từ bàn phím, hiển thị ma trận lên màn hình, cộng 2 ma trận).

```
#include<conio.h>
```

```

#include<stdio.h>
void Nhap(int a[][10],int M,int N)
{
    int i,j;
    for(i=0;i<M;i++)
        for(j=0; j<N; j++)
        {
            printf("Phan tu o dong %d cot %d: ",i,j);
            scanf("%d",&a[i][j]);
        }
}
void InMaTran(int a[][10], int M, int N)
{
    int i,j;
    for(i=0;i<M;i++)
    {
        for(j=0; j< N; j++)
            printf("%d ",a[i][j]);
        printf("\n");
    }
}
/* Cong 2 ma tran A & B ket qua la ma tran C*/
void CongMaTran(int a[][10],int b[][10],int M,int N,int c[][10])
{
    int i,j;
    for(i=0;i<M;i++)
        for(j=0; j<N; j++)
            c[i][j]=a[i][j]+b[i][j];
}
int main()
{
    int a[10][10], b[10][10], M, N;
    int c[10][10];/* Ma tran tong*/
    printf("So dong M= ");scanf("%d",&M);
    printf("So cot M= ");scanf("%d",&N);
    printf("Nhap ma tran A\n");    Nhap(a,M,N);
    printf("Nhap ma tran B\n"); Nhap(b,M,N);
    printf("Ma tran A: \n");
    InMaTran(a,M,N);
    printf("Ma tran B: \n");
    InMaTran(b,M,N);
    CongMaTran(a,b,M,N,c);
    printf("Ma tran tong C:\n");InMaTran(c,M,N);
    getch();
    return 0;
}

```

Kết quả chạy thử:

```

So dong M= 3
So cot M= 4
Nhap ma tran A
Phan tu o dong 0 cot 0: 1
Phan tu o dong 0 cot 1: 3
Phan tu o dong 0 cot 2: 5
Phan tu o dong 0 cot 3: 7
Phan tu o dong 1 cot 0: 2
Phan tu o dong 1 cot 1: 4
Phan tu o dong 1 cot 2: 6
Phan tu o dong 1 cot 3: 8
Phan tu o dong 2 cot 0: 11
Phan tu o dong 2 cot 1: 12
Phan tu o dong 2 cot 2: 13
Phan tu o dong 2 cot 3: 14
Nhap ma tran B
Phan tu o dong 0 cot 0: 1
Phan tu o dong 0 cot 1: 2
Phan tu o dong 0 cot 2: 3
Phan tu o dong 0 cot 3: 4
Phan tu o dong 1 cot 0: 5
Phan tu o dong 1 cot 1: 6
Phan tu o dong 1 cot 2: 7
Phan tu o dong 1 cot 3: 8
Phan tu o dong 2 cot 0: 9
Phan tu o dong 2 cot 1: 10
Phan tu o dong 2 cot 2: 11
Phan tu o dong 2 cot 3: 12
Ma tran A:
1 3 5 7
2 4 6 8
11 12 13 14
Ma tran B:
1 2 3 4
5 6 7 8
9 10 11 12
Ma tran tong C:
2 5 8 11
7 10 13 16
20 22 24 26

```

Bài tập mảng nhiều chiều

Bài tập 2. Nhập vào một ma trận 2 chiều gồm các số thực, in ra tổng của các phần tử trên đường chéo chính của ma trận này.

Hướng dẫn: Ma trận a có M dòng, N cột thì các phần tử của đường chéo chính là các phần tử có dạng: $a[i][i]$ với $i \in [0 \dots \min(M, N) - 1]$.

Bài tập 3. Viết chương trình nhập vào một ma trận (mảng hai chiều) các số nguyên, gồm m hàng, n cột. In ma trận đó lên màn hình. Nhập một số nguyên khác vào và xét xem có phần tử nào của ma trận trùng với số này không? Ở vị trí nào? Có bao nhiêu phần tử?

Bài tập 4. Viết chương trình chuyển đổi vị trí từ dòng thành cột của một ma trận vuông 4 hàng 4 cột (ma trận chuyển vị). Sau đó viết cho ma trận tổng quát cấp $m \times n$.

Ví dụ:

| | |
|---------|---------|
| 1 2 3 4 | 1 2 9 1 |
| 2 5 5 8 | 2 5 4 5 |
| 9 4 2 0 | 3 5 2 8 |
| 1 5 8 6 | 4 8 0 6 |

Bài tập 5. Viết chương trình nhập vào hai ma trận A có cấp m, k và B có cấp k, n. In hai ma trận lên màn hình. Tích hai ma trận A và B là ma trận C được tính bởi công thức:

$$C_{ij} = a_{i1} * b_{1j} + a_{i2} * b_{2j} + a_{i3} * b_{3j} + \dots + a_{ik} * b_{kj} \quad (i=0,1,2,\dots,m-1; j=0,1,2,\dots,n-1)$$

Tính ma trận tích C và in kết quả lên màn hình.

Bài tập 6. Viết chương trình thực hiện việc trộn hai dãy có thứ tự thành một dãy có thứ tự. Yêu cầu không được trộn chung rồi mới sắp thứ tự. Khi trộn phải tận dụng được tính chất đã sắp của hai dãy con.

4.3.15. Sinh viên hãy làm lại các bài tập, nhưng trong môi trường C++.

4.4. Kết luận buổi thực hành:

4.4.1. Danh sách sinh viên vắng:

.....

4.4.2. Công tác chuẩn bị bài ở nhà:

.....

.....

4.4.3. Tinh thần thái độ thực hành:

.....

.....

4.4.4. Những lỗi thường mắc phải trong quá trình thực hành:

a/ Lỗi cài đặt phần mềm:

.....

.....

b/ Lỗi tạo tập tin, thao tác chỉnh sửa dữ liệu.....

.....

.....

.....

c/ Lỗi về kiến thức cơ bản của ngôn ngữ lập trình.....

.....

.....

.....

.....

.....

.....

.....

.....

4.4.5. Hướng dẫn công việc ở nhà:

.....

.....

.....

.....

.....

.....

.....

9. ÔN TẬP

Mục tiêu:

- Ôn tập củng cố kiến thức cơ bản của môn học.
- Thảo luận và giải đáp các ý kiến của sinh viên
- Rèn luyện kỹ năng lập trình, vận dụng lý thuyết vào bài tập ở các nội dung trong chương trình môn học.

Bài tập

9.1. Tính tổng các số thực dương của một mảng các số thực.

Yêu cầu sinh viên nhập dữ liệu theo 2 cách: thủ công và máy chọn ngẫu nhiên.

9.2. Cho trước một mảng các số nguyên A. Hãy tách mảng A thành 2 mảng, một mảng B chỉ gồm các số chẵn, một mảng C chỉ gồm các số lẻ.

9.3. Làm lại bài 9.2. nhưng tách các số nguyên tố và còn lại thành 2 mảng.

9.4. Cho trước một mảng các số nguyên. Hãy sắp xếp mảng các số lẻ tăng dần, các số còn lại giữ nguyên vị trí

9.5. Nhập mảng các số nguyên:

- Sau khi nhập xong đã tự sắp xếp tăng dần, sau đó in mảng
- Xóa tất cả số lớn nhất trong mảng, sau đó in mảng
- Xóa tất cả các số âm trong mảng, sau đó in mảng

9.6. Lập chương trình để máy tính nói chuyện với người:

- Máy “nói” bằng các câu hỏi trên màn hình
- Người nói bằng cách gõ vào từ bàn phím

Nội dung cuộc nói chuyện: Máy hỏi bạn có gia đình chưa? Nếu chưa thì khuyên “Bạn hãy bình tĩnh và thận trọng!”, nếu rồi thì hỏi xem có mấy con. Nếu có 3 thì bảo “Hơi nhiều !”, nếu trên 3 thì bảo “Nhiều quá !”,

9.10. Làm lại các bài tập phần Ôn tập nhưng với lập trình hàm

Kết luận buổi thực hành:

1. Danh sách sinh viên vắng:.....

.....

2. Công tác chuẩn bị bài ở nhà:.....

.....

.....

3. Tinh thần thái độ thực hành:

.....

.....

4. Những lỗi thường mắc phải trong quá trình thực hành:

a/ Lỗi cài đặt phần mềm:.....

.....

.....

b/ Lỗi tạo tập tin, thao tác chỉnh sửa dữ liệu.....

.....

.....

.....

c/ Lỗi về kiến thức cơ bản của ngôn ngữ lập trình.....

.....

.....

.....

.....

5. Hướng dẫn công việc ở nhà:

.....

.....

.....

.....

TÀI LIỆU THAM KHẢO

1. Tài liệu tiếng Việt

1.1. Ngô Trung Việt - Ngôn ngữ lập trình C và C++ - Bài giảng- Bài tập - Lời giải mẫu NXB giao thông vận tải 1995

1.2. GERALD LEBLANC Turbo C, người dịch Lê Văn Doanh – Nhà xuất bản khoa học kỹ thuật Hà Nội 1996.

1.3. Lê Văn Doanh - 101 thuật toán và chương trình bằng ngôn ngữ C - Nhà xuất bản khoa học kỹ thuật Hà Nội 2002.

1.4. Phạm Văn Ất – Kỹ thuật lập trình C - Nhà xuất bản giao thông vận tải Hà Nội 2006.

1.5. G.S. Phạm Văn Ất – C++ và lập trình hướng đối tượng - Nhà xuất bản khoa học kỹ thuật Hà Nội 2000.

1.6. Hanoi Aptech Computer Education Center - Giáo trình **Lập trình C căn bản**.

2. Tài liệu tiếng Anh

- 2.1. B. Kernighan and D. Ritchie - The C Programming Language - Prentice Hall 1989
- 2.2. Programmer's guide Borland C++ Version 4.0 - Borland International, Inc 1993
- 2.3. Bile - Nabaiyoti - TURBO C++ - The Waite Group's UNIX 1991
- 2.4. IBM_Turbo_C_Users_Guide_1987
- 2.5. Ivor Horton - Beginning C: Fifth Edition 2013.